DOTTORATO DI RICERCA IN INFORMATICA
(CICLO XXXVII)

PhD Thesis

Advancing Multilingual DRS-based Semantic Parsing and Generation: A Framework for Data Transformation, Robust Evaluation, and Task Reversibility

by

Muhammad Saad Amin

UNIVERSITA' DEGLI STUDI DI TORINO
Dipartimento di Informatica
Torino, Italia

Supervisors

prof. Alessandro Mazzei
prof. Luca Anselma

PhD Coordinator
prof. Viviana Patti

Academic Years
(2021–2024)

Contents

Abstract

Recent advancements in semantic parsing and text generation through Discourse Representation Structures (DRS) underscore the critical need for innovative methodologies to enhance neural model performance, particularly in multilingual and resource-constrained environments. This research presents a comprehensive framework addressing these challenges through multiple complementary approaches: data transformation techniques, alternative evaluation methodologies, and task reversibility analysis.

The foundation of this work lies in novel data transformation strategies, encompassing both data augmentation and delexicalization. These techniques employ multilingual and multifaceted approaches, such as manipulating named entities, leveraging WordNet-based lexical substitutions, applying supersenses, and implementing grammatical transformations. The effectiveness of these methods has been demonstrated across typologically diverse languages: English, Italian, and Urdu. For English, the augmentation framework expanded the Parallel Meaning Bank (PMB) dataset ninefold, yielding substantial improvements in model performance. In Italian, the application of cross-lingual resources led to significant enhancements in semantic parsing and generation capabilities. For Urdu, a low-resource language, a novel rule-based alignment method was developed to transform English DRS, complemented by various augmentation strategies.

A key contribution of this research is the introduction of innovative bidirectional evaluation methodologies. The Parse-Generate (Pars-Gen) and Generate-Parse (Gen-Pars) approaches provide a holistic assessment framework that addresses the limitations of traditional metrics. While SMATCH effectively captures structural overlaps, it may miss nuances in linguistic expression. Conversely, generation metrics like BLEU, COMET, and BERTScore often overlook core semantic equivalences. This dual evaluation approach offers a more comprehensive assessment of system performance across languages.

Furthermore, the research explores task reversibility in semantic processing through Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG) pipelines. This investigation reveals complex dynamics between error propagation and mitigation across languages, with English demonstrating the highest stability, Italian showing moderate variations, and Urdu exhibiting the most volatility. The analysis spans multiple dimensions, including sentence length, structural complexity, sentence type, polarity, and voice, providing valuable insights into error behavior patterns.

Extensive experiments utilizing state-of-the-art neural language models, including byT5, mT5, T5, and mBART, as well as LSTM-based sequence-to-sequence architec-

tures, have demonstrated significant performance improvements across multiple evaluation metrics. These advancements represent a substantial contribution to computational semantics, introducing novel approaches for improving semantic parsing and text generation across diverse linguistic contexts. Moreover, they establish a foundation for developing more robust, generalizable, and linguistically inclusive natural language processing systems, particularly beneficial for low-resource languages and limited-data scenarios.

Acknowledgments

I would like to express my deepest gratitude to my supervisors prof. Alessandro Mazzei and prof. Luca Anselma, whose invaluable guidance, encouragement, and unwavering support have been instrumental throughout the course of my research. Their profound expertise, constructive feedback, and mentorship have not only shaped the direction of this thesis but have also greatly influenced my academic and professional growth. I am truly fortunate to have had the opportunity to work under their kind supervision, and I will forever be grateful for their time, patience, and belief in my abilities.

I would also like to extend my heartfelt thanks to my family. Their constant love, understanding, and encouragement have been the cornerstone of my motivation. Without their sacrifices, unwavering support, and patience throughout this journey, I would not have been able to pursue my academic aspirations. To my parents, thank you for instilling in me the values of hard work and perseverance.

I am also incredibly grateful to my colleagues and friends. Their insightful discussions and moral support have made this journey not only manageable but also enjoyable. Whether it was through shared experiences in the lab, helpful advice, or just being there during moments of stress, their contributions have been indispensable. I am especially thankful for the intellectual exchange and the moments of levity that helped me stay grounded throughout this demanding process.

To each and every one of you, thank you for being an essential part of this journey.

Chapter 1

Introduction

The field of natural language processing (NLP) has seen remarkable advancements in recent years, particularly in tasks that bridge the gap between human language and formal representations of meaning [Wang et al., 2023a]. Two fundamental tasks in this domain are semantic parsing and natural language generation. Semantic parsing involves converting natural language text into structured, formal representations of meaning, while natural language generation reverses this process, producing human-readable text from these formal representations [Wang et al., 2023a].

Figure 1.1 illustrates the bidirectional process of semantic parsing and natural language generation in a multilingual context. In Figure 1.1(a), we demonstrate semantic parsing, which involves the conversion of multilingual text into a formal meaning representation. This process extracts the semantic content from input sentences in various languages (English, Italian, Urdu, German, and Dutch) and represents it in a structured format. Conversely, Figure 1.1(b) depicts natural language generation, the reverse process of semantic parsing. Here, the formal meaning representation is transformed back into multilingual text, generating translations in the aforementioned languages. This bidirectional approach highlights the interconnectedness of semantic understanding and language generation in multilingual natural language processing systems [Bos, 2023].

At the heart of these tasks lies the challenge of accurately capturing and manipulating the intricate semantics of human language. Formal meaning representations, such as Discourse Representation Structures (DRS), have emerged as powerful tools for this purpose [Bos, 2021]. DRS provides a structured framework for representing the meaning of text, encompassing various linguistic phenomena including entities, events, time expressions, and logical relationships [Amin et al., 2024, Zhang et al., 2024, Wang et al., 2021b].

To accurately capture the semantics of natural language, formal meaning representations like DRS often draw from first-order logic (FoL), a powerful system for representing predicates, entities, and logical relationships. First-order logic allows for the formalization of statements about objects and their properties, making it well-suited for encoding the kind of rich semantic information required in tasks like semantic parsing and natural language generation. For instance, a simple sentence like "Tom was rude."

can be translated into a logical form using FoL as

$$\exists x \, (\text{Tom}(x) \wedge \text{Rude}(x))$$

where the existential quantifier $\exists$ asserts the existence of an entity x, identified as Tom, who possesses the property of being rude. This logical representation captures both the subject and predicate of the sentence in a structured, interpretable form. By using formal representations like DRS, which often build on FoL, we can effectively handle the complexities of natural language, such as coreference, negation, and temporal expressions, making it possible to bridge the gap between human language and machine-understandable meaning [Bos, 2008, 2021, 2023].

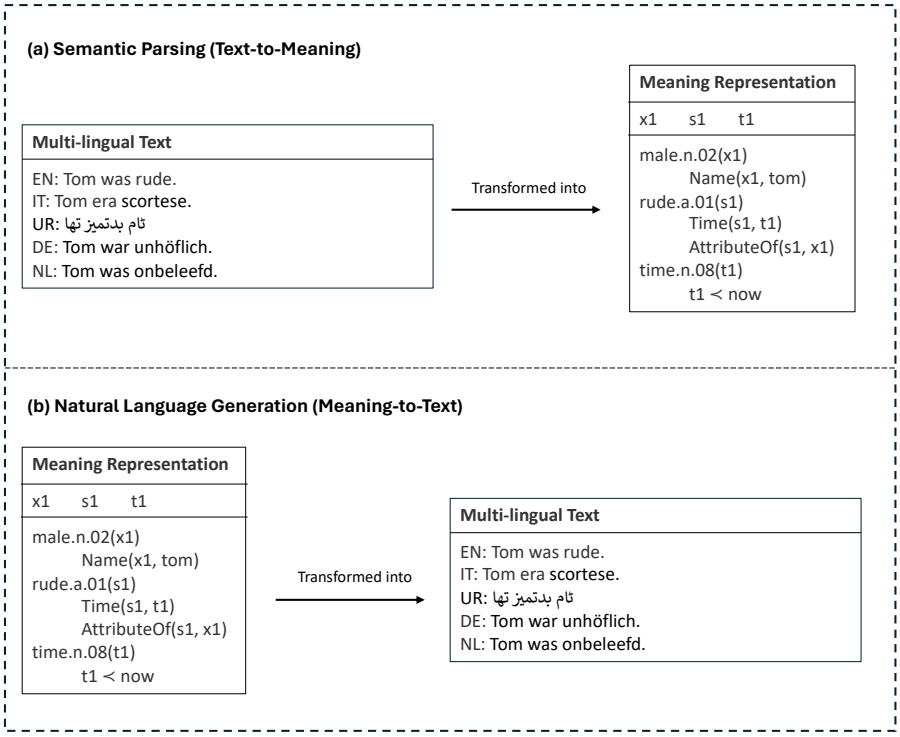


Figure 1.1: Graphical representation of the reversible processes of (a) semantic parsing (Text-to-Meaning) and (b) natural language generation (Meaning-to-Text) with multilingual textual variants.

Despite the progress made in semantic parsing and generation, significant challenges remain, particularly when dealing with low-resource languages or aiming to improve the robustness and generalizability of models [Amin et al., 2022b]. This dissertation addresses several of these challenges through innovative approaches in data augmentation, data delexicalization, resource development for underrepresented languages, and novel evaluation and error mitigation frameworks. By incorporating advanced evaluation methods and exploring reversible task pipelines, this work aims to advance both the

accuracy and resilience of DRS-based systems across multilingual and diverse linguistic contexts.

Our work primarily centers on three languages: English, Urdu, and Italian. The choice of these languages is motivated by both their linguistic diversity and their relevance to our research objectives. English serves as the primary language for many existing semantic resources [van Noord et al., 2018, Liu et al., 2019, Fu et al., 2020], Urdu represents a low-resource language [Jafar and Jafar, 2022] — for which we develop novel semantic tools, and Italian offers a case study for applying our techniques to a language with limited annotated data [Fancellu et al., 2019].

In the context of our experiments, we employed a range of neural sequence-to-sequence (seq-to-seq) models, encompassing both recurrent and transformer-based architectures. The seq-to-seq paradigm, particularly through the use of bi-directional Long Short-Term Memory (bi-LSTM) networks, has proven effective for various natural language processing tasks, including semantic parsing and text generation. Bi-LSTMs are capable of capturing dependencies within sequences more effectively than traditional recurrent neural networks (RNNs), although they still face limitations in handling very long-range dependencies. In contrast, transformer-based models, such as Bidirectional and Auto-Regressive Transformer (BART) [Lewis et al., 2020] and Text-to-Text Transfer Transformer (T5) [Liu et al., 2023] including different variants byT5 [Xue et al., 2022] and mT5 [Xue et al., 2021], have demonstrated superior performance due to their self-attention mechanisms, which can model relationships across entire sequences without the sequential nature of recurrence. BART is a denoising autoencoder for sequence generation tasks, while T5 frames all NLP tasks as text-to-text problems. byT5 is a monolingual model specifically for English text, whereas mT5 is a multilingual variant capable of handling multiple languages, making it particularly suitable for cross-lingual semantic parsing and text generation. These transformer models have been integral to our experiments, providing state-of-the-art results and demonstrating their robustness across multilingual datasets, including English, Italian, and Urdu. By leveraging these models, we explored the intricacies of meaning representation and language generation, pushing the boundaries of semantic parsing and text generation tasks in both monolingual and multilingual settings.

1.1 Research Objectives

The primary aim of this research is to enhance the performance, robustness, and applicability of semantic parsing and generation models across diverse linguistic contexts. To this end, we explore the following key areas:

1.1.1 Data Augmentation for DRS

Data augmentation has emerged as a powerful technique to improve the performance and generalizability of machine learning models, particularly in NLP tasks [Jaszczolt and Jaszczolt, 2023, Yang et al., 2022]. However, its application to logical data representations such as DRS presents unique challenges and opportunities. DRS, as a formal meaning representation, captures complex semantic structures and relationships that

go beyond surface-level text. This complexity makes traditional text-based augmentation techniques potentially inadequate or even harmful if applied naively [Imamura and Sumita, 2018]. The intricate connections between lexical entities, logical operators, and semantic roles within a DRS require careful consideration when developing augmentation strategies.

Our research focuses on developing and implementing multi-faceted data augmentation approaches for both semantic parsing (Text-to-DRS) and text generation (DRS-to-Text) tasks. We explore three main augmentation strategies: named entity augmentation, WordNet-based lexical substitutions, and grammatical transformations through tense variations. These approaches aim to significantly expand the Parallel Meaning Bank (PMB)[1] dataset while maintaining semantic correctness and contextual similarity.

For named entities, we investigate both "inside context" or "*in-context*" (substituting entities within the same dataset) and "outside context" or "*out-of-context*" (substituting entities outside the dataset) augmentation, replacing proper nouns with contextually appropriate and novel entities. Lexical augmentation involves WordNet-based lexical substitutions, including replacing common nouns with hyponyms, verbs with troponyms, adjectives with antonyms, and adverbs with synonyms. We also explore the use of a novel supersenses approach for nouns to enhance generalization. Grammatical augmentation focuses on tense variations, converting sentences between present, past, and future tenses.

The structured nature of DRS raises questions about the feasibility and methods of augmenting such representations while maintaining semantic coherence and logical consistency. Unlike free-form text, where simple substitutions or insertions might be sufficient, augmenting DRS requires a deep understanding of the underlying semantic and logical structure.

Another critical aspect is the interplay between in-context and out-of-context vocabulary in DRS parsing and generation. The role of world knowledge and pragmatic understanding in these tasks adds another layer of complexity to the augmentation process. As models become more sophisticated, leveraging pre-trained language models and transfer learning techniques, the question of how to effectively augment data to improve their performance becomes increasingly relevant.

In light of these challenges and opportunities, we investigate the following research questions:

RQ1.1: Is it possible to augment a logical data representation such as DRS while maintaining semantic coherence?

RQ1.2: How can we generate new data that is contextually similar to the original DRS representations?

RQ1.3: What roles do in-context and out-of-context vocabulary play for character-level and word-level decoder models in DRS parsing and generation?

RQ1.4: How does grammatical, semantic, and pragmatic world knowledge influence the learning process in DRS-based tasks?

RQ1.5: Does augmentation lead to improved performance when training sequence-to-sequence models like Long Short-Term Memory (LSTM) or fine-tuning Transformer

[1] The PMB is developed at the University of Groningen as part of the NWO-VICI project "Lost in Translation – Found in Meaning" (Project number 277-89-003), led by Johan Bos.

models for DRS tasks?
RQ1.6: How do pre-trained large language models (LLMs) like ChatGPT and Claude interpret and process DRS structures when given as prompts?
RQ1.7: How does the quality of augmented data, characterized by semantic and contextual accuracy, influence the effectiveness of data augmentation in enhancing performance in semantic parsing and natural language generation tasks?
RQ1.8: What is the relative contribution of manually corrected (gold) data compared to larger volumes of potentially less accurate silver data in improving model performance?

By addressing these questions, we aim to develop novel and effective data augmentation techniques specifically tailored for DRS, potentially improving the performance and robustness of models in semantic parsing and generation tasks. These findings underscore the effectiveness of our augmentation strategies in enhancing model performance for complex semantic parsing and generation tasks. Moreover, it highlights the potential of multi-faceted data augmentation in improving the robustness and generalization capabilities of neural models in the domain of DRS-based semantic processing. This research not only contributes to advancing the state-of-the-art in DRS parsing and generation for English but also lays the groundwork for applying similar techniques to other languages and semantic formalisms, as explored in subsequent sections.

1.1.2 Data Delexicalization for Enhanced Generalization

Delexicalization is a technique that has shown promising results in various NLP tasks, particularly in improving model generalization. However, its application to formal meaning representations like DRS presents unique challenges and opportunities that need thorough investigation. DRS, as a semantic formalism, relies heavily on lexical entities that are often tightly integrated with external knowledge bases such as WordNet and VerbNet [Zhang et al., 2024]. This integration provides rich semantic information but also introduces complexities when attempting to delexicalize the data. The challenge lies in maintaining the structural and semantic integrity of the DRS while abstracting away from specific lexical choices.

The potential benefits of delexicalization in DRS processing are significant. By reducing the dependency on specific lexical items, models could potentially generalize better to unseen vocabulary and novel semantic constructions. This is particularly important in multilingual or cross-domain applications where lexical variation can be substantial. However, the process of delexicalization in DRS is not straightforward. It requires careful consideration of how to represent abstract concepts without losing critical semantic information. The use of supersenses for nouns, for instance, offers a potential middle ground between full lexicalization and complete abstraction. Furthermore, the interaction between delexicalization and other techniques, such as data augmentation, remains an open question. There is potential for synergy between these approaches, but there are also risks of information loss or the introduction of inconsistencies.

The advent of large pre-trained language models adds another dimension to this research area. Understanding how these models interact with delexicalized DRS data, both in pre-training and fine-tuning scenarios, could provide insights into more effective ways of leveraging these powerful models for semantic parsing and generation tasks.

In light of these considerations, we pose the following additional research questions:

RQ2.1: How can we effectively delexicalize DRS representations while maintaining their connections to external lexical databases like WordNet and VerbNet?
RQ2.2: Can the use of supersenses for nouns contribute to enhancing the generalization power of neural models in DRS tasks?
RQ2.3: What is the impact of combining logically delexicalized data with fully lexicalized data on model performance?
RQ2.4: How do delexicalization and augmentation techniques interact to affect model performance?
RQ2.5: What are the differences in behavior between pre-training and fine-tuning approaches when applied to delexicalized DRS data?

By addressing these questions, we aim to develop effective delexicalization strategies for DRS that can enhance model generalization while preserving the rich semantic information encoded in these structures. This research has the potential to significantly improve the robustness and applicability of DRS-based semantic parsing and generation models across diverse linguistic contexts.

1.1.3 Low-Resource Language Support: The Case of Urdu

The field of NLP has made significant improvements in recent years, particularly for resource-rich languages like English. However, a considerable challenge remains in extending these advancements to low-resource languages, which lack the extensive annotated datasets and linguistic resources necessary for developing robust NLP systems. Urdu, an Indo-Aryan language with over 170 million speakers worldwide [Jafar and Jafar, 2022], represents a prime example of such a low-resource language in the context of semantic parsing and generation tasks (see Figure 1.1).

The development of semantic resources for Urdu is crucial not only for advancing NLP capabilities in this language but also for broadening our understanding of cross-linguistic semantic phenomena. Rich morphology (e.g., verbs in Urdu are inflected for tense, aspect, mood, gender, and number, such as پڑھا (parha - he read) vs. پڑھی (parhi - she read)), complex script (e.g., the Nastaliq script combines characters into ligatures, such as کتاب (kitab - book) and خوبصورت (khubsurat - beautiful)), and unique syntactic features (e.g., flexible word order: علی نے کتاب پڑھی (Ali ne kitab parhi - Ali read the book) vs. کتاب علی نے پڑھی (Kitab Ali ne parhi - The book was read by Ali), and the use of postpositions: علی کے ساتھ (Ali ke saath - with Ali)) of Urdu present both challenges and opportunities for semantic representation and processing [Butt and King, 2002].

Creating a semantically annotated corpus for Urdu is a foundational step toward enabling advanced NLP tasks such as semantic parsing and text generation. However, this process is labor-intensive and requires careful consideration of linguistic peculiarities. The scarcity of existing semantic resources for Urdu necessitates innovative approaches to resource development, potentially leveraging cross-lingual transfer techniques or adapting existing semantic formalisms to accommodate linguistic features. Moreover, the development of semantic parsing and generation models for Urdu raises questions about the applicability of neural architectures and training techniques that have been successful for rich-resource languages. Adapting these models to totally different linguistic characteristics while dealing with limited training data is a significant

challenge.

Data augmentation techniques, which have proven effective in other NLP tasks, offer a potential solution to the data scarcity problem [Shorten and Khoshgoftaar, 2019]. However, applying these techniques to Urdu requires careful consideration to ensure the generated data is linguistically valid and semantically coherent. Evaluating the performance of Urdu semantic processing systems presents another challenge, as it requires the development of appropriate evaluation metrics and test sets that can capture the nuances of Urdu semantics.

In light of these considerations, we investigate the following research questions:
RQ3.1: How can we create a high-quality, semantically annotated corpus for Urdu that is freely available for research purposes?
RQ3.2: What are the challenges and solutions in developing effective semantic parsing and generation models for Urdu?
RQ3.3: How can we adapt and apply sound semantic data augmentation approaches to Urdu, ensuring the generation of contextually similar and semantically correct data?
RQ3.4: To what extent does data augmentation enhance the generalization power of parsing and generation models for Urdu?
RQ3.5: How does the performance of Urdu semantic processing (parsing and generation) compare to that of other languages, and what insights can be gained from this comparison?

By addressing these questions, we aim to develop robust semantic processing capabilities for Urdu, contributing to the broader goal of making advanced NLP technologies accessible to low-resource languages. This research has the potential to not only benefit Urdu speakers but also to provide insights and methodologies that can be applied to other low-resource languages, ultimately working towards more linguistically inclusive NLP systems.

1.1.4 Improving Performance for Limited-Data Scenarios: Italian as a Case Study

While Italian is not typically classified as a low-resource language, it represents an interesting case study in the context of semantic parsing and text generation tasks, particularly those involving DRS. Despite being a widely spoken language with a rich linguistic tradition, Italian has limited annotated resources for advanced semantic tasks compared to rich resource languages like English [Wang et al., 2023a, Zhang et al., 2024]. This scenario presents a unique opportunity to explore techniques for improving model performance in situations where data is limited but not entirely scarce.

The challenge of limited data is prevalent in many languages and domains within NLP, making research in this area broadly applicable. For languages like Italian, which have some existing resources but not enough for optimal performance in complex semantic tasks, innovative approaches are needed to bridge the gap between low-resource and high-resource scenarios.

Cross-lingual techniques offer a promising avenue for addressing the limited-data challenge [Unanue et al., 2023, Wang et al., 2023a]. Leveraging resources from resource-rich languages, particularly those linguistically related to Italian, could potentially enhance the performance of Italian semantic processing models. The use of English Word-

Net, for instance, presents an intriguing possibility for augmenting Italian semantic datasets, given the shared Indo-European roots and considerable lexical similarities between English and Italian [Fancellu et al., 2019]. However, the application of cross-lingual techniques is not straightforward. It requires careful consideration of linguistic differences and the potential introduction of noise or inconsistencies in the augmented data. The challenge lies in developing methods that can effectively transfer semantic knowledge while respecting the unique linguistic features of Italian.

Furthermore, the effectiveness of such cross-lingual augmentation techniques may vary depending on the specific semantic phenomena being addressed. Italian-specific linguistic features, such as rich verbal morphology and flexible word order may present particular challenges or opportunities in the context of semantic parsing and generation. The scalability and generalizability of cross-lingual augmentation techniques are also important considerations. Insights gained from the Italian case study could potentially inform approaches for other languages with limited semantic resources, contributing to the broader goal of developing more linguistically diverse NLP systems.

In light of these considerations, we investigate the following research questions:
RQ4.1: How can we develop and implement a novel cross-lingual augmentation methodology that leverages English WordNet to enhance Italian semantic datasets?
RQ4.2: What is the effectiveness of this augmentation technique in improving performance scores for both DRS parsing and generation tasks in Italian?
RQ4.3: How does cross-lingual augmentation affect the handling of Italian-specific linguistic features in semantic processing?
RQ4.4: To what extent is this approach scalable and applicable to other low-resource languages in the domain of semantic NLP?

By addressing these questions, we aim to develop effective strategies for improving semantic parsing and generation performance in limited-data scenarios, using Italian as a case study. This research has the potential to not only enhance NLP capabilities for Italian but also to provide insights and methodologies that can be applied to other languages facing similar data limitations, ultimately contributing to more robust and linguistically diverse semantic processing systems.

1.1.5 Evaluating Structural and Linguistic Quality in DRS-based Parsing and Generation through Bidirectional Evaluation

Evaluating the accuracy and linguistic fidelity of DRS-based systems poses unique challenges due to the dual requirements of structural and semantic alignment. Traditional metrics like SMATCH are limited in their focus on structural similarity, which, while valuable, often overlooks crucial linguistic elements that contribute to semantic coherence. Similarly, surface-level generation metrics such as BLEU, METEOR, COMET, and BERTScore can penalize semantically equivalent yet syntactically diverse outputs, obscuring a model's true performance in capturing meaning [Kamp et al., 2010, Amin et al., 2024]. To overcome these limitations, we present two bidirectional methodologies—Parse-Generate (PARS-GEN) and Generate-Parse (GEN-PARS)—designed to bridge the gap between structural and linguistic evaluations by leveraging the reversible relationship between parsing and generation.

With PARS/PARS-GEN, our objective was to enhance parsing evaluation by analyzing the quality of text generated from DRS. This approach enables the identification of linguistic nuances, such as syntactic variety and semantic consistency, that purely structural metrics like SMATCH might miss. Through this methodology, we tried to capture linguistic aspects beyond those accessible by structure-only metrics, leading to a richer assessment of parsing accuracy.

Conversely, the GEN/GEN-PARS method evaluates text generation by parsing generated text back into DRS representations. This allows us to assess the semantic consistency and structural integrity of generated outputs in a way that surface-level text generation metrics, such as BLEU and METEOR, cannot fully achieve. By focusing on structural fidelity within semantic representations, we sought to complement surface-level evaluations with a deeper understanding of how well the generated text preserves the intended meaning.

A critical dimension of our work was to examine these evaluation methods across multiple languages—English, Italian, and Urdu. Recognizing that each language exhibits distinct syntactic and semantic properties, we aimed to understand how evaluation metrics behave in multilingual contexts. Our objective here was to uncover language-specific and universal evaluation patterns, providing insights into the applicability of structural and linguistic assessments across diverse linguistic landscapes. Our research also aimed to investigate the broader implications of structural accuracy in semantic parsing on the quality of generated text. By studying how parsing precision influences generation outcomes across different languages, we gained insights into the impact of structural fidelity on linguistic output, which is crucial for optimizing DRS-based systems in multilingual settings.

Finally, our work explores the reversible nature of semantic parsing and text generation to improve evaluations. By examining how parsing and generation outputs validate each other through this bidirectional framework, we aimed to harness the feedback loop potential of DRS-based tasks for more comprehensive assessments.

In this context, our research investigates several critical questions:
RQ5.1: How can we effectively evaluate semantic parsing and text generation beyond current structural and surface-level metrics?
RQ5.2: How does structural accuracy in semantic parsing influence linguistic quality in text generation?
RQ5.3: How do evaluation challenges and error patterns vary across different languages?
RQ5.4: Can the reversible nature of semantic parsing and text generation be exploited for improved evaluations?

By addressing these questions, we aim to advance DRS evaluation practices with frameworks that integrate both structural and linguistic dimensions, providing a more nuanced view of semantic parsing and generation quality across diverse languages. Our findings have practical implications, providing valuable tools for debugging and optimizing semantic processing systems and offering guidance for improving system performance across languages. These contributions underscore the importance of holistic evaluation strategies in advancing DRS-based semantic parsing and text generation systems.

1.1.6 Exploiting Task Reversibility of DRS Parsing and Generation: Challenges and Insights from Multi-lingual Perspective

Semantic parsing and text generation exhibit reversible properties when utilizing DRS [Wang et al., 2023a]. However, both processes—text-to-DRS parsing and DRS-to-text generation—are susceptible to errors [Amin et al., 2024]. In this work, we exploit the reversible nature of DRS to explore both error propagation, which is commonly seen in pipeline methods, and the less frequently studied potential for error correction.

First, we aim to examine the impact of DRS reversibility on error dynamics (propagation or mitigation), specifically analyzing whether errors introduced during parsing or generation can be propagated or mitigated in subsequent stages. This objective drives our exploration of two pipeline methods—Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG)—to better understand how these semantic tasks interact within a pipeline framework.

Our second objective is to utilize pre-trained language models in these DRS-based pipelines without additional model training. By implementing these pipelines on low-resource languages (Urdu and Italian) and a high-resource baseline (English), we seek to evaluate the feasibility of using existing language models to handle error dynamics across different languages. Our aim is to assess whether these models, when arranged in a pipeline, can naturally facilitate error correction or mitigation in semantic processing tasks. Moreover, we investigate the types and nature of errors that are either addressed or amplified by the PGP and GPG pipelines. Our detailed cross-linguistic error analysis allows us to identify which linguistic features—such as sentence structure, polarity, and length—are more resilient to errors in multilingual settings, revealing the strengths and limitations of these pipeline approaches.

Finally, our objective is to establish the capabilities and limitations of reversible pipeline methods within multilingual contexts. We place particular emphasis on the unique challenges faced by low-resource languages in semantic processing, comparing results across languages to provide a nuanced understanding of how resource availability affects DRS-based parsing and generation.

In light of these considerations, we pose the following research questions:
RQ6.1: How does the reversible nature of semantic parsing and text generation with DRS affect error propagation and correction across different languages?
RQ6.2: Can language models be effectively utilized in a pipeline approach to investigate error dynamics without additional model training?
RQ6.3: What are the performance changes achieved by the proposed reversible pipelines compared to baseline models across different languages?
RQ6.4: Which types of errors are more effectively addressed or amplified by the PGP and GPG pipelines in each language?
RQ6.5: What are the capabilities and limitations of the reversible pipeline approaches in different linguistic contexts?

By addressing these questions, through our comparative analysis, we aim to contribute valuable insights into the potential and boundaries of these pipeline approaches for enhancing DRS-based semantic tasks across diverse linguistic landscapes.

1.1.7 Additional Research Contributions to NLP

This research, conducted during my PhD, also explores several related areas that complement and extend our primary focus on semantic parsing and generation:

- We investigate the role of activation functions in neural Named Entity Recognition (NER) for large semantically annotated corpora, contributing to the broader understanding of neural network behavior in semantic tasks [Amin et al., 2022a].
- We develop an assistive data glove for recognizing isolated static postures in American Sign Language using neural networks, demonstrating the application of similar neural techniques to multimodal communication systems [Amin et al., 2023].
- We apply machine learning algorithms to distinguish discrete digital emotional fingerprints in web pages related to back pain, showcasing the potential of our semantic analysis techniques in health informatics and affective computing [Caldo et al., 2023].

These additional contributions highlight the versatility and broad applicability of the neural approaches developed in this thesis.

1.2 Thesis Structure

This dissertation comprises seven chapters, including the introduction (this chapter) and the conclusion (Chapter 7). After providing the background in Chapter 2, each subsequent chapter addresses specific aspects of computational semantics, focusing on innovative approaches for advancing multilingual DRS-based semantic parsing and generation through novel data transformation strategies (Chapter 3 and 4), robust evaluations (Chapter 5), and task reversibility (Chapter 6).

Specifically, Chapter 5 and Chapter 6 of this dissertation are the preliminary works that lay foundational insights and experimental frameworks to inform the later developments in reversible DRS semantic parsing and generation. Chapter 5 introduces bidirectional evaluation methodologies, and Chapter 6 explores the reversible nature of DRS parsing and generation through reversible pipeline approaches. Both chapters provide preliminary investigations that are critical to understanding and refining DRS-based parsing and generation tasks across different languages.

Chapter 2 - Background

This chapter lays the foundation for the dissertation by introducing essential concepts in computational semantics. We provide a comprehensive definition of DRS and describe the semantic formalisms integral to our research. Additionally, we review the neural network models central to this dissertation, including sequence-to-sequence models and pre-trained neural models. We also discuss various strategies aimed at enhancing the performance of these neural models.

Chapter 3 - Data Transformation Strategies for DRS Clause Format: Augmentation and Delexicalization for DRS-to-Text Generation

In this chapter, we explore innovative approaches to data augmentation and data delexicalization for DRS-based tasks. We investigate the feasibility of augmenting logical data representations like DRS while maintaining semantic coherence (RQ1.1-RQ1.8). Additionally, we examine data delexicalization techniques to enhance model generalization, addressing the challenges posed by the strong integration of lexical entities with external resources in DRS (RQ2.1-RQ2.5). This chapter aims to develop robust methods for improving model performance and generalizability in semantic parsing and generation tasks.

Chapter 4 - Improving Semantic Parsing and Text Generation through Multi-Faceted Multi-Lingual Data Augmentation: Working with Variable Free DRS

This chapter presents a comprehensive exploration of multi-faceted data augmentation techniques for improving semantic parsing and text generation across multiple languages, with a focus on variable-free DRS. We begin by detailing our novel multi-faceted data augmentation approach for English DRS. This includes named entity augmentation, WordNet-based lexical substitutions, and grammatical transformations through tense variations. We discuss how these techniques significantly expanded the PMB dataset while maintaining semantic correctness and contextual similarity. The chapter presents the substantial performance improvements achieved across various metrics for both semantic parsing and text generation tasks using transformer models like byT5, mT5, T5, and mBART.

Building on these insights, we then extend our focus to low-resource languages, presenting case studies on Urdu and Italian:

1. **Urdu Case Study:** We discuss the creation of a semantically annotated corpus for Urdu and the development of semantic parsing and generation models. This section explores how our augmentation techniques can be adapted to a low-resource language context, addressing the challenges and opportunities specific to diverse linguistic features for Urdu (addressing RQ3.1-RQ3.5).

2. **Italian Case Study:** For Italian, we explore cross-lingual augmentation techniques to enhance performance in a limited-data scenario. This section investigates how resources from resource-rich languages, particularly English, can be leveraged to improve Italian semantic processing models (addressing RQ4.1-RQ4.4).

Throughout the chapter, we highlight the challenges and solutions in adapting DRS-based approaches to diverse linguistic contexts. We discuss how the multi-faceted augmentation strategies developed for English can be modified and applied to other languages, considering language-specific features and resource constraints.

Chapter 5 - Evaluating Structural and Linguistic Quality in DRS-based Parsing and Generation through Bidirectional Evaluation

In this chapter, we develop novel evaluation frameworks to improve the assessment of DRS-based semantic parsing and text generation. Traditional metrics often miss nuanced structural and linguistic aspects, so we introduce two complementary approaches: Parse-Generate (PARS/PARS-GEN) and Generate-Parse (GEN/GEN-PARS). PARS/PARS-GEN assesses parsing by examining the linguistic quality of generated text, capturing syntactic and semantic features overlooked by metrics like SMATCH. Conversely, GEN/GEN-PARS evaluates generation by parsing outputs back into DRS to measure semantic consistency beyond surface metrics like BLEU. Spanning English, Italian, and Urdu, our research provides multilingual insights into evaluation patterns and the relationship between parsing accuracy and generation quality (RQ5.1-RQ5.4). These contributions advance DRS-based system evaluation, promoting holistic strategies to optimize multilingual semantic parsing and generation.

Chapter 6 - Exploiting Task Reversibility of DRS Parsing and Generation: Challenges and Insights from Multi-lingual Perspective

In this chapter, we have designed and evaluated reversible pipeline-based methodologies for DRS-based semantic parsing and text generation systems. By introducing the Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG) approaches, we aim to assess error dynamics in DRS reversibility, exploring whether errors in parsing or generation stages can be mitigated or amplified when processed in reverse. We tested these methods across English, Italian, and Urdu, using pre-trained language models without additional fine-tuning to see how effective the pipelines are for both high-resource and low-resource languages (RQ6.1-6.5). Our findings provide key insights into which linguistic features are more resilient, and also highlight the limitations and capabilities of these reversible pipeline methods in multilingual DRS-based semantic tasks, especially for languages with varying amounts of available resources. This comprehensive analysis gives us a better understanding of how to optimize DRS-based systems across diverse linguistic contexts.

Chapter 7 - Conclusions

The final chapter summarizes the key findings and contributions of our research. We revisit the research questions posed in the introduction, discussing how our work has addressed these challenges and advanced the field of computational semantics. Additionally, we reflect on the implications of our findings for future research and applications in advancing multilingual semantic parsing and generation.

Appendix - Additional Research Contributions

This section of the thesis presents three additional research projects in natural language processing that complement and extend the main focus of this thesis. The first project explores the role of activation functions in neural Named Entity Recognition, the second project sheds light on the development of an assistive data glove for American Sign Language recognition, and the third project applies machine learning to analyze

emotional content in health-related web pages. These studies demonstrate the broader applicability of the neural techniques developed in this thesis to diverse domains within computational linguistics and beyond.

1.3 Contributions

This thesis makes several notable contributions to the field of computational semantics:

1. We introduce novel techniques for data augmentation in logical data representations such as DRS. We demonstrate the feasibility of augmenting DRS through supersenses while maintaining semantic coherence and explore the impact of in-context and out-of-context vocabulary on model performance. Additionally, we investigate the behavior of LLMs in analyzing DRS structures **(Chapter 3)**.

2. We present effective methods for delexicalizing DRS representations, enhancing model generalization while preserving connections to external lexical resources. We explore the use of supersenses for nouns and analyze the impact of combining delexicalized and fully lexicalized data on model performance **(Chapter 3)**.

3. We develop and implement a comprehensive multi-faceted data augmentation approach for English DRS, encompassing named entity augmentation, WordNet-based lexical substitutions, and grammatical transformations. We provide empirical evidence of significant performance improvements in both semantic parsing and text generation tasks using various transformer models **(Chapter 4)**.

4. We develop the Urdu Meaning Bank, a pioneering semantically annotated corpus for Urdu. We demonstrate the effectiveness of semantic data augmentation techniques for Urdu, enhancing the generalization power of parsing and generation models for this low-resource language **(Chapter 4)**.

5. We propose a novel cross-lingual augmentation methodology that leverages English WordNet to enhance Italian semantic datasets. We provide empirical evidence of its effectiveness in improving performance for both DRS parsing and generation tasks in Italian, offering insights into its potential applications for other low-resource languages **(Chapter 4)**.

6. We introduce two bidirectional approaches: Parse-Generate (PARS/PARS-GEN) and Generate-Parse (GEN/GEN-PARS). PARS/PARS-GEN assesses parsing quality through generated text, addressing syntactic and semantic nuances beyond structural metrics like SMATCH. GEN/GEN-PARS evaluates text generation by re-parsing outputs into DRS, capturing semantic consistency beyond surface metrics like BLEU **(Chapter 5)**.

7. We developed reversible pipeline methodologies—Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG)—to analyze error dynamics in DRS-based parsing and generation. Additionally, we investigated the underlying causes of pipeline failures and emphasized the potential for establishing more standardized and balanced test sets for semantic processing **(Chapter 6)**.

8. We provide insights into the role of activation functions in neural Named Entity Recognition for large semantically annotated corpora, contributing to the optimization of neural models for semantic tasks **(Appendix: Additional Research Contributions-A)**.

9. We develop an innovative assistive data glove for American Sign Language recognition, demonstrating the application of neural techniques to multimodal communication systems **(Appendix: Additional Research Contributions-B)**.

10. We apply machine learning algorithms to analyze emotional content in health-related web pages, showcasing the potential of our semantic analysis techniques in health informatics and affective computing **(Appendix: Additional Research Contributions-C)**.

1.4 Research Publications

The research presented in this dissertation is based on the following original works:

Published

1. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2024. "Exploring Data Augmentation in Neural DRS-to-Text Generation". In Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2164–2178, St. Julian's, Malta. Association for Computational Linguistics. https://aclanthology.org/2024.eacl-long.132

2. Amin, M.S., Anselma, L., Mazzei, A. (2024). Improving DRS-to-Text Generation Through Delexicalization and Data Augmentation. In: Rapp, A., Di Caro, L., Meziane, F., Sugumaran, V. (eds) Natural Language Processing and Information Systems. NLDB 2024. Lecture Notes in Computer Science, vol 14762. Springer, Cham. https://doi.org/10.1007/978-3-031-70239-6_9

3. Amin, Muhammad Saad, A. Mazzei, and L. Anselma. "Towards data augmentation for drs-to-text generation." Proceedings of the Sixth Workshop on Natural Language for Artificial Intelligence (NL4AI 2022) co-located with 21th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2022), Udine, Italy. CEUR WORKSHOP PROCEEDINGS, vol. 3287, pp. 141-152. CEUR-WS, 2022. https://ceur-ws.org/Vol-3287/paper14.pdf

4. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei, "Data Augmentation for Low-Resource Italian NLP: Enhancing Semantic Processing with DRS", in Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), Pisa, Italy, December 4-6, 2024 (F. Dell'Orletta, A. Lenci, S. Montemagni, and R. Sprugnoli, eds.), vol. 3878 of CEUR Workshop Proceedings, CEUR-WS.org, 2024. https://ceur-ws.org/Vol-3878/5_main_long.pdf

5. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2025. Evaluating Structural and Linguistic Quality in Urdu DRS Parsing and Generation through Bidirectional Evaluation. In Proceedings of the First Workshop on Natural Language Processing for Indo-Aryan and Dravidian Languages, pages 33–43, Abu Dhabi. Association for Computational Linguistics. https://aclanthology.org/2025.indonlp-1.4.pdf

6. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2025. Exploiting Task Reversibility of DRS Parsing and Generation: Challenges and Insights from a Multi-lingual Perspective. In Proceedings of the First Workshop on Language Models for Low-Resource Languages, pages 268–286, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. https://aclanthology.org/2025.loreslm-1.22.pdf

7. M. S. Amin, L. Anselma and A. Mazzei, "The Role of Activation Function in Neural NER for a Large Semantically Annotated Corpus" 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 2022, pp. 1-6, https://doi.org/10.1109/ETECTE55893.2022.10007317

8. Amin, M.S.; Rizvi, S.T.H.; Mazzei, A.; Anselma, L. Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network. Electronics 2023, 12, 1904. https://doi.org/10.3390/electronics12081904

9. Davide Caldo, Silvia Bologna, Luana Conte, Muhammad Saad Amin, Luca Anselma, Valerio Basile, Md. Murad Hossain, Alessandro Mazzei, Paolo Heritier, Riccardo Ferracini, Elizaveta Kon & Giorgio De Nunzio. Machine learning algorithms distinguish discrete digital emotional fingerprints for web pages related to back pain. Sci Rep 13, 4654 (2023). https://doi.org/10.1038/s41598-023-31741-2

Under review

1. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei, "Improving Semantic Parsing and Text Generation through Multi-Faceted Data Augmentation", submitted to IEEE Access.

2. Muhammad Saad Amin, Xiao Zhang, Luca Anselma, Alessandro Mazzei, and Johan Bos, "Semantic Processing for Urdu: Corpus Creation, Parsing, and Generation", Submitted to Language Resources and Evaluation.

Chapter 2

Background

This chapter examines various approaches to meaning representation, with a primary focus on Discourse Representation Structures (DRSs). Particular attention is given to the diverse graphical representations of DRSs, including Box, Clause, and Variable-free formats, as well as the assessment of language models. Additionally, the chapter provides a comparative analysis of DRSs in relation to alternative formalisms and offers an in-depth discussion on the application of sequence-to-sequence and pre-trained language models in this context.

2.1 Meaning Representations

Structured representations of linguistic meaning serve to encapsulate the key components of events, actions, times, locations, causes, and manners. These representations can take various forms, such as Discourse Representation Theory (DRT) [Kamp and Reyle, 1993], Abstract Meaning Representation (AMR) [Shou et al., 2022, Bevilacqua et al., 2021], Minimal Recursion Semantics (MRS) [Horvat et al., 2015], and BabelNet Meaning Representation [Lorenzo et al., 2022]. These frameworks act as intermediaries between the complexities of language and our non-linguistic comprehension of reality. They can be viewed as organized formats that distill the core elements of linguistic input, operating under the premise that all language structures convey information about the world. These meaning representations have practical applications in diverse fields, including information extraction [Rao et al., 2017, Solawetz and Larson, 2021], machine translation [Song et al., 2019, Li and Flanigan, 2022], sentiment analysis [Marasović and Frank, 2018], and various other domains [Pan et al., 2015, Kapanipathi et al., 2020].

This work concentrates on DRSs, which are formal semantic representations rooted in DRT. DRSs are recursive logical frameworks that incorporate discourse referents and represent the relationships between them. Compared to other representational systems i.e., AMR (see section 2.1.4), DRSs offer enhanced semantic expressiveness and encompass a broad spectrum of linguistic phenomena, such as quantification, negation, reference resolution, comparison, discourse relations, and presuppositions. While the traditional box format of DRS provides an intuitive visual representation, it is less con-

ducive to computational modeling. Consequently, DRS is often converted into formats that are more flexible for processing by current neural networks. This research primarily explores the clause-format DRS [van Noord et al., 2018] and the variable-free DRS [Bos, 2023], both of which are available in the Parallel Meaning Bank (PMB) [Abzianidze et al., 2017]. The subsequent section will provide a comprehensive examination of these formalisms.

2.1.1 Discourse Representation Structures

Discourse Representation Structures (DRSs) are meaning representations derived from Discourse Representation Theory (DRT). DRT is a formal semantic framework extensively studied in linguistic semantics, particularly suited for compositional semantics. It was developed to address semantic and pragmatic issues related to anaphora and tense [Kamp, 1981, Kamp and Reyle, 1993]. DRSs, the expressive units in DRT, possess a recursive structure typically depicted as boxes. This discussion will focus on the specific DRS formats used in the Parallel Meaning Bank (PMB) throughout this work.

A fundamental DRS box comprises two elements: discourse referents and conditions. The upper section contains discourse referents, while the lower section lists atomic or compound conditions associated with these referents. Discourse referents, also termed variables, denote discourse elements such as persons or events. Conditions, which include concepts, roles, constants, and comparison operators, assert information about these discourse elements.

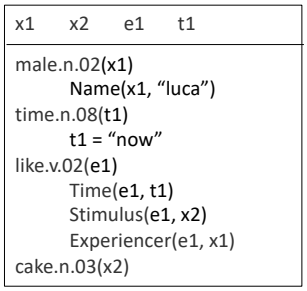


Figure 2.1: Graphical representation of DRS in the Box format for the text "Luca likes cake."

For instance, in a DRS listed in Figure 2.1, discourse referents might represent a male person `x1` and a cake `x2`, with conditions specifying their attributes and interactions e.g., `x1` is a male person named Luca, `x2` is a cake, and `x1` likes `x2`. Concepts are represented using WordNet synsets [Fellbaum, 1998], consisting of lemma, part-of-speech, and sense number e.g., `male.n.02` and `cake.n.03`. These synsets cover nouns, verbs, adjectives, and adverbs. Events in DRSs have their own discourse referents `e1`, with associated concepts represented by verbal WordNet synsets e.g., `like.v.02`.

The main verb in a sentence typically introduces the tense, which is consistently represented as a time period `t1` and associated with the WordNet synset `time.n.08`. To illustrate the roles of participants in events, VerbNet [Kipper et al., 2006] employs

two-place predicates. For instance, the verb "like" introduces the role, signifying action upon `cake`, while `Time` denotes the tense of the verb. In this context, DRSs adopt a neo-Davidsonian approach [Bos, 2008], which simplifies the representation of events by associating each event with its participants and other properties (such as `Time`) through separate relations. This allows each argument (e.g., `Agent`, `Theme`) to be independently linked to the event, making the representation more modular and flexible, particularly in handling events with multiple participants. For example, in a neo-Davidsonian framework, an event like "Luca likes cake." would be decomposed into multiple relations such as $\texttt{like}(e_1), \texttt{Agent}(e_1, \text{Luca}), \texttt{Theme}(e_1, \text{cake})$ clearly distinguishing between the event and its roles. DRSs also incorporate important constants, representing elements such as discourse direction ("speaker", "hearer"), interrogatives ("?"), proper nouns ("Luca"), numerical values ("7"), and temporal references ("now") [van Noord et al., 2020]. To relate and compare discourse referents, comparison operators are utilized. These operators can involve variables or constants, such as `t1 = now` (temporally precedes) in Figure 2.1, which expresses the present tense. Past tense is indicated when $\texttt{t1} < \texttt{now}$, signifying that the event in the DRS occurs in the past time frame.

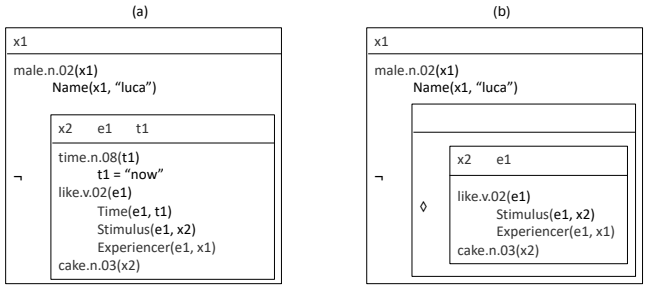


Figure 2.2: Modeling (a): negation "Luca does not like cake." and (b): possibility "Luca never likes cake." expressions for the text "Luca likes cake." in the nested box format of the DRS.

A complete DRS is typically viewed as a set of interconnected boxes, potentially nested within each other as displayed in Figure 2.2. When a basic box is nested within another, the lower part of the encompassing DRS box is termed a complex condition. These complex conditions indicate logical relations between sets of conditions and represent their scope. The formal syntax of DRS and DRS-conditions can be defined as follows as stated in [Bos et al., 2017, Abzianidze et al., 2020]:

DRS Syntax: A DRS is defined by a set of discourse referents U and a set of DRS-conditions C, collectively denoted as $\langle U, C \rangle$. When two DRSs, B and B', are related by a discourse relation D, the resulting structure $D(B, B')$ also qualifies as a DRS. No other constructions meet the criteria to be considered a DRS.

DRS-Condition Syntax: A DRS-condition can take the following forms as stated in

[Bos et al., 2017, Abzianidze et al., 2020]: $P(x)$, where P is a concept and x is a first-order variable; xOy, where x is a first-order variable, y is either a first-order variable or a constant, and O is a comparison operator; and $R(x, y)$, where R is a role and both x and y are first-order variables. Additionally, if B is a DRS, then the structures $\neg B$, $\Box B$, and $\diamond B$ are also considered DRS-conditions. No other constructions qualify as a DRS condition.

In logical representation, the operators $\neg$, $\Box$, and $\diamond$ are primarily used in modal logic, which extends classical logic to include the concepts of necessity and possibility. These operators are incorporated into DRS to handle more complex expressions of modality and negation within discourse contexts. The $\neg$ operator negates a DRS. In logical terms, $\neg$ means that the conditions specified within the DRS do not hold. For example, if a DRS represents "Luca likes cake," $\neg$ would represent "Luca does not like cake." The $\Box$ operator is used to express necessity. In the context of DRS, $\Box$ would represent that the conditions within DRS necessarily hold in all possible worlds or contexts. For example, $\Box$ might express "It is necessarily true that Luca likes cake," meaning that in every possible scenario, this holds true. The $\diamond$ operator expresses possibility and indicates that the conditions in DRS are possible in at least one context or possible world. Its expressive power lies in capturing modal information about what could be true. For instance, $\diamond$ could represent "It is possible that Luca likes cake," meaning there is at least one possible world or scenario where this is true.

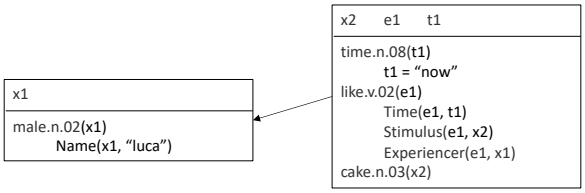


Figure 2.3: Modeling presupposition in a separate DRS box for the text "Luca likes cake."

Connected boxes can model presuppositions, represented in separate boxes outside the main DRS box [Van der Sandt, 1992]. Figure 2.3 illustrates the representation of an affirmative statement: "Luca likes cake." While the semantic contents in Figures 2.1 and Figure 2.3 are equivalent, they differ in that the latter models the presupposition in a separate box. This structure allows for the representation of complex semantic relationships, including negation, possibility, and presupposition.

Consider the representation of Luca in Figure 2.2(a), where the conditions are placed outside the negation scope. This arrangement implies the existence of a male named Luca, regardless of whether he likes cake or not. Furthermore, Figure 2.2(b) demonstrates a DRS incorporating a possibility expression. This can be interpreted as stating that there exists a male individual named Luca for whom participation in a liking event is not possible.

The current DRS format is capable of handling multi-sentence documents by employing explicit discourse relations to link various, potentially nested, boxes. Each box

is assigned a unique identifier (e.g., b1, b2), enabling the indication of inter-box relationships. These connections are influenced by rhetorical relations found in Segmented DRT, as proposed by [Asher, 2012] and further developed by [Asher and Lascarides, 2003].

To illustrate, consider the example "Luca did not like cake. He was rude." The discourse relation for this example is CONTINUATION (as shown in Figure 2.4). Sentences separated by periods or commas are represented by individual boxes. Other common discourse relations include CONTRAST (e.g., "The movie was long, but it was entertaining.") and CONSEQUENCE (e.g., "It started raining, so we canceled the picnic.").

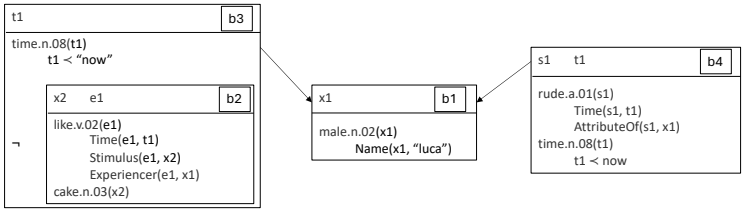


Figure 2.4: Modeling DRS having multi-sentence representation by using explicit discourse relations "CONTINUATION" to connect the text "Luca did not like cake. He was rude."

DRS is a highly expressive formalism compared to other meaning representations. It explicitly models scope i.e., negation, quantification, and presuppositions, and can manage multi-sentence documents. No other semantic representation has previously demonstrated the capacity to encompass as many semantic phenomena as DRS does while also having available annotated corpora. The DRS data used in this work is sourced from the Parallel Meaning Bank (PMB) [Abzianidze et al., 2017], which will be described in subsequent sections.

Parallel Meaning Bank: The Parallel Meaning Bank (PMB) [Abzianidze et al., 2017] is a semantically annotated parallel corpus with English as the pivot language. Each document in the PMB contains English text along with various DRS representation formats. It also includes translations of the English text into languages such as Italian, German, Dutch, Chinese, and Japanese. This research contributes to the expanding spectrum of DRS semantic representations by introducing the Urdu Meaning Bank (UMB), a pioneering semantic resource for the Urdu language. The UMB represents a significant addition to the Parallel Meaning Bank (PMB), further diversifying its linguistic coverage and enhancing its cross-lingual capabilities. The corpus is developed through cross-lingual projection, where automatically generated (and manually corrected) semantic annotations for English sentences are mapped onto their word-aligned translations, assuming meaning preservation. The PMB aims to create a language-neutral final DRS for each text, integrating various semantic phenomena within a single formalism.

The PMB incorporates several layers of semantic annotation, including:

1. **Text Segmentation:** This step involves identifying word and sentence boundaries, treating multiword expressions as single tokens, and decomposing transparent compound words to assign atomic meanings to tokens. The segmentation follows an IOB (Inside-Outside-Beginning) annotation scheme at the character level, using the Elephant tokenizer [Evang et al., 2013] with language-specific models.

2. **Syntactic Parsing using CCG:** At this step, syntactic analysis is performed using Combinatory Categorial Grammar (CCG) [Bos et al., 2004], a lexicalized grammar theory that facilitates cross-lingual projection of grammatical information. A modified version of EasyCCG [Lewis and Steedman, 2014] is used to adapt to the PMB's specific needs, ensuring accurate and compositional syntactic derivations across multiple languages.

3. **Universal Semantic Tagging:** This PMB layer employs a language-neutral tagset to assign semantic tags (semtags) to tokens. These tags generalize over parts of speech and named entities and include specific semantic information, such as negation triggers and modal expressions. A semantic tagger based on deep residual networks is used to achieve the best performance of annotation.

4. **Symbolization:** The process of symbolization converts words into symbols that represent their meanings, combining lemmatization and normalization. Lexical disambiguation is performed, mapping different word forms or synonyms to a common symbol. Machine learning-based models, along with external knowledge sources like WordNet, are used for symbolization.

5. **Compositional Semantic Analysis based on DRT:** This layer of PMB uses DRT for semantic interpretation. It constructs DRSs in a recursive, compositional way. The Boxer system, adapted to handle universal semtags, generates a language-neutral semantic representation for each sentence [Abzianidze et al., 2017].

These layers employ statistical models trained semi-supervisedly. The outputs from these layers are then processed by the rule-based semantic parser Boxer [Bos, 2008] to produce the final DRS. The goal is to create annotations that capture the most probable interpretation of a sentence, using language-neutral annotation models without ambiguities or under-specification techniques.

The PMB utilizes "Bits of Wisdom" [Bos et al., 2017] to allow human annotators to correct machine-generated output. These corrections focus on intermediate layers rather than the final meaning representation, serving to improve model training and create a gold standard annotated subset. Annotation quality is categorized into three levels for each layer and language: bronze (fully automatic), silver (automatic with some manual corrections), and gold (fully manually checked and corrected).

The PMB has undergone multiple releases. This work specifically uses PMB release 3.0.0 and PMB release 5.0.0. Detailed descriptions of these datasets will be provided in the coming respective chapters.

PMB vs. GMB: The Groningen Meaning Bank (GMB) [Bos et al., 2017] and the PMB

both contribute to the field of semantic annotation through DRT, but they differ significantly in scope and annotation methods. The GMB, developed first, is primarily a semantically annotated corpus of English texts, providing rich syntactic and semantic representations within a single language framework. Its annotation focuses on detailed syntactic structures and employs CCG for syntactic parsing, emphasizing the complexity of English semantics. In contrast, the PMB was developed later with a focus on multilingualism, aligning English texts with their translations in various languages, including Italian, German, Dutch, Chinese, Japanese, and now Urdu as well. The PMB's annotation process is designed for cross-linguistic compatibility, creating language-neutral annotations that support the projection of semantic structures across languages. This involves multiple layers of semantic annotation, including text segmentation, universal semantic tagging, and symbolization, which aim to ensure compositionality and meaning preservation across translations. These differences highlight the emphasis of GMB on detailed analysis of English semantics and the aim of PMB to foster multilingual semantic understanding through diverse annotation techniques.

2.1.2 Clause-format DRS

While DRSs are typically presented in a recursive box format that is intuitive and reader-friendly, this format poses challenges for machine learning models due to its inherently hierarchical and nested structure. Machine learning models, especially sequence-based models like RNNs or transformers, are designed to process linear sequences of tokens. However, the box format in DRS represents a non-linear, hierarchical organization of discourse referents and conditions. This discrepancy between the structured representation of DRS and the linear processing nature of most machine learning models makes it difficult to encode and interpret the complex relationships between different elements in the DRS. For example, in a DRS representing a sentence with multiple clauses, such as "If John leaves, Mary will stay," the DRS would involve multiple layers of nested conditions: one representing the condition of John leaving, and another representing Mary's action contingent on the first condition. This requires the machine learning model to not only understand each condition independently but also track their interdependencies across different levels of the box hierarchy. Linear models often struggle to capture this kind of nested, hierarchical information without significant pre-processing or flattening of the DRS structure, which can result in loss of information or reduced accuracy in downstream tasks.

To cope with these challenges, DRSs are often converted into a clause format, which is more conducive to machine learning due to its simple, flat structure and its facilitation of partial DRS matching, which is valuable for evaluation purposes [van Noord et al., 2018]. This clause format represents DRSs as a linear set of clauses, with each clause serving as the smallest unit to represent discourse referents, DRS conditions, and discourse structure. The conversion process involves applying labels to DRSs as introduced by [Venhuizen, 2015] and [Venhuizen et al., 2018], and deconstructing the recursive DRS structure by labeling elements with the DRS in which they appear.

Figure 2.5 illustrates examples of DRSs in both box notation (top) and their corresponding clause format (bottom). In the clause format, each distinct variable is represented by a unique variable name, and vice versa. This necessitates the assignment

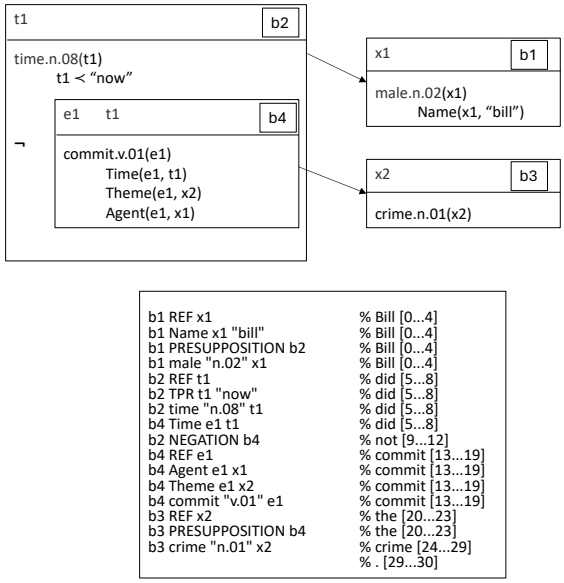


Figure 2.5: Graphical representation of the DRS box (top) and clause (bottom) format for the text "Bill did not commit the crime."

of distinct variable names to different discourse referents in a DRS before its transformation into clause form. To enhance the readability of these semantic representations, specific letters are assigned to variables: 'x', 'e', 's', and 't' are used for discourse referents denoting individuals, events, states, and time respectively, while 'b' is used for variables representing DRS boxes. This approach not only facilitates the reconstruction of the original box notation from the clause form but also significantly simplifies the process of matching clause forms for evaluation purposes [van Noord et al., 2018].

The DRS depicted in Figure 2.5 employs a box-format structure comprising three primary boxes (labeled b1, b2, and b4) and one presuppositional box (b3). Notably, b2 incorporates negation through a condition ¬b4, which contains a nested box b4. Within b4, conditions are expressed as unary and binary relations over discourse referents, which are introduced by b4, b2, and the presuppositional DRS (either b1 or b3).

The transformation between box notation and clause form is designed to be straightforward and reversible. In the clause form, each element — a discourse referent, condition, or discourse relation — is prefixed with the label of its originating box. It is worth noting that the letters used for variables in these semantic representations are assigned automatically, primarily to enhance readability.

The presence of logical operators is significant, as each introduces its own scope. Consequently, the number of these operators provides a baseline indication of the scope count within a given meaning representation. In addition to logical operators, scopes can also be introduced by presupposition triggers, such as proper nouns or pronouns.

DRS, as a scoped meaning representation, integrates word senses, thematic roles, and operators to form the final product of the semantically annotated corpus. While the clause-format DRS is a flattened version of the standard box notation, the explicit scopes (boxes) and scopal operators like negation contribute to the complexity of the clauses. DRS clauses, varying in length from three to four, can incorporate three variables and contain two types of variables for scopes and discourse referents. The five types of DRS clauses are defined as follows:

Definition of DRS Clauses:

1. A referent clause is a DRS clause denoted as "`b1 REF x1`", where $b1$ is a label for a DRS and $x1$ is a discourse referent. e.g., "Saad is a student.", here `b1`is the label for the DRS and `x1` is the discourse referent corresponding to "Saad." The referent clause introduces Saad as a referent in the discourse.

2. When $b1$ is a label for a DRS, $x1$ is a discourse referent, and P is a concept, the resulting DRS clause "`b1 P x1`" is termed a concept clause. e.g., considering the same example, if we have "`b1 student x1`", in this case, `b1` is the DRS label, `student` is the concept, and `x1` is the referent for "Saad." This clause identifies Saad as belonging to the concept of "student."

3. If $b1$ is a label for a DRS, $x1$ is a discourse referent, t is a term, and O is a comparison operator, the DRS clause can be written as "`b O x t`" or "`b O t x`", referred to as a comparison clause e.g., "John is older than 20." Here, `b1` is the DRS label, `x1` is the referent for "John," > is the comparison operator, and 20 is the term. The comparison clause asserts that John's age is greater than 20 following the DRS clause `b1 > x1 20`.

4. Similarly, for b as a DRS label, x as a discourse referent, t as a term, and R as a role, the DRS clause "`b R x t`" or "`b R t x`" is identified as a role clause. For example, "John gave Mary a book." with the DRS clauses `b1 Agent x1 x2, b1 Theme x1 x3`. In this case, `b1` is the DRS label, `x1` refers to the event (the action of giving), `x2` is the referent for "John" (`the agent`), and `x3` is the referent for "the book" (`the Theme`). These clauses define the roles of "John" and "the book" in the event.

5. Additionally, a discourse clause "`b D b`"' is formed when b and b' are labels for DRSs and D is a discourse relation. Considering the example "John left, and Mary stayed." with the DRS clause `b1 AND b2`, in this example, `b1` and `b2` are labels for the DRSs representing the events "John left" and "Mary stayed." The discourse relation `AND` connects the two DRSs to represent the logical conjunction of the two actions.

No other constructions qualify as DRS clauses.

Figure 2.5 illustrates the clause-format DRS, adhering to this formal definition. For instance, tense-related information is encoded with three additional clauses expressing a WordNet concept, semantic role, and comparison operator. While tree-format and

graph-format DRSs were proposed by [Liu et al., 2018] and [Fancellu et al., 2019] respectively, their complex conversion processes and reduced comprehensibility have limited their adoption. Consequently, this dissertation will focus on working with clause-format DRS in Chapter 3.

2.1.3 Variable-free DRS

Variable-free DRS, also known as Simplified Box Notation (SBN), is a recently proposed format of DRS by [Bos, 2023]. This representational variant of clause-format DRS eliminates explicit variables from the meaning representation. Unlike clause-format DRS, which uses variables to distinguish discourse references and depict their relationships, SBN employs physical distance between discourse references to represent their connections [De Bruijn, 1972]. This physical distance is defined by indices (e.g., +1, -2) that can bind in both forward and backward directions, encompassing discourse structure within its scope, as illustrated in Figure 2.6.

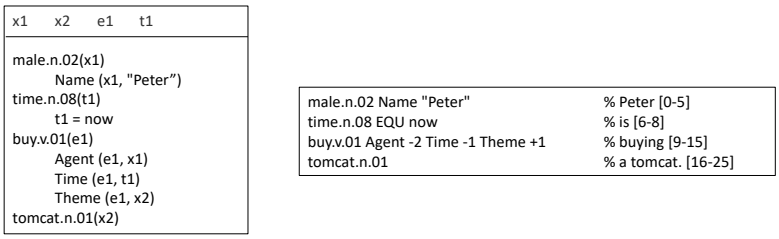


Figure 2.6: Graphical representation of the DRS box (top) and SBN (bottom) format for the text "Peter is buying a tomcat."

A variable-free DRS consists of sequences of concepts (one-place predicates), relations (two-place predicates), or structural constraints [Bos, 2023]. Concepts serve a dual role: introducing discourse referents and establishing their type with a conceptual one-place predicate. In practice, concepts are represented by WordNet and formally ordered based on the word order in the phrase, sentence, or text being analyzed. This alignment between natural language and DRS is crucial for manual annotation and machine learning techniques in natural language processing.

Relations, referred to as "roles", begin with an uppercase character and connect two conceptual predicates. They can be ordered with predicates first, followed by the relation instantiated with negative indices. This ordering constraint can be addressed by reversing roles: for any role R, R(X, Y) and ROf(Y,X) are equivalent. Roles without the "Of" suffix are interpreted as "event X has Y as R", while roles with the "Of" suffix represent inverted roles, rephrased as "Y is R of X" For example, in Figure 2.6, the variable-free DRS can be interpreted as "the buying event has tomcat as Theme" or "the tomcat is the Theme of buying event".

Figure 2.6 illustrates the translation from box-format DRS to variable-free DRS, expressed as a basic SBN sequence of concepts and relations without structural constraints.

For complex SBNs, structural constraints provide information about discourse structure or relations. In DRT, these constraints are expressed using logical operators, making DRS recursive. SBN introduces explicit discourse structure markers, segmenting the sequence of concepts and roles into distinct discourse units [Bos, 2023]. These markers are capitalized to differentiate them from concepts and relations.

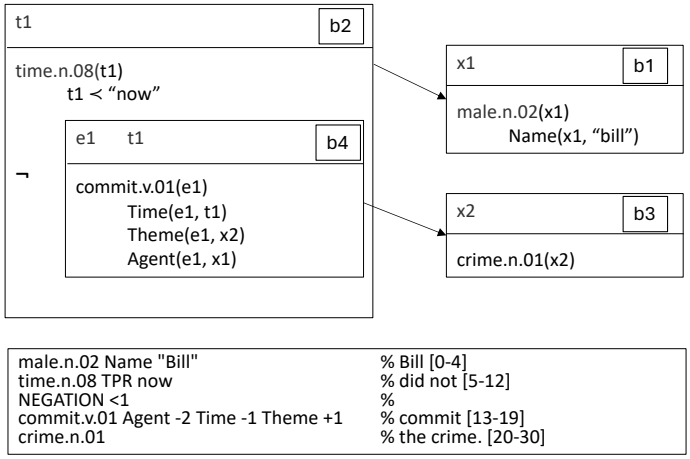


Figure 2.7: Graphical representation of the DRS box (top) and SBN (bottom) format for the text "Bill did not commit the crime."

Figure 2.7 demonstrates SBN with negation, where "`NEGATION`" serves as a marker dividing the meaning into two segments: the sequence before and after the marker. Generally, n markers within an SBN representation result in n+1 separate discourse units. Compared to clause-format DRS, variable-free DRS significantly reduces the sequence length of semantic representations. This development enhances the alignment between meaning representations and sentences, thereby optimizing the process through which machine learning models acquire semantic representations for extended textual content. Such optimization facilitates more efficient and accurate semantic parsing of longer texts, potentially advancing the field of natural language understanding. Simpler representations enhance readability and understanding, providing further insight into the expressive power of logic languages and simplifying manual annotation or correction of DRSs.

A variable-free meaning representation is also more easily transformed into a directed acyclic graph (DAG) structure (as shown in Figure 2.8) and integrated into the syntax-semantics interface. This is particularly useful for converting syntactic parser output into formal meaning representations [Poelman et al., 2022]. In semantic parsing evaluation, this transformation into a DAG format plays a crucial role, particularly when using evaluation tools like SMATCH [Cai and Knight, 2013]. SMATCH operates by aligning and comparing the model-generated logical representation with a reference logical representation. These representations, once transformed into DAGs, allow for an efficient matching process based on triplet overlaps.

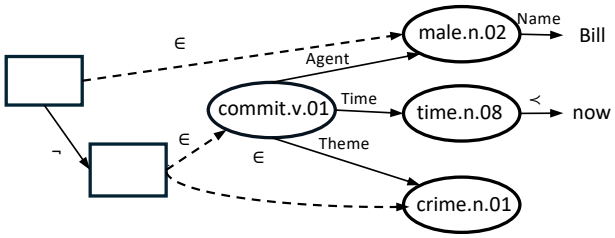


Figure 2.8: SBN in graphical notation for text "Bill did not commit the crime." corresponding to Figure 2.7.

Each triplet in the DAG structure consists of a relation between two nodes (`Entities` or `Concepts`), capturing the semantic relationships in the DRS. SMATCH calculates the degree of overlap between triplets in the system output and the reference, using precision, recall, and F1-score metrics. The F1-score, which balances precision and recall, reflects the accuracy of the model's parsing performance. By mapping the semantic elements in both the generated and reference graphs, SMATCH identifies corresponding triplets, focusing on their correct alignment. This triplet-matching mechanism in DAG format provides a structured, interpretable way to evaluate semantic parsing, ensuring that the model's output not only aligns syntactically but also captures the correct underlying semantics of the discourse. The ability to transform a DRS into a DAG structure facilitates the evaluation by making it easier to detect and quantify semantic similarities and discrepancies between the model output and the reference through triplet overlaps. We have worked extensively with SBN in Chapter 4 of this thesis.

2.1.4 DRS vs. Alternative Formalisms

This section provides a comparative discussion of various semantically annotated corpora and alternative meaning representation formalisms including Abstract Meaning Representation (AMR) [Banarescu et al., 2013], Minimal Recursion Semantics (MRS), and BabelNet Meaning Representation (BMR).

DRS vs. AMR

AMR, illustrated in Figure 2.9, is a popular semantic formalism that differs from DRS variants in two main categories: surface representation format and intrinsic expressive power. In terms of surface representation, DRS uses WordNet and VerbNet for concepts and roles, while AMR relies on PropBank [Palmer et al., 2005] for verbs and lacks explicit grounding for nouns. AMR also incorporates wikification [Cucerzan, 2007] for named entities, a feature absent in DRS.

Regarding intrinsic expressive power, DRS is generally more expressive than AMR. It can model a wider range of semantic phenomena, including quantification, negation,

Figure 2.9: Three equivalent AMR representations of the text "Bill did not commit the crime." (a) graph format, (b) string format, and (c) triple format. The corresponding DRS representations are shown in Figure 2.7 and Figure 2.8.

comparison, presupposition, and discourse relations. DRS can handle multi-sentence documents through explicit discourse relations, whereas AMR is primarily designed for sentence-level representation, with limited multi-sentence capabilities [O'Gorman et al., 2018]. Additionally, DRS explicitly models tense, a feature not present in AMR [van Noord, 2021].

While clause-format DRS resembles the triple notation of AMR, DRS clauses are more complex due to the inclusion of scope. DRS clauses vary in arity (typically 3-4) and use two types of variables (for scopes and discourse referents), whereas AMR triples have a fixed length and use only one variable type. Clause-format DRS is generally about twice as extensive as AMR for representing the same semantic content [van Noord, 2021].

DRS vs. MRS

MRS is a descriptive language for first-order object language formulas with generalized quantifiers. It facilitates the expression of grammatical constraints governing lexical and phrasal semantics, including principles of semantic composition [Copestake et al., 2005]. MRS uses underspecified representations comprising elementary predications and handles scoping constraints [Niehren and Thater, 2003]. Primarily it is focused on representing natural language semantics with minimal recursion, allowing it to manage scope ambiguity and underspecified meanings efficiently. MRS is well-suited for dealing with complex linguistic phenomena like quantifier scope ambiguity. For example, in a sentence like "Every student read a book," MRS can represent the ambiguity of whether every student read the same book or different books by leaving the scope of the quantifiers underspecified.

In contrast to DRS, which focuses on a logical structure to represent semantic information, including quantifier scopes and relations, MRS adopts a graphical structure to represent semantic information in a lightweight manner using features and constraints. DRS provides a clear way to handle anaphora resolution, where entities referred to in

one part of a discourse are linked to later references. For example, in the DRS for the sentence "John arrived. He was tired," the pronoun "He" is explicitly linked to "John" through discourse referents. This makes DRS more suitable for tasks requiring fine-grained semantic understanding and explicit logical relations. Additionally, while MRS is often integrated with syntactic parsing systems to create underspecified semantic representations, it doesn't provide the same level of detail for logical and discourse relations as DRS. DRS, with its formal structure, allows for deeper inferences, making it a better fit for applications that need to represent and process the underlying logic of language rather than just semantic constraints.

DRS vs. BMR

BMR is an interlingual formalism created to surpass language-specific limitations [Lorenzo et al., 2022]. It utilizes the extensive multilingual semantic resources offered by BabelNet [Navigli and Ponzetto, 2010] and VerbAtlas [Di Fabio et al., 2019]. This emerging language-independent semantic formalism abstracts entirely from syntax, providing a lexical-semantic representation that supports integration across various languages. For example, in BMR, the word "apple" in English, "mela" in Italian, and "manzana" in Spanish would all be linked to the same concept in BabelNet, facilitating multilingual tasks like machine translation or cross-lingual information retrieval. Additionally, Universal Conceptual Cognitive Annotation (UCCA) [Abend and Rappoport, 2013] has been developed as a cross-lingual annotation formalism that links words in a sentence through language-independent semantic relations. This graph-based semantic formalism, akin to AMR, functions independently of syntax and can be viewed as a multi-layered formalism, with each layer defining the relations it encodes.

Despite its strength in multilingual representation, BMR's abstraction from syntax makes it less useful for detailed discourse-level reasoning. BMR is excellent for linking words across languages but doesn't offer the explicit logical structure that DRS provides, which is crucial for tasks that require logical form generation, such as natural language inference, discourse interpretation, or coreference resolution. For example, in BMR, the sentence "John ate an apple" would be linked to its semantic concepts (`John`, `eat`, `apple`), but BMR would not explicitly capture the logical relations between these elements (e.g., who performed the action and what was consumed). DRS, on the other hand, would provide a detailed logical structure that identifies `John` as the `Agent` and the `apple` as the `Object`, facilitating a deeper semantic understanding. Furthermore, there are other formalisms like Universal Meaning Representation (UMR) [Van Gysel et al., 2021], which are currently being developed.

2.2 Neural Models

This section provides a brief description of Artificial Neural Networks (ANN) used throughout the dissertation. Neural networks, inspired by brain structure and function, use numerous artificial neuron connections to model complex input-output relationships or explore data patterns [Hopfield, 1982]. They automatically transform specific inputs into outputs using learnable functions and weights, trained on example data.

Among early neural models, the Feed-Forward Neural Network (FFNN) [Bebis and Georgiopoulos, 1994, Jain et al., 1996] is prominent. It consists of multiple neuron layers, where each neuron receives signals from the previous layer and propagates them to the next. Figure 2.10 illustrates this network structure.

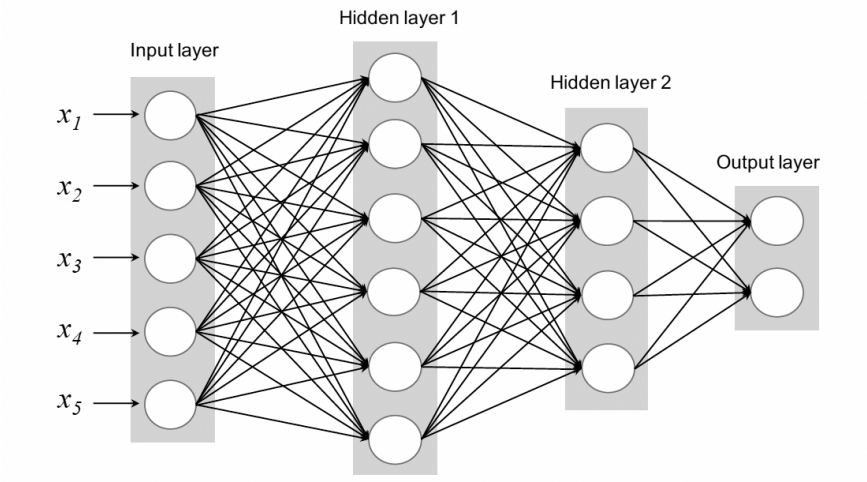


Figure 2.10: Diagrammatic representation of the FFNN architecture.

FFNN layers are categorized into three types [Anderson and McNeill, 1992, Paola and Schowengerdt, 1995, Warner and Misra, 1996]:

1. Input layer: Receives raw data input (e.g., word or character embeddings for text).
2. Hidden layer(s): Processes incoming information, similar to human perceptual nerves. Multiple layers may be involved in information transmission.
3. Output layer: Generates the final output of the network, with the number of neurons depending on the task type.

Each neuron processes information by multiplying incoming signals by corresponding weights, summing the results, and passing them through a nonlinear activation function to produce an output value.

Training a neural network begins with random weight initialization [Cao et al., 2018, Yu and Xu, 2014]. The network is then continuously adjusted to produce outputs similar to training data labels by modifying parameters. This process employs a loss function to measure the gap between output and target values [Hoffman et al., 2018, Jadon, 2020]. For instance, in text generation tasks, cross-entropy is often used to measure the distance between predicted and actual distributions. The smaller the loss value, the closer the predicted distribution is to the actual distribution.

The loss function serves as the objective function in neural network optimization, with the training process aimed at minimizing this function. This approach allows neural

networks to learn and improve their performance on various tasks by adjusting their internal parameters based on the feedback provided by the loss function.

The minimization of the loss function is achieved through the backpropagation algorithm [Rumelhart et al., 1986]. This process begins at the output layer and employs gradient descent to progressively update parameters in each preceding layer. The direction of this update is opposite to that of forward propagation, hence the term "backpropagation". This method, based on gradient descent, aims to update parameter values in the direction opposite to the gradient of the objective function, thereby minimizing it.

In practice, neural networks are predominantly optimized using the batch stochastic gradient descent algorithm [Robbins and Monro, 1951, Amari, 1967, Bottou et al., 1991]. The principle of this algorithm is as follows [Bottou, 1998, Bottou and Bousquet, 2007]:

$$w_{t+1} = w_t - \eta \frac{1}{n} \sum_{x \in \xi} \nabla l(x, w_t) \tag{2.1}$$

Where at the t-th training iteration, w_t represents the model parameters. The batch size is denoted by n, and the learning rate is represented by η. The gradient of the loss function with respect to w_t is given by $\nabla l(x, w_t)$, where x is a sample from the batch ξ.

The batch size denotes the number of samples used in a single training step, while the learning rate determines the magnitude of parameter adjustments during each iteration, influencing the convergence of the loss function to its minimum value.

Building on these fundamental neural network concepts, our focus shifts to a specific model type i.e., sequence-to-sequence (seq2seq) model. This model is employed in this dissertation, each adapted to distinct DRS format representations. Furthermore, we introduce pre-trained neural models, a widely adopted technique in recent Natural Language Processing (NLP) tasks that generally yields significant performance improvements.

2.2.1 Sequence-to-Sequence Models

While Feed-Forward Neural Networks (FFNNs) serve as a fundamental neural architecture, they have limitations in practical applications. FFNNs treat each input independently, disregarding sequence order or correlations. To address these limitations, Recurrent Neural Networks (RNNs) [Elman, 1990] and their variant, Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] networks, were developed.

RNNs feature a recurrent structure that passes information between time steps, making them suitable for sequence data processing, including NLP tasks. However, RNNs struggle with capturing long-distance dependencies and face the vanishing gradient problem [Basodi et al., 2020]. LSTM networks use a specialized gating mechanism to control information flow, mitigating shortcomings of RNN in processing long sequences and improving information retention and transfer.

LSTMs have shown effectiveness in various NLP tasks, including text classification [Zhou et al., 2016, Liu and Guo, 2019] and sentiment analysis [Divate, 2021]. For text-

to-text tasks like machine translation or dialogue systems, LSTMs often serve as core components in sequence-to-sequence (seq2seq) models, also known as encoder-decoder frameworks [Cho, 2014, Sutskever, 2014]. These models typically use an encoder to process and encode an input sequence into a fixed-length context vector, followed by a decoder that generates an output sequence. Seq2seq models can handle variable-length input and output sequences without requiring pre-specified fixed lengths [Venugopalan et al., 2015, McCann et al., 2017].

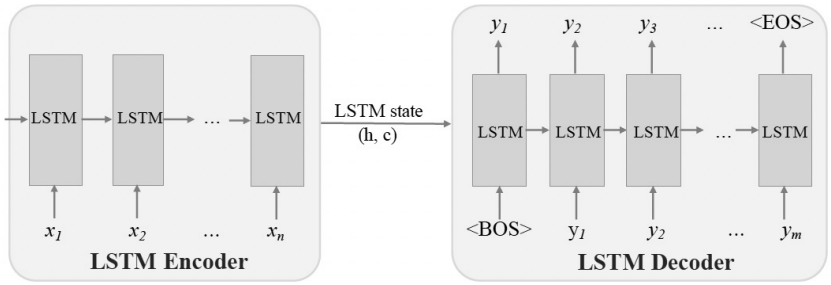


Figure 2.11: Diagrammatic representation of the basic sequence-to-sequence architecture utilizing LSTM with single encoder-decoder layers.

Figure 2.11 illustrates an LSTM-based seq2seq architecture. In this model, an LSTM layer (or multi-layer LSTM) functions as an encoder, processing the input sequence x = (x1, x2, ..., xn) and producing a final internal state (h, c) as the context vector. Here, h represents the hidden state (current working memory) and c represents the cell state (long-term memory). Another LSTM layer acts as a decoder, using the intermediate state vectors of encoder (h, c) as its initial state.

Special tokens $\langle BOS \rangle$ (beginning of sequence) and $\langle EOS \rangle$ (end of sequence) are added to the target sequence. The decoder generates an output probability distribution for each possible token in the vocabulary, typically using a fully connected layer and softmax activation function. To generate the actual output token, the decoder employs beam search, considering multiple high-probability tokens and selecting the most promising one at each time step. This process repeats until an $< EOS >$ token is generated or the maximum output sequence length is reached [Post and Vilar, 2018, Kang et al., 2022].

Attention Mechanism: In the traditional encoder-decoder model, the decoder relies solely on the final state of the encoder as context information to generate the entire output sequence. This approach can be problematic for very long input sequences or when crucial information is dispersed across different time steps. These limitations can result in information loss and decreased performance, primarily due to the constrained ability of the encoder to effectively capture distributed information.

To address this issue, the attention mechanism was introduced [Bahdanau et al., 2014]. This innovation transforms the functionality of the decoder by allowing it to dynamically compute a new context vector based on the currently generated token, rather than encoding the entire input sequence into a fixed-length context vector. This dynamic approach employs distinct context vectors at each time step, effectively resolving the problem of information loss.

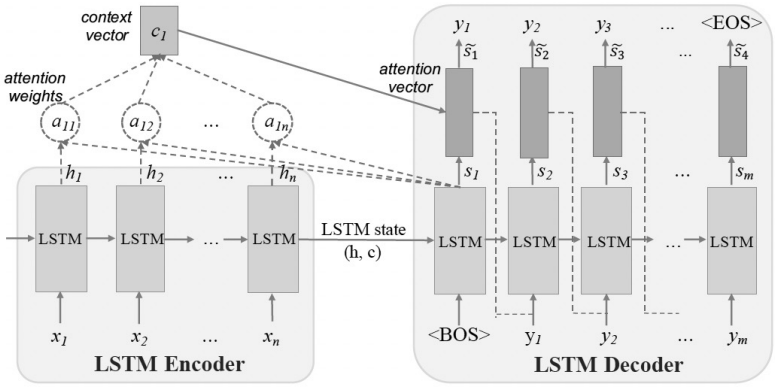


Figure 2.12: Diagrammatic representation of the basic sequence-to-sequence architecture with an attention mechanism. Although the decoder can receive a different context vector at each time step, only the first time step is illustrated here.

Figure 2.12 illustrates the seq2seq architecture incorporating the attention mechanism. Unlike the standard seq2seq model, the attention-enhanced decoder calculates a weighted sum at each time step, considering the current context vector and the intermediate states of the encoder. This process determines which parts of the input sequence should be prioritized. The resulting attention weights enable the decoder to adjust its focus dynamically, capturing relevant information from the input sequence more effectively.

This approach allows the decoder to obtain a different context vector at each step, improving the ability of the model to handle long and variable-length sequences and to flexibly capture important aspects of the input [Hao et al., 2019, Lin et al., 2022].

Mathematically, the context vector is defined as [Luong et al., 2015, Sennrich, 2017]:

$$c_t = \sum_{i=1}^{T} \alpha_{ti} h_i \tag{2.2}$$

where α_{ti} represents the attention weights, calculated by normalizing attention scores e_{ti} through the softmax function:

$$\alpha_{ti} = \text{softmax}(e_{ti}) = \frac{\exp(e_{ti})}{\sum_{k=1}^{T} \exp(e_{tk})} \tag{2.3}$$

The initial hidden state of the decoder at time step t is computed as:

$$s_t = \tanh(W[s_{t-1}, y_{t-1}]) \tag{2.4}$$

where y_{t-1} is the previously generated output (often referred to as "previous label/token") at time step t-1, and tanh is the activation function. The attention scores are calculated as:

$$e_{ti} = s_t^T W_a h_i \tag{2.5}$$

The attentional hidden state of the decoder is then determined by [Luong et al., 2015]:

$$\tilde{s}_t = \tanh(W_c[s_t, c_t]) \tag{2.6}$$

Finally, this attentional vector is passed through a softmax layer to produce the predictive distribution:

$$o_t = \text{softmax}(V \cdot \tilde{s}_t) \tag{2.7}$$

In these equations, V, W, Wa, and Wc are weight matrices to be learned during training. We will be using the same architecture in our experiments listed in the coming chapters.

2.2.2 Pre-trained Language Models

In recent years, large-scale pre-trained sequence-to-sequence language models have demonstrated significant utility across various NLP tasks, including semantic parsing and text generation [van Noord et al., 2020, Bai et al., 2022, Bevilacqua et al., 2021]. This section will primarily discuss the BART model [Lewis et al., 2020], a prominent pre-trained seq2seq language model that features heavily in Chapter 4 of this thesis. To provide context, we will first examine the evolution leading to pre-trained language models (PLMs) and briefly introduce techniques employed in other chapters of this dissertation.

NLP tasks typically require the conversion of text data into numerical vectors to enable processing and comprehension by neural network models. The quality of these representations significantly impacts both model performance and overall task effectiveness. Early approaches utilized pre-trained static word vectors such as Word2Vec [Mikolov et al., 2013] and GloVe [Pennington et al., 2014]. Despite the importance of Word2Vec, researchers continued developing methods to train embeddings from scratch as part of model parameters for specific NLP tasks. This approach, while resource-intensive, offers greater flexibility in addressing task-specific requirements. Subsequently, PLMs-based methods gained prominence, focusing on context-aware word embeddings. Models like ELMo [Peters et al., 2018], OpenAI GPT [Radford et al., 2018], and BERT [Devlin et al., 2019] have significantly advanced the field by capturing rich contextual information and achieving state-of-the-art results across various NLP tasks. Chapters 3 of this work

focus on embeddings trained from scratch, while Chapter 4 utilizes BERT embeddings, which are particularly beneficial for tasks with natural language input to the encoder, such as parsing tasks [van Noord et al., 2020].

While BERT excels in capturing bidirectional context in text, BART adopts a different approach to text generation. BART integrates bidirectional and auto-regressive functions, offering a robust text generation model applicable to various text and sequence generation tasks [Lai et al., 2021, Cui et al., 2021]. Unlike the focus of BERT on language understanding and contextualization, BART specializes in generating coherent and contextually accurate text. Although recent GPT models [Brown et al., 2020, OpenAI, 2023] have surpassed BART in overall performance and functionality, BART maintains advantages in certain practical applications due to its relatively compact model size. Chapter 4 of this work focuses on leveraging the BART model to enhance performance in multilingual semantic parsing and text generation tasks.

Text-to-Text Transfer Transformer (T5): T5 [Sheang and Saggion, 2021], along with its multilingual variant mT5 [Xue et al., 2021] and byte-level version byT5 [Xue et al., 2022], represent another significant advancement in pre-trained language models. These models, which feature prominently in Chapters 3, Chapters 4, Chapters 5, and Chapters 6 of this work, adopt a unified text-to-text framework that reformulates all NLP tasks as text generation problems. This approach allows for a more versatile and generalizable model architecture compared to task-specific designs. T5 builds upon the transformer architecture [Vaswani et al., 2017] and employs a novel pre-training objective called "span corruption", which involves masking random spans of input text and training the model to reconstruct them. This method proves particularly effective for both understanding and generation tasks. The mT5 model extends the capabilities of T5 to support 101 languages, making it a powerful tool for multilingual NLP tasks. byT5 further innovates by operating directly on UTF-8 bytes rather than tokenized inputs, potentially improving performance on tasks involving rare words or non-standard text. While these models may not match the sheer scale and performance of LLMs, they offer a balance of efficiency and effectiveness that makes them particularly suitable for a wide range of research and practical applications. Chapters 4 of this thesis demonstrate the application of T5, mT5, and byT5 in improving performance across various multilingual NLP tasks, showcasing their versatility and effectiveness in handling complex linguistic challenges. Chapters 3, Chapters 5, and Chapters 6, we focused on byT5 only with different aspects of NLP applications.

Fine-tuning: Fine-tuning is a crucial technique that complements pre-trained models [Dai and Le, 2015, Howard and Ruder, 2018]. This process involves further training and adjusting some or all model parameters of a pre-trained model for specific tasks and domains. Pre-trained models, typically trained on extensive datasets, possess inherent versatility and generalization capabilities. The primary objective of fine-tuning is to adapt these pre-trained models to specific tasks with limited data, thereby enhancing their performance in targeted applications.

Fine-tuning serves as a bridge between pre-training and task-specific training, enabling models to adapt more effectively to new tasks while conserving computational resources and time. The conventional fine-tuning approach involves continued training

of a pre-trained model using a small amount of task-specific data. During this process, the pre-trained model's weights are updated to better align with the task at hand.

The extent of fine-tuning required is dependent upon the similarity between the pre-training corpus and the task-specific corpus. When these corpora share significant similarities, minimal fine-tuning may suffice. Conversely, more substantial fine-tuning may be necessary when the corpora differ significantly.

In Chapter 3 and Chapter 4 of this thesis, we employ this fine-tuning technique to enhance task performance. In Chapter 5 and Chapter 6 we used our fine-tuned models without any additional pre-training or fine-tuning to exploit different applications of semantic processing. This approach allows us to leverage the power of pre-trained models while tailoring them to our specific research objectives, potentially leading to improved results in our targeted applications.

2.2.3 Neural Semantic Parsing and Text Generation

This section examines previous neural network-based approaches to semantic parsing and meaning-to-text generation, with a particular focus on DRS parsing and generation systems. While research on DRS is relatively limited, valuable insights can be drawn from semantic parsing and generation efforts in other formalisms, especially the widely adopted AMR. Given the rich corpus resources and extensive research base of AMR, it is crucial to conduct a comprehensive review of the relevant literature in this domain, as it may offer applicable insights for DRS-related tasks.

Semantic Parsing: Semantic parsing methods can be broadly categorized into two groups: rule-based and neural network-based approaches. Rule-based methods typically apply a set of manually crafted rules to the syntactic analysis of a sentence to generate a formal meaning representation. These rules are often domain-specific and require substantial expertise to design [van Noord, 2021]. A notable DRS-related parser is Boxer [Bos, 2008], which combines rules and statistical methods. However, this thesis focuses on neural network-based methods due to their transformative impact on the field.

Neural models have become the dominant approach in semantic parsing, typically yielding the best performance. Various techniques have been incorporated into neural-based systems, including linguistic feature additions [van Noord et al., 2019], character-level neural networks [van Noord et al., 2018], and pre-trained language model embeddings [van Noord et al., 2020]. Chapters 3 and Chapter 4 of this work employ the latter two technologies as baseline models for English semantic parsing, acknowledging their efficacy in improving overall parsing performance.

Beyond the seq2seq model, two other research directions have emerged: tree-based methods [Liu et al., 2018, 2019] and graph-based methods [Fancellu et al., 2019, Fu et al., 2020]. These approaches convert box-format DRS into tree-based or graph-based representations, respectively, and use structure-aware decoders to improve performance. While most studies have focused on English texts, research is gradually extending to other languages. [Fancellu et al., 2019] made an initial attempt at multilingual DRS parsing, training parsers for each language from scratch with a complex graph data transformation process. In Chapter 4, we experimented with multi-lingual semantic

parsing by expanding the parsing scope to include English, German, Dutch, and Italian—aligning with the languages studied by [Fancellu et al., 2019]—and Urdu as a novel contribution to semantic parsing paradigm.

AMR parsing techniques employing neural models can be categorized into three main approaches: transition-based methods [Fernandez Astudillo et al., 2020, Zhou et al., 2021], sequence-to-graph methods [Zhang et al., 2019, Cai and Lam, 2020], and sequence-to-sequence methods [Xu et al., 2020, Bevilacqua et al., 2021]. Transition-based parsers process sentences sequentially, constructing the graph incrementally by alternating between inserting new nodes and establishing new edges. This approach frames parsing as a sequence of action predictions, with performance heavily dependent on effective modeling of the parser state at each decision step.

Sequence-to-graph AMR parsers operate by simultaneously determining new nodes and their connections to existing nodes at each time step. This approach allows for dynamic expansion of the graph structure, capturing node relationships as the input sequence is processed. In contrast, sequence-to-sequence parsing linearizes the AMR graph, transforming the task into a sequence-to-sequence transduction. A key feature of this method is its use of shared vocabulary and equal treatment of concepts and relational predictions [Cai and Lam, 2020].

Recent advancements in AMR parsing performance have been largely attributed to the successful implementation of pre-training techniques. For instance, [Zhang et al., 2019, Cai and Lam, 2020] utilize the pre-trained language model BERT for sentence encoding, a technique also applied in DRS parsing [van Noord et al., 2020]. [Bevilacqua et al., 2021] fine-tuned BART for AMR parsing, extending a Transformer encoder-decoder model pre-trained on English text denoising to work with AMR. These developments in AMR parsing offer valuable insights that could potentially be applied to enhance DRS parsing techniques in future research.

DRS parsing research has predominantly relied on the seq2seq model, largely due to the clause-format structure of available DRS corpora suitable for neural network models. Unlike AMR, applying graph models to clause-format DRS is less straightforward. However, variable-free DRS presents a promising avenue to address this limitation. While this thesis does not employ transition-based or graph-based parsing approaches, the related work in AMR parsing serves as inspiration for future DRS parsing endeavors.

Meaning-to-text Generation: Similar to DRS parsing, previous efforts in DRS-to-text generation can be broadly categorized into rule-based methods [Basile and Bos, 2011] and neural network-based methods [Liu et al., 2021]. However, DRS-to-text generation has only recently gained attention from NLP researchers [Basile and Bos, 2011, Narayan and Gardent, 2014, Basile, 2015]. [Liu et al., 2021] introduced a novel approach to transforming DRS into trees, with boxes representing subtrees and conditions as children. They presented a tree-LSTM-based DRS-to-text model, which, despite notable results, faces challenges similar to tree-based DRS parsing methods, particularly in reproducibility due to the numerous rules required for DRS tree format conversion.

The extensive annotated corpora for AMR have led to a greater research focus on AMR-to-text compared to DRS-to-text. AMR-to-text generation approaches can be categorized into two main classes: graph-to-sequence models [Beck et al., 2018, Damonte and Cohen, 2019] and sequence-to-sequence models [Konstas et al., 2017]. Graph-

to-sequence models use a graph encoder for AMR graphs and a sequence decoder for sentence generation, aiming to preserve structural information [Song et al., 2020, Wang et al., 2020]. Similar approaches have been applied to other graph-structured data like Knowledge Graphs [Ribeiro et al., 2020, Song et al., 2020] and SQL queries [Xu et al., 2018]. Sequence-to-sequence models linearize AMR graphs into sequences with bracket representation, treating it as a seq2seq task using either randomly initialized or pre-trained models for encoding [Ribeiro et al., 2021b, Bevilacqua et al., 2021, Procopio et al., 2021].

AMR research has expanded to non-English languages, including both parsing [Blloshmi et al., 2020] and multilingual AMR-to-text generation tasks [Sobrevilla Cabezudo and Pardo, 2019, Ribeiro et al., 2021a]. This research, like DRS, is based on the theory of language-neutral meaning representation, although some studies indicate AMR's limitations as an interlingua due to underlying ontological distinctions [Xue et al., 2014].

In Chapter 3 of this work, we employ LSTM-based and byT5-based seq2seq models for data augmentation and data delexicalization experiments for DRS-to-text generation, similar to [Konstas et al., 2017]. In Chapter 4, we adopt a fine-tuned BART (mBART) and T5 (including all variants e.g., mT5 and byT5) for both DRS parsing and generation in the context of multi-faceted data augmentation approaches, akin to [Bevilacqua et al., 2021] on AMR parsing. Our approach uniquely integrates multi-tasking (parsing and generation) and multi-lingual support including English, Italian, German, Dutch, and Urdu. A related work by [Bai et al., 2022] proposes a monolingual AMR-based framework where pre-training and fine-tuning share the same data format to facilitate knowledge transfer. In Chapter 5, we provide alternate evaluation measures that take into account both the structural similarity and linguistic fidelity to evaluate semantic parsing and generation tasks. For this evaluation, we used pre-trained byT5 to perform data transformations from DRS to text and text to DRS representations. In Chapter 6, with the motivation of exploiting task reversibility of DRS parsing and generation, we utilized fine-tuned byT5 through Parse-Generate-Parse and Generate-Parse-Generate pipeline approaches.

Chapter 3

Data Transformation Strategies for DRS Clause Format: Augmentation and Delexicalization for DRS-to-Text Generation

Neural approaches to natural language generation from DRS have shown promising results in recent years. However, these models often struggle with data scarcity and the complexity of translating logical structures into natural language. Our research addresses these challenges by exploring innovative techniques in data augmentation and delexicalization for DRS-to-Text generation.

We investigate whether it is possible to effectively augment logical data representations like DRS, and how to generate contextually similar new data. Our approach focuses on selectively augmenting proper and common nouns using WordNet supersenses, allowing us to replace noun entities inside the dataset i.e., *in-context*, and outside the dataset i.e., *out-of-context*. This raises interesting questions about the role of grammatical, semantic, pragmatic, and world knowledge in the learning process of neural models.

Furthermore, we examine the impact of delexicalization on DRS-to-Text generation. Given that DRS representations are tightly linked to external lexical resources like WordNet and VerbNet, we developed novel methods for selective delexicalization. We explore how supersenses can enhance the generalization power of neural models, particularly for nouns, and investigate the effects of combining delexicalized and fully lexicalized data during training.

Our experiments employ both sequence-to-sequence models and fine-tuned Transformer models to assess the effectiveness of these techniques. We evaluate whether augmentation and delexicalization lead to improved model performance and generaliz-

ability. Additionally, we explore the behavior of pre-trained LLMs like ChatGPT and Claude when presented with DRS structures as prompts, offering insights into how these advanced general-purpose models handle complex logical representations.

The results of our study are encouraging. We find that carefully designed data augmentation and delexicalization strategies can indeed improve the performance of DRS-to-Text generation models. Our findings suggest that these techniques help models focus more on the underlying semantic structure rather than specific lexical content, leading to more robust and generalizable text generation. This research not only advances the field of DRS-to-Text generation but also provides valuable insights for handling logical data representations in natural language processing more broadly.

Chapter adapted from

1. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2024. "Exploring Data Augmentation in Neural DRS-to-Text Generation". In Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2164–2178, St. Julian's, Malta. Association for Computational Linguistics.

2. Amin, M.S., Anselma, L., Mazzei, A. (2024). Improving DRS-to-Text Generation Through Delexicalization and Data Augmentation. In: Rapp, A., Di Caro, L., Meziane, F., Sugumaran, V. (eds) Natural Language Processing and Information Systems. NLDB 2024. Lecture Notes in Computer Science, vol 14762. Springer, Cham. https://doi.org/10.1007/978-3-031-70239-6_9

3.1 Data Augmentation

Data augmentation has emerged as a powerful technique in the field of machine learning, particularly in scenarios where data scarcity poses a significant challenge. Originally developed for computer vision applications, data augmentation involves systematically increasing the number of data examples by introducing controlled variations to the original data [Feng et al., 2021]. In computer vision, this might involve simple transformations such as rotating, flipping, or cropping images. For instance, a single picture of a cat might be augmented to create multiple training examples by slightly altering its orientation or zoom level, thereby enriching the dataset without the need for additional real-world data collection.

However, the application of data augmentation techniques in NLP presents unique challenges. Unlike the continuous nature of image data, textual data is discrete and highly structured, making it more sensitive to manipulations. Common approaches in NLP include random swapping of words, random insertion or deletion of text, synonym replacement, back translation, and the use of generative models to create contextually aware new data [Feng et al., 2021, Shorten and Khoshgoftaar, 2019]. For example, the sentence "The cat sat on the mat." might be augmented to "The feline rested on the rug." using synonym replacement. However, these techniques must be applied sensibly to maintain grammatical correctness and preserve the original meaning of the text.

The complexity of language makes it crucial that augmented text remains not only grammatically correct but also semantically and pragmatically coherent. A poorly augmented sentence could lead to illogical or misleading training data. For instance, blindly replacing words in the sentence "The Eiffel Tower is in Paris" could result in an incorrect statement like "The Eiffel Tower is in London" which would be detrimental to the learning of the model. This highlights the fine balance required in NLP data augmentation between increasing dataset size and maintaining data quality [Hou et al., 2018, Dong et al., 2017].

Recent advancements in NLP have seen increased focus on text generation from meaning representations, such as Abstract Meaning Representation (AMR) and Discourse Representation Structure (DRS). Researchers have employed transformer and encoder-decoder-based neural models to tackle these complex tasks [Basile and Bos, 2011, Dušek et al., 2019, Noord, 2019, Wang et al., 2021b, Amin et al., 2022b, Wang et al., 2023a]. The DRS-to-Text generation task presents a unique opportunity for data augmentation. Unlike more straightforward text-to-text tasks, DRS-to-Text involves translating a logical representation into natural language. This logical structure allows for more controlled and semantically aware augmentation strategies. For instance, we can modify specific elements of the DRS (such as entities or properties) while maintaining the overall logical structure, potentially leading to more meaningful and diverse augmentations.

Our research focuses on exploiting this unique aspect of DRS-to-Text generation by designing and evaluating data augmentation strategies specifically for proper nouns and common nouns. We propose using Supersense Tagging (SST) to create new training sentences with both in-context and out-of-context nouns. This approach allows us to investigate how lexical information impacts the performance of neural DRS-to-Text systems.

For example, consider the DRS representation of the sentence "John bought a car in New York." We might augment this by replacing "John" with other proper nouns (e.g., "Maria"), "car" with other common nouns of similar supersense (e.g., "bicycle" or "boat"), and "New York" with other location names (e.g., "Tokyo" or "Berlin"). This creates multiple variations of the original sentence while maintaining the true pragmatics and logical structure of the DRS.

It is important to note that our augmentation techniques may occasionally generate factually incorrect texts. For instance, augmenting "At dawn, the sun rises" could potentially produce "At midnight, the sun rises". However, we argue that this capability is not necessarily detrimental. Human language often includes factually incorrect or fictional statements, and the ability to generate such texts can be valuable in certain contexts, such as creative writing or hypothesis generation.

This research makes several key contributions regarding data augmentation for DRS-to-Text generation. First, it demonstrates the feasibility of effectively augmenting logical data representations, specifically DRS, through advanced augmentation techniques. Additionally, methods are developed for generating new data that maintains contextual similarity to the original, ensuring semantic and contextual integrity in augmented datasets. The research further investigates the role of in-context and out-of-context vocabulary in the performance of both character-level and word-level decoder models, providing valuable insights into their impact on language modeling. A thorough analysis of grammatical, semantic, pragmatic, and world knowledge contributions to the learning process is conducted, highlighting their significance in model training. Moreover, this work shows that data augmentation results in enhanced performance for sequence-to-sequence models and fine-tuned Transformer models, improving overall robustness and generalization. Finally, the research explores how general-purpose LLMs, such as ChatGPT and Claude, interpret and process DRS structures when provided as prompts, contributing to the understanding of prompt-based NLP applications. To our knowledge, this represents the first comprehensive study on data augmentation in DRS-to-Text generation, building upon preliminary work on tense augmentation [Amin et al., 2022b].

The statistical nature of neural networks complicates the analysis of the types of knowledge the system acquires. For instance, when presented with an example like "Brad Pitt is an actor" it is unclear whether the network learns that verbs follow subjects (grammatical competence), that a man can be an actor (semantic and pragmatic knowledge), or that a specific individual, Brad Pitt, is an actor (world knowledge). Understanding how to leverage this multi-layered learning process remains a challenge. As a supplementary contribution to our study on data augmentation, we aim to explore these theoretical questions as well.

The remaining chapter is organized as follows: in Section 3.2, we explain in detail logical data augmentation with nouns—both proper and common nouns. Section 3.3 of this chapter discusses data delexicalization strategies applied on the dataset, with specific focus on proper and common nouns in Section 3.4. Methodological implementation for DRS-to-Text generation is presented in detail in Section 3.5. Section 3.6 presents data augmentation results while Section 3.7 presents data delexicalization results conducted in this study. The chapter is concluded in Section 3.8 of this dissertation.

3.2 Logical Data Augmentation with Nouns

Data augmentation in the era of neural DRS-to-text generation is a complex process. This task involves the creation of new augmented examples for training datasets where each example consists of a novel DRS structure paired with its corresponding newly generated sentence. The complexity lies in the need to maintain consistency between the semantic structure represented by the DRS and the linguistic expression of that structure in natural language.

To illustrate this process, consider a simple sentence like "The scientist conducts an experiment in the laboratory". The associated DRS for this sentence would encode various semantic components, including the entities involved (scientist, experiment, laboratory), the action (conducts), and the relationships between these elements. The DRS captures the deeper semantic structure, representing the scientist as the agent performing the action, the experiment as the object of that action, and the laboratory as the location where it occurs.

When augmenting this data, systematic transformations must be applied to both the DRS and the sentence while maintaining consistency between them. For instance, the original sentence might be transformed to "The researcher performs a study in the facility". In this case, the DRS must be modified accordingly, replacing the entity "scientist" with "researcher", changing the action from "conducts" to "performs", substituting "experiment" with "study", and replacing "laboratory" with "facility". This example demonstrates the complex nature of data augmentation in DRS-to-text generation, highlighting the need for careful tracking and modification of both DRS structures and their textual translations.

The importance of this careful tracking cannot be overstated, as these DRS-sentence pairs serve as input-output examples for the neural model. Ensuring accuracy in these pairings is crucial for the neural network to learn correct mappings between DRS structures and their corresponding sentences. To achieve effective data augmentation, the transformations applied must be identical and symmetrical for both elements of the pair. This involves careful consideration of the order and nature of meaning representations in the DRS, ensuring that the corresponding textual translations accurately reflect these changes.

In the context of DRS-to-text generation tasks, we employ various augmentation techniques, often focusing on manipulating proper and common nouns to enhance the diversity and robustness of the dataset. These methods might involve replacing terms with semantically similar alternatives across multiple examples, ensuring that the model learns to generalize across different but related entities. We utilize the gold version of the PMB dataset for experimental purposes. Developed at the University of Groningen as part of a larger research initiative, the PMB serves as a valuable resource for meaning representations and their corresponding textual translations [Abzianidze et al., 2017]. This dataset is typically organized into the standard train-dev-test split, allowing for a structured approach to training, validating, and testing neural models. The train set is used to teach the model the relationships between DRS structures and sentences, while the development set aids in tuning the hyperparameters of the model and assessing its performance during training. The test set is then used to evaluate the model's ability to generalize to unseen data. By leveraging such comprehensive datasets and applying

sophisticated data augmentation techniques, we aim to improve the performance and robustness of neural DRS-to-text generation models.

3.2.1 Data Augmentation with Proper Nouns

In the process of proper noun augmentation, we focused on enriching the dataset by transforming named entities, particularly person names (PER) and geopolitical entities (GPE). Through this technique, we aimed to improve the robustness and generalization capabilities of language models by exploiting them to a wider variety of proper nouns during training. For augmentation perspectives, we concentrated on person names (PER) i.e., both male and female names, and geopolitical entities (GPE), which encompass cities, states, and countries. To identify and extract these entities within the dataset, we utilized spaCy's named entity recognizer (https://spacy.io) — a sophisticated tool available freely for text processing. This analysis revealed a total of $3,773$ proper noun instances in the Gold version of the PMB-3.0.0 dataset, with a distribution skewed towards person names (57%) and city names (30%). The remaining instances were composed of state names (6%), country names (6%), and a small fraction (1%) representing other types, such as island names. Figure 3.1 shows the graphical distribution of proper noun entities (percentage-wise) in the Gold-PMB dataset.

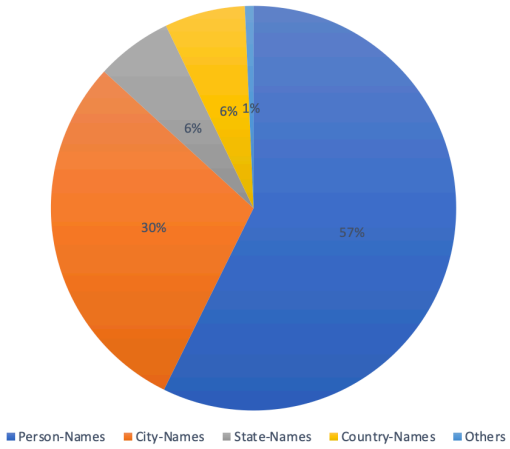


Figure 3.1: Distribution of proper noun entities in Gold-PMB dataset.

To investigate the effects of incorporating external linguistic information into the dataset vocabulary, we employed two distinct strategies for proper noun replacement. The first method, termed "inside context" or "*in-context*" replacement, involved substituting proper nouns with other names already present within the same dataset. This approach maintains the internal consistency of the dataset while still introducing variability. For instance, in a sentence like "John visited Paris", potential augmentations might include "Alice visited Paris" or "John visited Rome" where both the new person name and the new city name are drawn from the existing dataset.

The second strategy, known as "outside context" or "*out-of-context*" replacement, took a more expansive approach by introducing names that were not originally present in the dataset. This method aimed to broaden the vocabulary and expose the model to a wider range of proper nouns. For person names, we selected replacements based on global frequency rankings, using information sourced from ChatGPT to identify names that were not already included in the dataset. When dealing with GPE names, replacements were chosen based on their geographical distribution, ensuring that the new entities were not part of the existing dataset but shared similar characteristics with the original names. For *out-of-context* replacement, we performed a manual filtration process to exclude named entities that were already present in the data set. This approach helped to maintain the semantic integrity of the sentences while introducing new vocabulary. An illustrative example of this "outside context" replacement is the transformation of the sentence "The weather of Dubai is very hot and dry". In this case, "Dubai" might be replaced with "Sharjah" as both cities experience similar climatic conditions. This substitution preserves the overall meaning and context of the sentence while introducing a new geographic entity to the dataset.

The implementation of these augmentation strategies serves multiple purposes. First, it allows us to assess how the inclusion of both internal (from within the dataset) and external (from outside the dataset) entities impacts the overall effectiveness of the dataset. Second, it provides a means to evaluate the performance of the model when exposed to a more diverse range of proper nouns. This approach can potentially improve the ability of the model to generalize across different named entities and contexts. To provide a clear illustration of these augmentation techniques, we have included different examples demonstrating various instances of name substitutions in Table 3.1. These examples showcase how different types of proper nouns (person names, city names, etc.) are replaced using both the "inside context" and "outside context" approaches, offering a concrete visualization of the augmentation process and its potential to diversify the dataset.

3.2.2 Data Augmentation with Common Nouns

The augmentation of common nouns presents unique challenges, particularly in preserving the contextual meaning of sentences. To address this, we developed a sophisticated approach utilizing WordNet-based Super Sense Tagging (SST). This method associates each noun with a category based on its contextual sense within the sentence, providing a more refined way to perform noun replacements.

For the extraction of noun entities, we have used SpaCy again which resulted in $6,193$ common nouns from the dataset, categorizing them into 26 different lexical categories of WordNet. Figure 3.2 displayed the SST-based categorical division of common noun entities in the Gold-PMB dataset. These categories encompass a wide range of semantic fields, including act, artifact, body, cognition, communication, event, feeling, food, group, and motion, as outlined in the work of [Ciaramita and Johnson, 2003].

The common noun augmentation strategy employed in this research involves two primary dimensions: the source of replacement nouns (either within the dataset or external to it) and the preservation or non-preservation of the supersense (SS) category. This approach results in four distinct combinations for noun replacement:

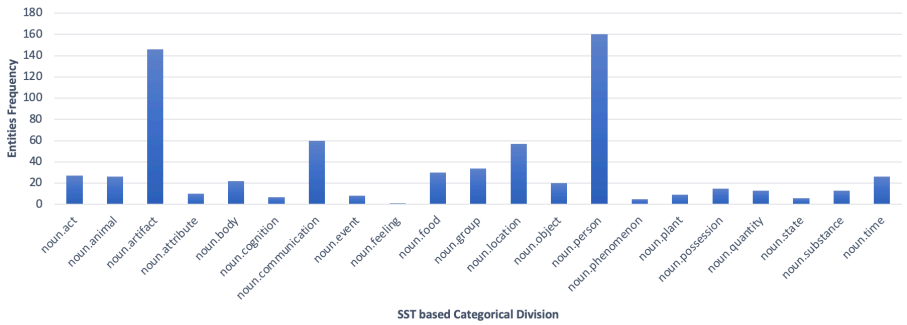


Figure 3.2: Supersense tagging based categorical division of common noun entities in Gold-PMB dataset.

1. Inside Context without SS: This method involves replacing a common noun with another from within the dataset, disregarding the preservation of the supersense category. While this approach maintains grammatical correctness, it can significantly alter the semantic context of the sentence. For example, replacing "cat" with "chair" would fall under this category, potentially changing the meaning of the sentence due to the different SS categories of these nouns ("noun animal" vs. "noun artifact").

2. Inside Context with SS: Here, a common noun is replaced with another from within the dataset that belongs to the same SS category. This method preserves both semantic and contextual integrity. An example would be substituting "cat" with "dog", as both belong to the "noun animal" category.

3. Outside Context with SS: This approach involves replacing a common noun with one from outside the dataset while maintaining the same SS category. We utilized the WordNet lexical database to identify hypernyms of the original nouns that fall within the same category. For instance, replacing "cat" with "feline" (a hypernym in the same SS category) from outside the dataset.

4. Outside Context without SS: In this method, a common noun is replaced with one from outside the dataset, potentially belonging to a different SS category. This is achieved by using WordNet synonyms, though it may disrupt the semantic coherence of the sentence. An example would be replacing "cat" with "vehicle".

For the procedures involving external lexical information (Methods 3 and 4), the WordNet lexical database plays a crucial role. In Method 3, we replaced common nouns with their WordNet hypernyms while ensuring the new noun remains within the same SS category. Method 4 involved noun replacement using WordNet synonyms without necessarily preserving the SS category. It is important to note that certain combinations were not explored for proper nouns, such as changing geographic locations (GPE)

between different types (e.g., city to state or country). This decision was based on the principle of minimal variation in meaning, aiming to maintain the integrity of the sentence while still augmenting the dataset effectively.

The graphical representation in Figure 3.3 illustrated the transformation process in DRS format, showcasing how both proper nouns and common nouns are augmented while maintaining semantic consistency. This visual aid is particularly useful for understanding how the underlying semantic structure is preserved during the augmentation process. In the example provided, the original sentence "Brad Pitt is an actor" is transformed into "Louis Olivia is a performer". This transformation demonstrates two key aspects of the augmentation process: (1) "Brad Pitt" (highlighted in blue) is replaced with "Louis Olivia", showcasing how person names can be substituted while maintaining the overall semantic role within the sentence; (2) "actor" (highlighted in green) is replaced with "performer", illustrating how common nouns can be substituted with semantically related terms.

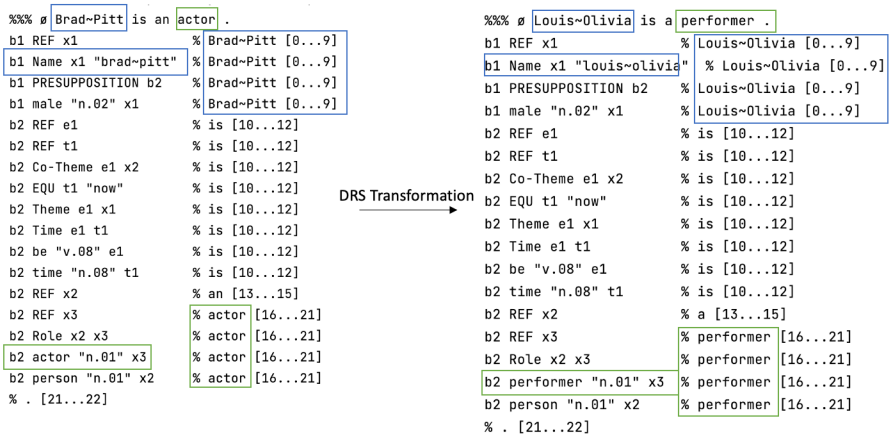


Figure 3.3: Graphical representation of the DRS transformation as a proper noun (in blue) and common noun (in green). The DRS on the left generates the sentence "Brad Pitt is an actor.", while the DRS on the right generates "Louis Olivia is a performer."

The Table 3.1 further elaborates on these concepts by providing textual examples of various augmentation techniques. It is organized into three main categories: (1) Proper Noun Augmentation: This section demonstrates how proper names, including both person names and place names, are transformed. For example, "Brad Pitt" becomes "Louis Olivia", and "Turin" is changed to "Venice". These examples show how the augmentation process can introduce variability in named entities while preserving the overall context of the sentence. (2) Common Noun Augmentation: This part illustrates how common nouns are substituted with semantically related terms. The transformation of "actor" to "performer" and "book" to "novel" shows how the augmentation process can introduce synonyms or related concepts while maintaining the overall meaning of the sentence. (3) Combined Proper Noun and Common Noun Augmentation: This section

demonstrates how both proper and common nouns can be simultaneously augmented within a single sentence. The example "The Mona Lisa hung above the antique table." becoming "The Leonardo da Vinci hung above the antique furniture." showcases a more complex transformation involving multiple elements of the sentence.

Table 3.1: Different flavors of augmentation are applied to the dataset as single and blended data transformations. (PN = Proper Noun; CN = Common Noun)

Transf Type	Original Text	Transformed Augmented Text
PN	Brad Pitt is an actor.	Louis Olivia is an actor.
	Alice and Bob work for this company.	Maria and Tom work for this company.
	Turin is a beautiful city.	Venice is a beautiful city.
	Indiana is a very famous state.	Georgia is a very famous state.
	China is one of the top 5 populous countries in the world.	Indonesia is one of the top 5 populous countries in the world.
CN	Brad Pitt is an actor.	Brad Pitt is a performer.
	Alice and Bob work for this company.	Alice and Bob work for this institution.
	Turin is a beautiful city.	Turin is a beautiful municipality.
	We painted the house green.	We painted the building green.
	The book rested on the table.	The novel rested on the furniture.
PN and CN	Brad Pitt is an actor.	Louis Olivia is a performer.
	The Mona Lisa hung above the antique table.	The Leonardo da Vinci hung above the antique furniture.
	Alice and Bob work for this company.	Maria and Tom work for this institution.
	Noah and Sophia watched a movie at the local theater.	Liam and Emma watched a show at the local edifice.
	Oliver and Isabella enjoyed the view of the mountains from the cabin.	Daniel and Lily enjoyed the view of the elevations from the compartment.

3.3 Data Delexicalization

Delexicalization is a well-known linguistic technique that entails the extraction of lexical information from data, with the primary objective of generalizing it by emphasizing syntactic structures and sentence patterns rather than specific semantic content [van der Lee et al., 2019]. This approach has gained significant attraction in the domains of audio and speech processing, where delexicalization and relexicalization procedures are employed to enhance dialogue systems by preserving prosodic features for text-to-speech applications [Vainio et al., 2009]. The application of data delexicalization/annonymization has also expanded to various NLP tasks including AMR-based semantic parsing and generation [Konstas et al., 2017]. The fundamental rationale behind the utilization of

delexicalization is to augment the model's capacity to generate more natural sentences with improved grammatical and syntactic structures [Sharma et al., 2016]. Furthermore, this technique has been demonstrated to enhance model performance when encountering unseen or out-of-vocabulary words [Shimorina and Gardent, 2018]. It is noteworthy that while recent neural approaches to language generation have achieved optimal performance through end-to-end training, they have also gained prominence in diverse NLG applications, including concept-to-text generation, machine translation (MT), and summarization [Dušek et al., 2018]. For instance, in the context of machine translation, delexicalization can be applied to improve the translation of idiomatic expressions. Consider the English phrase "It is raining cats and dogs". A delexicalized version might replace the specific nouns with placeholders: "It is raining [NOUN] and [NOUN]". This allows the model to focus on the structural pattern of the sentence, potentially improving its ability to generate appropriate translations in the target language.

The methodologies commonly employed for delexicalization in NLG encompass named entity-dependent or exact delexicalization, language-agnostic delexicalization, and delexicalization through pre-trained language models [Zhou and Lampouras, 2020]. It is crucial to recognize that the importance of generating pragmatically accurate text while preserving the semantic and syntactic structure of the sentence presents data delexicalization as a particularly complex task. Indeed, the introduction of grammatically inaccurate or contextually inappropriate delexicalized textual input may result in suboptimal model performance [Dong et al., 2017]. During the delexicalization process, lexical entities are systematically replaced with placeholder tokens. To illustrate, consider the sentence "Tom knocked at the door". Focusing exclusively on nouns, the delexicalized version would be "[PLACEHOLDER] knocked at the [PLACEHOLDER]". In this context, the placeholder can be any generalized tag, depending upon the specific type of delexicalization applied to the data. An additional example to demonstrate this concept could involve a more complex sentence: "The CEO of Apple announced a new product at the annual conference". A delexicalized version might appear as: "[PERSON] of [ORGANIZATION] announced a [OBJECT] at the [EVENT]". This approach allows the model to focus on the underlying structure and relationships within the sentence, rather than specific entities.

Recent research efforts focusing on transformers and encoder-decoder-based neural models for text generation from conceptual or meaning representations have predominantly concentrated on generating text from logical representations and vice versa. These representations include graph-based AMR [Banarescu et al., 2013, Fan and Gardent, 2020, Flanigan et al., 2016], Resource Description Framework (RDF) triples [Castro Ferreira et al., 2020], and DRS [Basile and Bos, 2011, Dušek et al., 2019, Noord, 2019, Wang et al., 2021b, Amin et al., 2022b, Wang et al., 2023a]. This section emphasizes the critical importance of data delexicalization within the framework of formal meaning representation, specifically DRS, in the context of neural DRS-to-Text generation tasks.

In comparison to alternative logical or conceptual data representations, such as AMR and RDF, DRS has been selected for its superior expressiveness and capacity to represent a diverse range of semantic phenomena, effectively capturing logical relations in complex and lengthy sentences. Moreover, the utilization of DRS facilitates a fine examination of the logical form of individual sentences in DRS-to-Text generation, emphasizing syn-

tactic or discourse-level structure rather than only semantic content. As stated in the literature, "by modifying a DRS meaning in a controlled manner, the system's robustness can be closely observed and evaluated accordingly" [Wang et al., 2021b]. However, it is imperative to note that LLMs, lacking essential knowledge of DRS, may introduce undesirable noise into the data if employed for delexicalization or relexicalization processes. This inherent resilience property serves as a significant reason not to utilize LLMs for data delexicalization in this context. For instance, consider a complex sentence like "Although John believed Mary was happy, he later discovered she was actually feeling quite sad". A DRS representation of this sentence would capture not only the basic propositional content but also the temporal and epistemic relationships between the beliefs and discoveries. This level of detail and expressiveness is crucial for accurate text generation and cannot be easily replicated by simpler representations. To elucidate this concept, a graphical representation of the DRS for the sentence "Tom was carrying a bucket of water" is provided in Figure 3.4.

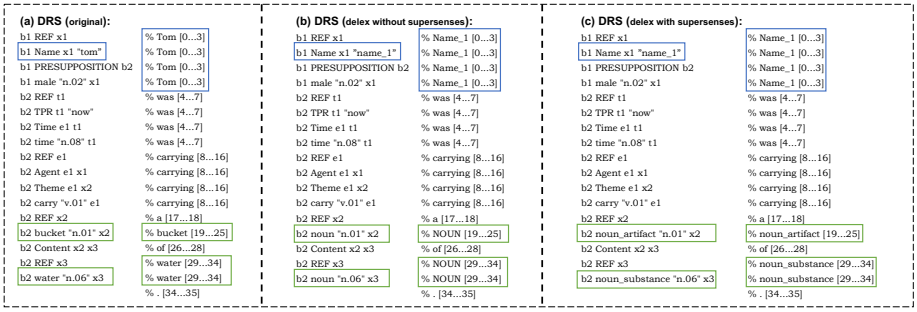


Figure 3.4: Graphical representation of the DRS with lexical (a) and delexical without (b) and with (c) supersenses for the text "Tom was carrying a bucket of water".

The statistical nature of neural networks presents significant challenges in analyzing the specific types of information that the system has acquired during the learning process. Typically, these networks learn grammatical patterns, such as the conventional subject-verb order in sentences like "Brad Pitt is an actor", as well as semantic and pragmatic knowledge, including the understanding that men can assume the role of actors or that a particular individual is an actor (world knowledge). Our study also raises fundamental theoretical questions regarding the optimal methods for leveraging the multilevel structure of neural learning, underscoring the imperative for further investigation into these complex issues.

For instance, consider how a neural network might process the sentence "The old man the boat". The correct interpretation requires understanding that "man" is being used as a verb, not a noun. This example highlights the challenge of extracting syntactic, semantic, and pragmatic knowledge in neural language models, and emphasizes the need for sophisticated delexicalization techniques that can preserve crucial structural information while abstracting away specific lexical content. To further exemplify, consider the sentence "Every farmer who owns a donkey beats it". A DRS representation of this sentence would involve multiple discourse referents (e.g., for "farmer", "donkey", and

the beating event), along with logical operators to represent the universal quantification ("every") and the conditional relationship between owning a donkey and beating it. The delexicalization process in this context might involve replacing specific predicates like "farmer" and "donkey" with more general semantic categories while preserving the logical structure of the representation.

This research makes several key contributions regarding delexicalization and data augmentation of DRS. First, it develops a novel approach for delexicalizing DRS while addressing the strong dependency of lexical entities on external lexical databases like WordNet and VerbNet. Additionally, the experiments demonstrate that incorporating supersenses for nouns significantly enhances the generalization power of neural models. Furthermore, it investigates the behavior of models when augmenting logically delexicalized data with fully lexicalized data, offering valuable insights into data augmentation strategies. The experiments also show that combining delexicalization and augmentation leads to notable improvements in model performance, particularly in sequence-to-sequence tasks. Moreover, it analyzes the behavior of sequence-to-sequence models during pre-training with bi-LSTM and fine-tuning with byT5, providing a comparative understanding of both training approaches. Finally, the research explores how LLMs, such as ChatGPT and Claude, process DRS structures when used as prompts, contributing to advancements in prompt-based language modeling.

3.4 Logical Data Delexicalization with Nouns

In this section, we specifically develop and evaluate data delexicalization methodologies for two distinct lexical categories: (i) proper nouns (PNs) and (ii) common nouns (CNs), leveraging the inherent robustness of neural DRS-to-text generation systems. We have selectively developed and assessed the process of substituting the actual lexical information in the original DRS-text pairs with appropriate placeholders by modifying the proper nouns and common nouns within the dataset. Our investigation encompasses various approaches, both incorporating and excluding the use of supersenses, to generate novel delexicalized training sentences. These methodologies are designed to examine the significance of lexical and semantic information, as well as to enlighten the crucial role of the syntactic structure of logical representations in neural DRS-to-text generation.

Figure 3.4 presents a visual illustration of the DRS transformations, progressing from a fully lexical representation (a), to a delexical form without supersenses (b), and a delexical version incorporating supersenses (c). These transformations highlight the modifications applied to PNs (denoted in blue) and CNs (indicated in green). The initial DRS (a) generates the sentence "Tom was carrying a bucket of water", reflecting a fully lexical translation of the DRS. In contrast, the delexicalized DRS (b) produces "Name_1 was carrying a NOUN of NOUN" exemplifying a more generalized delexicalization approach. Finally, the DRS (c) yields "Name_1 was carrying a noun_artifact of noun_substance" demonstrating a text generation process with enhanced contextual control over delexicalized placeholders through the strategic application of supersenses. To facilitate a comprehensive understanding of the proposed delexicalization methodologies, additional textual examples are provided in Table 3.2.

Furthermore, we have implemented a novel approach to data augmentation through

Table 3.2: Different flavors of delexicalization applied to the dataset referring to data transformations without and with supersense. (Transf. = Transformation; PN placeholders in blue; CN placeholders in green).

Transf. Type	**Lexicalized Text**	**Delexicalized Text**
Delex w/o SS	Brad Pitt is an actor.	Name_1 is a NOUN.
	The Mona Lisa hung above the antique table.	The Name_1 hung above the antique NOUN.
	Paris is a beautiful city.	City_1 is a beautiful NOUN.
	Noah and Sophia watched a movie at the local theater.	Name_1 and Name_2 watched a NOUN at the local NOUN.
Delex with SS	Brad Pitt is an actor.	Name_1 is a noun_person.
	The Mona Lisa hung above the antique table.	The Name_1 hung above the antique noun_artifact.
	Paris is a beautiful city.	City_1 is a beautiful noun_location.
	Noah and Sophia watched a movie at the local theater.	Name_1 and Name_2 watched a noun_communication at the local noun_artifact.

delexicalization by integrating both lexical and delexical aspects of the data. The primary objective of this methodology is to understand the critical role of semantic knowledge in identifying and preserving the syntactic structure of sentences when evaluating a delexicalized test set. This approach aims to enhance the model's ability to generate syntactically accurate sentences while maintaining semantic coherence.

To the best of our knowledge, this study represents the first attempt to explore data delexicalization in the context of neural DRS-to-Text generation. While preliminary research efforts have been directed towards data augmentation involving verbs and nouns in DRS-to-Text generation [Amin et al., 2024, 2022b], this investigation distinguishes itself as the first comprehensive study to focus on data augmentation through delexicalization with the explicit aim of evaluating the syntactic structure of the generated text through subsequent relexicalization. This novel approach provides valuable insights into the relationship between semantic content and syntactic structure in neural text generation.

3.4.1 Data Delexicalization with Proper Nouns

In our experimentation, we focused on two specific Named Entity (NE) categories for PNs: person names (PER), encompassing both male and female names, and geopolitical entities names (GPE), which include names for cities, states, countries, and islands. To facilitate the extraction of proper names from the textual data, we employed the spaCy named entity recognizer (https://spacy.io). Our analysis revealed a total of 3,773 instances of PNs within the PER and GPE categories. Further categorization of these PNs yielded the following distribution: person names constitute 57% of the total, city names account for 30%, state names represent 6%, country names comprise 6%, and other

types, such as island names, make up the remaining 1%. This categorical distribution of named entities closely aligns with the distribution observed in our data augmentation experiments (see Figure 3.1).

In the process of delexicalizing PNs, we have implemented two distinct approaches to replace named entities with placeholders. This methodology aims to analyze the impact on model generalizability by systematically removing lexical information from the dataset. The two approaches are as follows:

1. Custom Substitution: This method involves replacing named entities with custom placeholders, such as person_name, city_name, state_name, and country_name. This substitution process resulted in the creation of 6 distinct custom placeholders for all the named entities under observation in the dataset.

2. SpaCy-oriented Substitution: This approach utilizes the predefined placeholders from spaCy, specifically PER (for person) and GPE (for geopolitical entity). This substitution method resulted in only 2 placeholders for all named entities in the dataset. To explain these approaches, consider the sentence "Tom is living in Boston now". Applying the first approach yields "Name_1 is living in city_1 now", while the second approach produces "PER is living in GPE now".

Through our experimental investigations with delexicalized PNs, we discovered that custom delexicalization proves more effective compared to spaCy-oriented delexicalization. This superiority is particularly evident in the context of model evaluation through relexicalization. Custom delexicalization facilitates the preservation of true pragmatics within the logical input while maintaining accurate semantic correlations between delexicalized named entities. Conversely, when utilizing spaCy-defined placeholders, the model frequently encounters difficulties in accurately identifying the precise location of the named entity placeholder within the delexicalized translation of the meaning representation.

To illustrate this point, consider the sentence "Tom went to London and called Mary". Employing custom delexicalization yields "Name_1 went to City_1 and called Name_2", which presents a semantically more comprehensible structure for the neural model. In contrast, the spaCy-oriented placeholder approach may lead to confusion in the order of semantic entities, potentially generating erroneous outputs such as "PER went to PER and called GPE" or "GPE went to PER and called PER". Consequently, for all subsequent experiments in our study, we have opted to utilize custom delexicalization for named entities. Table 3.2 provides additional examples demonstrating the various delexicalization procedures employed in our research.

3.4.2 Data Delexicalization with Common Nouns

CNs that preserve the true contextual sense of a sentence are critical lexical entities, particularly in logical input representations such as DRS. To extract CNs from the textual data, we again employed the spaCy, which resulted in the identification of $6,193$ lexical entities within the dataset. For the purpose of delexicalization, we have implemented two distinct procedures for replacement:

1. SpaCy-oriented Substitution: This method involves replacing all lexical entities of CNs with a single spaCy-based placeholder, namely "NOUN". This type of delexical

substitution results in a fully generalized dataset, utilizing only one placeholder for all lexical entities classified as CNs.

2. Supersense-oriented Substitution: This novel methodology has proven beneficial in maintaining the categorical and contextual sense of the sentence. Utilizing supersenses, we categorized CNs according to the top 26 lexicographic categories defined in WordNet, based on the instances present in our data. These categories encompass a wide range of semantic fields, including noun-act, noun-artifact, noun-body, noun-cognition, noun-communication, noun-event, noun-feeling, noun-food, noun-group, and noun-motion, among others. To visually represent the distribution of CNs across these supersense categories, we have provided a graphical illustration in Figure 3.2. This categorical distribution of supersense entities closely aligns with the distribution observed in our data augmentation experiments.

3.5 Methodological Implementation for Neural DRS-to-Text Generation

The task of DRS-to-Text generation presents a significant challenge in the domain of logic-to-text generation, necessitating the use of computationally efficient neural models to transform logical representations into natural language. In our research, we have employed three distinct neural architectures to address this complex task. The first two models are founded on an encoder-decoder framework utilizing recurrent sequence-to-sequence neural networks, specifically incorporating two bi-directional LSTM layers [Hochreiter and Schmidhuber, 1997, Junczys-Dowmunt et al., 2018]. These models differ in their lexical encoding approaches: one employs character-based encoding (referred to as CB-bi-LSTM), while the other utilizes word-based encoding (designated as WB-bi-LSTM). Additionally, we have implemented a third model, which involves fine-tuning a byT5 variant from the Transformer family for the DRS-to-Text generation task (denoted as FT-byT5) [Xue et al., 2022].

We acknowledge that state-of-the-art DRS-to-text generation models typically employ sophisticated neural architectures [Liu et al., 2021, Wang et al., 2023a]. This awareness has motivated our decision to incorporate a Transformer-based model in our research. However, it is essential to emphasize that the primary objectives of this study are centered on analyzing the effects of data augmentation and data delexicalization within the context of neural DRS-to-text generation, rather than attempting to develop a system that achieves the highest performance metrics. Our focus lies in understanding the impact of data transformation techniques on the generation process.

It is crucial to highlight the fundamental distinctions between the CB-bi-LSTM and WB-bi-LSTM models, which primarily originate from their divergent approaches to input and output data representations. These differences exhibit in their handling of characters or words and their respective capabilities in managing out-of-vocabulary words. The character-based model (CB-bi-LSTM) demonstrates a notable advantage in seamlessly processing out-of-vocabulary words, as it operates on character sequences. Conversely, the word-based model (WB-bi-LSTM) may encounter challenges when con-

fronted with out-of-vocabulary words, as its effectiveness is depending upon the extent of its included vocabulary. We emphasize that these contrasting methodologies can significantly influence the impact of specific data transformation techniques.

The bi-LSTM model employed in our experiments utilizes word embeddings with a dimension of 300, providing a rich representation of input tokens. Both the encoder and decoder are constructed using LSTM cells, which are specifically designed to capture long-range dependencies in sequential data. The architecture incorporates two layers in both the encoder and decoder, allowing for more sophisticated input data representations. During training, the model processes mini-batches of 48 samples simultaneously, optimizing computational efficiency. A normalization rate of 0.9 is applied to enhance training stability, while a learning rate decay of 0.5 is implemented on an epoch basis to facilitate model convergence. The Adam optimizer is employed for parameter updates, leveraging its adaptive learning rate capabilities. Model performance is evaluated using cross-entropy as the validation metric, with the cost function being the mean cross-entropy loss. During the decoding phase, a beam search algorithm with a beam size of 10 is used to generate the most probable output sequences. The initial learning rate is set at 0.002, governing the step size in the optimization process. Table 3.3 lists LSTM-based hyperparameters used in our experiments.

Table 3.3: Hyperparameter settings for CB-bi-LSTM and WB-bi-LSTM.

HyperParameters	**Values**
Embedding Dimensions	300
Enc/Dec Cell	LSTM
Enc/Dec Depth	2
Mini-batch	48
Normalization Rate	0.9
lr-decay	0.5
lr-decay-strategy	Epoch
Optimizer	Adam
Validation Metric	Cross-Entropy
Cost-Type	ce-mean
Beam Size	10
Learning Rate	0.002

The FT-byT5 model, based on the T5 architecture, employs a distinct set of hyperparameters optimized for fine-tuning. This model processes batches of 15 samples in parallel, with parameter updates occurring every 8 steps. The learning rate is dynamically adjusted within a range of 1e-5 to 1e-4. To ensure stable training initiation, a warmup period of 3000 updates is implemented, during which the learning rate gradually increases from zero to its maximum value. Subsequently, the learning rate decays over 30000 steps, transitioning from its peak to its minimum value. The entire training process spans 15 epochs, constituting complete passes over the training dataset. For optimization, the FT-byT5 model employs the AdamW optimizer, an Adam variant that incorporates weight decay. This choice aids in preventing overfitting by implementing regularization on the model parameters, thereby enhancing the model's generalization

capabilities. Table 3.4 displays hyperparameters used in fine-tuning experiments with byT5.

Table 3.4: Hyperparameter settings for FT-byT5.

HyperParameters	**Values**
Batch size	15
Update steps	8
Max learning Rate	1e-4
Min learning Rate	1e-5
Warmup updates	3000
Max decay steps	30000
No. of epochs	15
Optimizer	AdamW

Our research utilizes the English version of the PMB dataset. Among the various dataset classifications available (gold, silver, and bronze), we have focused our efforts on the gold dataset, which represents the fully manually annotated and corrected version. The gold-PMB adheres to the standard dataset division protocol, comprising training, development, and testing files. These files contain 6 620, 885, and 898 data examples, respectively, providing a robust foundation for our experimental work.

In our data augmentation experiments, we have adopted two distinct approaches to transform examples. The first approach involves applying a single type of transformation and concatenating the result with the original data examples. This method yields an expanded dataset featuring one specific type of data transformation, such as PN or CN (denoted by the '+' sign in Table 3.6, Table 3.7 and subsequent Table 3.8). It is important to note that we have applied data augmentation exclusively to training examples, while the development and test files remain unchanged. The second approach entails applying multiple possible transformations (referred to as a blend) to each example. For instance, we might apply both proper noun and common noun augmentations to a single example (indicated by the '-' sign in Table 3.6, Table 3.7 and subsequent Table 3.8 in the following). This approach results in a smaller training set size compared to the first method, thereby emphasizing the role of transformations rather than the increased volume of training data. Table 3.5 provides a comprehensive overview of individual and blended data transformations, along with their corresponding training example sizes.

The pipeline described in Figure 3.5 illustrates an advanced data augmentation process for DRS-to-text generation, utilizing an RNN-based encoder-decoder model (in the case of bi-LSTM). This approach focuses on enhancing the diversity of PNs and CNs within the input data. The process begins with a DRS, which is transformed into a vector representation by the Encoder RNN. The Decoder RNN then converts this vector back into a sentence, completing the basic text generation cycle. The key innovation lies in the data augmentation step, where various PNs (e.g., "Brad Pitt", "Louis Olivia") and CNs (e.g., "actor", "performer") are systematically substituted within the DRS.

This substitution technique generates multiple variations of sentences from a single DRS input. For instance, a DRS containing "Brad Pitt is an actor" can yield augmented versions such as "Louis Olivia is an actor", "Brad Pitt is a performer", and "Louis Olivia

Table 3.5: Dataset size ('+' indicates augmentation applied in individual form and '-' in blended form).

Transformation Type	Size	Examples
Original Training Examples	x1	6620
Original + Proper Noun Augmentation	x2	13240
Original + Common Noun Augmentation	x2	13240
Original + Proper Noun-with-Common Noun Augmentation	x2	13240
Original + Proper Noun + Common Noun Augmentation	x3	19860
Validation Examples		885
Test Examples		898

is a performer". The primary objective of this augmentation is to significantly expand the lexical diversity of the training data. By exposing the model to a wider range of linguistic expressions that convey the same underlying meaning, it enhances the model's ability to generalize and interpret various lexical variations in the generated text. Consequently, this approach aims to produce a more robust and versatile DRS-to-text generation model, capable of handling a broader spectrum of inputs and generating more diverse outputs.

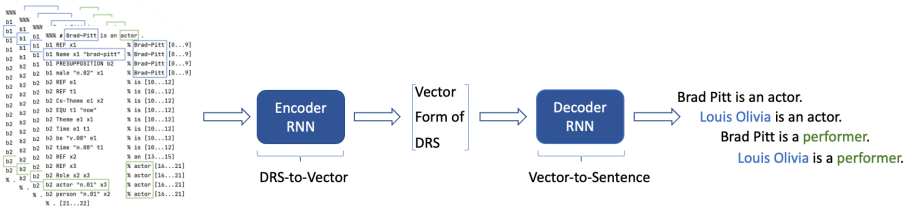


Figure 3.5: Implementation pipeline for DRS-to-Text generation.

In our research, we also employed two distinct methods of sample transformation to delexicalize the dataset. The first method referred to as delex1, involved the delexicalization of PNs using custom placeholders and CNs utilizing a spaCy-oriented placeholder approach. The second method, denoted as delex26, encompassed the delexicalization of PNs through custom placeholders and CNs using supersense-based placeholders. In our experimental setup, we applied delexicalization to the DRS-text pairs of both the training and development sets. However, we only delexicalized the DRS for the test set, maintaining the text in its fully lexical form. This strategic decision allows us to perform relexicalization on the model-generated text, enabling a comprehensive evaluation of model performance through comparison with the gold test set.

Figure 3.6 demonstrates the text generation process from structured data using a neural model, incorporating both pre-processing and post-processing steps to manage lexical entities. Initially, lexical entities such as PNs and CNs are replaced with placeholders during the pre-processing stage. This standardization allows the neural model to generate text without being biased by specific lexical items. The pre-processed input

is then fed into the neural model, which produces a sentence containing the generalized terms. In the post-processing step, these generalized terms are relexicalized to their original specific entities, ensuring that the final output text accurately reflects the intended meaning. For example, the generalized term noun_artifact in the generated text "The noun_artifact is opening now" is replaced with the original term door, resulting in the final output, "The door is opening now".

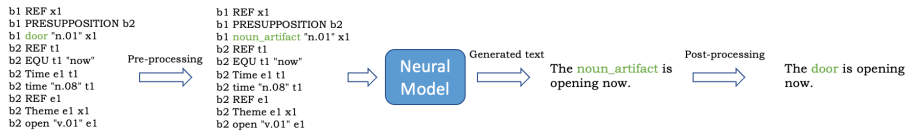


Figure 3.6: Implementation pipeline of DRS-to-Text generation through the pre-processing (delexicalization) and post-processing (relexicalization) of the text "The door is opening now".

Furthermore, a novel contribution also lies in the analysis of data augmentation through logical data delexicalization. Our objective is to understand the significant role played by data augmentation, with a particular focus on addressing the data-hungry nature of neural models. To this end, we have conducted four distinct experiments:

1. We concatenated fully lexical and delex1 data examples, subsequently evaluating model performance on the delex1 test set.

2. We combined fully lexical and delex26 data examples, assessing model performance on the delex26 test set.

3. We merged fully lexical, delex1, and delex26 data examples, then evaluated model performance on the delex1 test set.

4. Finally, we concatenated fully lexical, delex1, and delex26 data examples, evaluating model performance on the delex26 test set.

These comprehensive approaches allow for an in-depth understanding of the impacts of various delexicalization strategies on model performance.

3.6 Experimental Results for Data Augmentation

Our research involved a comprehensive series of experiments aimed at evaluating model performance through systematic modifications of lexical semantics-based input representations. We employed three distinct neural models, focusing on pre-training and fine-tuning sequence-to-sequence architectures. The experimental results, detailed in Table 3.6, Table 3.7, and Table 3.8, showcase the outcomes for pre-trained character-level and word tokenization-based bidirectional LSTM models, as well as the fine-tuned transformer-based byT5 model respectively. These experiments provide valuable insights into the impact of various input representation strategies on model performance across different neural architectures.

3.6.1 Augmentation Evaluation with Automatic Metrics

For automatic metric-based evaluations, we have used BLEU [Papineni et al., 2002], NIST [Jones et al., 2016], METEOR [Banerjee and Lavie, 2005], ROUGE [Lin, 2004], CIDEr [Oliveira dos Santos et al., 2021], and BERTScore [Zhang et al., 2020] as utilized in [Wang et al., 2021b, Amin et al., 2022b].

PN Augmentation: Our PN augmentation experiments were conducted in two contexts: inside and outside. Results are presented in Experiments 2-3 of Table 3.6, 11-12 of Table 3.7, and 20-21 of Table 3.8. For LSTM architectures, findings highlight the crucial role of vocabulary, particularly for the CB-bi-LSTM model, which demonstrates greater independence in sequence generation. This led to CB-bi-LSTM achieving the highest score in proper noun augmentation outside context (Exp.3). Conversely, the word-level decoder, being more vocabulary-focused, excelled in PN augmentation inside context (Exp.11). This underscores the effectiveness of the word-level models in generating coherent, grammatically correct outputs while accurately capturing input DRS syntax and semantics. The FT-byT5 model showed the best performance in PN augmentation (Exp.21), achieving the highest values across all metrics in all experiments. While this may be attributed to T5's original model characteristics, the key takeaway is that data augmentation can significantly enhance performance even in pre-trained LLMs.

Table 3.6: CB-bi-LSTM (Exp.01-09) with individual and blended PN and CN augmentation experiments. † shows that the model is statistically significant using *Wilcoxon Test* on all evaluation metrics scores w.r.t. the baselines (Exp.01). All experiments are an average of 5 runs.

Exp.	Implementation Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
01	Gold-PMB (no Aug)	47.72	7.68	39.42	72.59	4.84	95.3
02	Orig + PN (in ctx) Aug	51.37 †	7.96 †	41.19 †	74.78 †	5.15 †	95.8
03	Orig + PN (out ctx) Aug	53.16 †	8.11 †	42.00 †	75.30 †	5.27 †	95.9
04	Orig + CN (in ctx with SS) Aug	50.28 †	7.94	40.90 †	74.24 †	5.02 †	95.7
05	Orig + CN (in ctx w.o. SS) Aug	49.99 †	7.91	40.14 †	74.06 †	4.96 †	95.6
06	Orig + CN (out ctx with SS) Aug	50.89 †	7.98 †	40.70 †	74.38 †	5.08	95.7
07	Orig + CN (out ctx w.o. SS) Aug	50.63 †	7.93 †	40.39 †	74.33 †	5.06 †	95.7
08	Orig + PN (out ctx)-with-CN (out ctx with SS) Aug	52.51 †	8.06 †	41.23 †	75.28 †	5.24 †	96.0
09	Orig + PN (out ctx) + CN (out ctx with SS) Aug	**54.00** †	**8.19** †	**42.32** †	**76.15** †	**5.35**	**96.1**

CN Augmentation: For CN augmentation, we conducted four types of experiments: (1) inside context with SS, (2) inside context without SS, (3) outside context with SS, and (4) outside context without SS, as shown in Experiments 4-7 (see Table 3.6), 13-16 (see Table 3.7), and 22-25 (see Table 3.8). The importance of vocabulary observed in PN augmentation extends to CN as well. The CB-bi-LSTM decoder performed best for outside context with SS (Exp.6), while WB-bi-LSTM excelled in inside context with SS (Exp.13). The FT-byT5 model once again outperformed, showing the best results

for CN augmentation in outside context with SS (Exp.24) and consistently producing the highest scores across all common noun augmentation variations when compared to both LSTM models.

Experiments 8-9 in Table 3.6, 17-18 in Table 3.7, and 26-27 in Table 3.8 applied the most effective augmentation techniques for PNs and CNs (outside context for CB-bi-LSTM, inside context for WB-bi-LSTM, and outside context for FT-byT5) as both blended and individual data examples. In Experiments 8, 17, and 26, augmentation techniques were applied simultaneously to each input data example, creating blended examples (original + PN-with-CN). Conversely, Experiments 9, 18, and 27 applied these techniques separately and concatenated them as (original + PN + CN) augmentation data examples. Analysis of all experimental results revealed that LSTM models achieved their highest scores when applying the best augmentation flavors of PN and CN concatenated as separate individual training examples (Exp.9 and Exp.18). The FT-byT5 model also performed best with concatenated examples (Exp.27). Interestingly, unlike LSTM models, FT-byT5 did not achieve its best values in Exp.27 compared to Exp.21. This divergence in performance patterns between T5 and LSTM models is likely attributable to the unique characteristics of the original T5 model.

Table 3.7: WB-bi-LSTM (Exp.10-18) with individual and blended PN and CN augmentation experiments. ‡ shows that the model is statistically significant using *Wilcoxon Test* on all evaluation metrics scores w.r.t. the baselines (Exp.10). All experiments are an average of 5 runs.

Exp.	Implementation Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
10	Gold-PMB (no Aug)	32.91	5.80	29.99	61.39	3.49	94.4
11	Orig + PN (in ctx) Aug	44.37 ‡	7.37 ‡	36.56 ‡	69.54 ‡	4.38 ‡	95.1
12	Orig + PN (out ctx) Aug	42.70 ‡	7.16 ‡	35.39 ‡	67.69 ‡	4.18	94.9
13	Orig + CN (in ctx with SS) Aug	**44.41** ‡	7.28 ‡	36.22 ‡	68.78 ‡	**4.34** ‡	95.1
14	Orig + CN (in ctx w.o. SS) Aug	42.94 ‡	7.14 ‡	35.11 ‡	67.56 ‡	4.19	94.8
15	Orig + CN (out ctx with SS) Aug	41.84 ‡	6.97 ‡	34.25 ‡	66.38 ‡	4.05	94.6
16	Orig + CN (out ctx w.o. SS) Aug	42.41 ‡	7.13 ‡	35.01 ‡	67.47 ‡	4.16 ‡	94.8
17	Orig + PN (in ctx)-with-CN (in ctx with SS) Aug	43.78 ‡	7.21 ‡	35.87 ‡	68.52 ‡	4.27 ‡	95.0
18	Orig + PN (in ctx)+CN (in ctx with SS) Aug	44.39 ‡	**7.36** ‡	**36.63** ‡	**69.53** ‡	4.29 ‡	**95.2**

A comparison of the overall performance between CB-bi-LSTM and WB-bi-LSTM models reveals that CB-bi-LSTM consistently outperforms across all input data aspects, as anticipated. This superiority demonstrates the char-level model's proficiency in handling out-of-vocabulary words and its ability to effectively capture micro-level aspects and data patterns within the input DRS. The results highlight the char-level model's effectiveness and morphological accuracy in generating correct output sequences. Despite these strengths, it is noteworthy that the FT-byT5 model surpasses the performance of bi-LSTM-based models in the majority of experiments conducted.

In our final experiment (Exp.28), we conducted a preliminary evaluation of the impact of augmented data size. We replicated Experiment 21 while halving the size of the augmented portion of the training set. The resulting scores, which fell between those of the baseline and the best model, suggest a linear increase in performance relative

Table 3.8: FT-byT5 (Exp.19-28) with individual and blended PN and CN augmentation experiments. ◇ shows that the model is statistically significant using *Wilcoxon Test* on all evaluation metrics scores w.r.t. the baselines (Exp.19). All experiments are an average of 5 runs.

Exp.	Implementation Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
19	Gold-PMB (no Aug)	51.88	7.94	43.55	76.04	5.63	96.7
20	Orig + PN (in ctx) Aug	55.72 ◇	8.23 ◇	45.05 ◇	77.81 ◇	5.91 ◇	97.1
21	Orig + PN (out ctx) Aug	**57.15** ◇	**8.33** ◇	**45.90** ◇	**78.81** ◇	**6.08** ◇	**97.2**
22	Orig + CN (in ctx with SS) Aug	53.08	8.04	44.20	76.64	5.68	96.8
23	Orig + CN (in ctx w.o. SS) Aug	52.85	8.00	44.50	76.32	5.69	96.8
24	Orig + CN (out ctx with SS) Aug	54.71 ◇	8.13 ◇	44.77	77.27	5.84◇	97.0
25	Orig + CN (out ctx w.o. SS) Aug	52.78	8.02	44.29	76.31	5.66 ◇	96.8
26	Orig + PN (out ctx)-with-C.N. (out ctx with SS)	52.89	8.03	44.68	76.60	5.76	96.9
27	Orig + PN (out ctx) + C.N. (out ctx with SS) Aug	53.34	8.02	44.60	77.05	5.71	96.9
28	Orig + half PN (out ctx) (randomly sampled) Aug	53.42	8.04	44.44	76.50	5.74	97.0

to the size of the augmented training set. However, this hypothesis requires further experimentation for verification. To ensure the statistical validity of our results, we employed a *Wilcoxon Signed Rank Test*, as described by [Dror et al., 2018].

3.6.2 Comparing Augmented Models with LLMs

To gain preliminary insights into our approach's performance relative to general-purpose LLMs, we compared our neural DRS-to-Text systems with two recent LLMs: ChatGPT-3.5 [OpenAI, 2023] and Claude-2.0 [Turpin et al., 2023]. These LLMs were not fine-tuned for our specific task. We employed both zero-shot and few-shot learning techniques to analyze their performance in comparison to our specialized models.

For performance analysis, we used a sample of 215 sentences from the test set. We applied these to our best neural DRS-to-Text models (CB-bi-LSTM and FT-byT5) and provided data as prompts to ChatGPT-3.5 and Claude-2.0. We then evaluated the outputs using automatic evaluation metrics (results in Table 3.9). The experimental results clearly demonstrate that general-purpose LLMs underperform in this domain-specific task. This underscores the necessity for task-specific neural models in DRS-to-Text generation.

Robust Overall Semantic Evaluation (ROSE): Our final evaluation involved two expert human evaluators who assessed the generated text by analyzing model-generated systematic errors in semantics, grammaticalization, and phenomenon coverage. They produced a ROSE score for each output. Table 3.12 presents notable examples generated by our best augmentation model, "byT5 PN Aug". As defined by [Wang et al., 2021b], the ROSE score is a conjunction of three binary (0-1) evaluation scores: (1) Semantic preservation, (2) Grammatical correctness, and (3) Phenomenon coverage. A text receives a score of 1 if it passes all three criteria, and 0 otherwise. Table 3.10 reports the average ROSE scores for a 100-sentence sample from the test set. This evaluation

Table 3.9: Evaluation of DRS-to-Text by LLMs reporting scores for the baseline (without augmentation), ChatGPT-3.5, Claude-2.0, and our best (augmented) models.

Model Type	Data Type	BLEU	NIST	METEOR	ROUGE	CIDEr	BERT Score
CB-bi-LSTM	Gold without Aug	45.42	6.43	38.42	71.70	4.75	95.4
	PN Aug	50.64	6.69	40.67	74.22	5.22	95.9
	CN Aug	48.70	6.70	39.67	73.38	5.03	95.7
Claude-2.0	zero-shot	11.33	3.05	29.39	42.43	1.69	92.3
	few-shot	27.25	5.39	38.58	64.25	3.51	95.3
ChatGPT-3.5	zero-shot	9.82	2.63	27.91	39.80	1.59	91.9
	few-shot	9.58	2.51	26.01	37.40	1.53	91.5
byT5	Gold without Aug	47.55	6.46	42.90	74.56	5.49	96.5
	PN Augmentation	**54.28**	**6.86**	**45.81**	**78.25**	**5.96**	**97.1**
	CN Augmentation	53.04	6.73	45.21	76.97	5.90	96.9

further validates the quality of our best augmentation model in producing high-quality texts, as it also achieved the best results in the ROSE measure.

Table 3.10: Expert evaluation based on Semantics, Grammatical Structure, and Phenomenon for the baseline (byT5 without augmentation), LLMs (ChatGPT and Claude), and our best (augmented) models byT5 PN and byT5 CN. All scores are listed in (%).

Implementation	Semantics	Grammatical	Phenomenon	ROSE
byT5 wo Aug	45	86	48	43
ChatGPT-3.5	44	62	23	13
Claude-2.0	25	84	61	23
byT5 CN Aug	49	90	64	49
byT5 PN Aug	**57**	**92**	**65**	**55**

For instance, in Table 3.11, the generated text "I'm milking my squirrel" correctly scores a 1 in both grammar and phenomena, but receives a 0 in the semantic measure because of a meaning error. Nevertheless, the semantic failure is reflected in the ROSE score of 0. On the other hand, instances like "We arrived two days ago" and "Three times five is fifteen" have a perfect ROSE score of 1, indicating that they are accurate in all assessed dimensions. This examination highlights the model's advantages and disadvantages in terms of producing text that complies with grammatical and semantic criteria similar to those of humans.

Table 3.11: Analysis of byT5 PN augmentation model-generated examples w.r.t reference text for ROSE evaluation.

Ex.	Reference Text	Generated Text	Sem.	Gram.	Phen.	ROSE
1	I am milking my goat.	I'm milking my squirrel.	0	1	1	0
2	Susan shined your father's shoes.	Susan shined your father's.	0	0	1	0
3	We arrived two days ago.	I arrived two days ago.	1	1	1	1
4	Three times five is fifteen.	3 times 5 is 15.	1	1	1	1
5	An elephant has a long nose.	The elephant has a long nose.	1	1	1	1

3.6.3 Error Analysis of Augmentation Results

To gain deeper insights into the evaluations, we conducted a manual inspection of the model-generated text, comparing examples produced by different models: those without augmentation, LLMs (ChatGPT and Claude), and the byT5 model with PN augmentation — our best model. Table 3.12 showcases examples illustrating various aspects such as semantics, grammar, missing information, LLM hallucinations, extra information, and perfectly generated examples. We evaluated the model-generated text based on three criteria: (1) wrong information (marked in red), (2) extra irrelevant information (underlined), and (3) semantically correct but differently expressed information (in blue).

Table 3.12: Error analysis of different byT5 P.N. augmented model-generated examples w.r.t. reference text.

Reference Text	Model Type	Model Generated Text
You can't live on that island.	Without Aug	Everybody can't live in the island.
	ChatGPT-3.5	If a person does not live on an island, it is possible.
	Claude-2.0	If x1 is a person, then x1 does not live on an island, which is presumed.
	With Aug	You cannot live on the island.
It will cost around 10000 yen.	Without Aug	It will cost about 1000 yen.
	ChatGPT-3.5	The cost of the entity referred to by x1 is presupposed to be now and is equal to 10,000 yen per unit.
	Claude-2.0	An entity costs 10000 yen now.
	With Aug	It will cost about 10 thousand yen.
Have you googled her?	Without Aug	You googled her.
	ChatGPT-3.5	The hearer (you), who is a female, is currently googling something at the present time.
	Claude-2.0	The hearer is currently googling a presumed female.
	With Aug	Have you googled her?

Table 3.12 highlights three crucial aspects of natural language generation: negation, question formation, and quantity expression. The model without augmentation struggled to capture the true semantics of sentences (with completely wrong semantics highlighted in red). It also faced challenges in identifying exact quantities and maintaining correct grammatical structures, as evident from the examples provided in Table 3.12.

ChatGPT and Claude underperformed in this task, failing to generate exact translations for the DRS examples. The examples reveal that these models tended to explain the logical representation of the DRS rather than producing precise translations (irrelevant text is underlined). We attribute this to the lack of semantic/formal meaning representations in the training data of these LLMs. Our manual inspection focused on examples from the best-performing configurations: few-shot text for Claude and zero-shot text for ChatGPT (as shown in the LLM results in Table 3.9 for few-shot and zero-shot approaches).

Our best augmentation model successfully captured semantic and grammatical representations, though it occasionally struggled to replicate information exactly as listed in the test set. These minor alterations (highlighted in blue) in the model-generated text do not significantly impact human evaluation, as the generated text maintains the exact meaning, semantics, and grammatical structure of the sentences. However, these slight differences may result in lower scores in automatic evaluations due to the reliance on

exact word overlap between text pairs.

3.7 Experimental Results for Data Delexicalization

Our research consisted of a thorough series of experiments designed to assess model performance by systematically transforming input representations based on lexical semantics. We utilized three different neural models, concentrating on the pre-training and fine-tuning of sequence-to-sequence architectures. The experimental findings, presented in Table 3.13, Table 3.14, and Table 3.15, highlight the performance of pre-trained character-level and word tokenization-based bidirectional LSTM models, as well as the fine-tuned transformer-based byT5 model. These experiments offer significant insights into how different input representation strategies influence model performance across various neural architectures.

3.7.1 Delexicalization Evaluation with Automatic Metrics

To facilitate a clear understanding of the results presented in Table 3.13, Table 3.14, and Table 3.15, we have organized the evaluation scores into four distinct blocks. The first block (exp.01) represents our baseline, showing DRS-to-text generation results for models trained or fine-tuned on a fully lexical dataset without delexicalization. The second block (exp.02-03) displays results for two delexicalization approaches: with supersenses (exp.02) and without supersenses (exp.03), where models are pre-trained or fine-tuned solely on delexicalized data. The third block presents results for data augmentation through delexicalization, combining fully lexical data with one type of delexicalization - either with supersenses (exp.04) or without (exp.05). The fourth block showcases compound augmentation results, concatenating fully lexical and delexicalized data examples both with and without supersenses. We evaluate these final training examples on two different test sets: a delexicalized test set with supersenses (exp.06) and one without supersenses (exp.07).

Table 3.13: CB-biLSTM results for delexicalization with supersenses (delex26) and without supersenses (delex1) on Gold-PMB dataset. (Note: MET. = METEOR; RUG. = ROUGE; CMT. = COMET; B.Scr = BERTScore; CB = Character-Based; tst = testing)

Exp.	Implementation Type	BLEU	chrF	MET.	RUG.	CMT.	B.Scr
CB-01	Fully Lexical	46.80	66.22	39.12	72.54	79.33	95.31
CB-02	Delex26	48.51	63.58	40.34	74.24	75.37	94.67
CB-03	Delex1	51.10	61.24	40.80	74.43	74.11	94.26
CB-04	Lex+delex26	60.45	**71.12**	46.46	80.78	82.48	96.19
CB-05	Lex+delex1	57.68	69.85	44.33	78.62	81.63	95.94
CB-06	Lex+delex26+delex1(tst delex26)	60.95	70.52	46.25	80.70	81.94	96.10
CB-07	Lex+delex26+delex1(tst delex1)	**61.38**	70.66	**46.53**	**81.41**	**82.60**	**96.20**

Table 3.13 presents results for the pre-training of the bi-LSTM model using character-based tokenization (CB-biLSTM). Compared to the baseline (CB-01), CB-biLSTM outperforms in all experimental aspects involving delexicalization (with and without su-

persenses) and augmentation. Data augmentation consistently improves model performance by enhancing generalization capabilities. For individual data delexicalization (CB-02, CB-03), the approach without supersenses performs better (CB-03, in italics). In data augmentation scenarios, delexicalization with supersenses shows significant improvement (CB-04 in italics) compared to delexicalization without supersenses (CB-05). With compound augmentation, the CB-biLSTM model generalizes more effectively for a test set without supersenses (CB-07, in bold and italics), which also represents the highest score across all CB-biLSTM experimental formats.

Table 3.14: WB-biLSTM results for delexicalization with supersenses (delex26) and without supersenses (delex1) on Gold-PMB dataset. (Note: MET. = METEOR; RUG. = ROUGE; CMT. = COMET; B.Scr = BERTScore; WB = Word-Based; tst = testing)

Exp.	Implementation Type	BLEU	chrF	MET.	RUG.	CMT.	B.Scr
WB-01	Fully Lexical	40.36	56.06	33.42	65.26	73.66	94.44
WB-02	Delex26	52.32	62.77	41.67	75.37	76.94	94.95
WB-03	Delex1	49.80	58.47	40.33	73.32	73.94	94.45
WB-04	Lex+delex26	56.49	**67.90**	**44.24**	**78.73**	80.93	95.74
WB-05	Lex+delex1	53.98	65.98	41.99	75.85	80.17	95.75
WB-06	Lex+delex26+delex1(tst delex26)	57.05	67.11	44.00	78.16	**80.94**	95.87
WB-07	Lex+delex26+delex1(tst delex1)	**57.07**	67.38	44.11	78.42	80.59	**95.76**

Table 3.14 illustrates results for the WB-biLSTM model pre-trained with delexicalized and augmented training examples, both with and without supersenses. Similar to the CB-bi-LSTM model, WB-bi-LSTM outperforms the baseline (WB-01) in all experimental implementations. However, WB-bi-LSTM exhibits significantly different results compared to CB-biLSTM. For individual delexicalization (WB-02, WB-03), the model trained on supersense-based delexicalized data performs better (WB-02, in italics). Interestingly, compound data augmentation appears less effective for WB-biLSTM, as the model achieves the highest scores (except for BLEU) with data augmentation using supersense-based delexicalized data (WB-04 in bold and italics). The highest BLEU score, however, is observed in compound augmentation when testing the subset with delexicalized data without supersenses (WB-07).

Table 3.15 presents the results for the byT5 model fine-tuned on delexicalized data, both with and without supersenses. The adoption of data delexicalization generally enhances the model's overall generalization ability. Augmentation further improved performance, with the best results achieved through compound augmentation when testing the delexicalized subset without supersenses (T5-07 in bold and italics). The impact of individual delexicalization procedures (with and without supersenses, T5-02 and T5-03 respectively) on the model's generalization capacity is not significant, as the results are very similar. Likewise, in data augmentation scenarios (T5-04, T5-05), different delexicalization approaches combined with lexical augmentation do not lead to substantial improvements in model performance.

When comparing the overall performance across all models, byT5 demonstrates superior results from a fine-tuning perspective (see Table 3.15, T5-07) compared to the pre-training-based biLSTM models (see Table 3.13 and Table 3.14). This underscores the importance of using state-of-the-art sequence-to-sequence models for complex, task-

Table 3.15: FT-byT5 results for delexicalization with supersenses (delex26) and without supersenses (delex1) on Gold-PMB dataset. (Note: MET. = METEOR; RUG. = ROUGE; CMT. = COMET; B.Scr = BERTScore; T5 = fine-tuned byT5; tst = testing)

Exp.	Implementation Type	BLEU	chrF	MET.	RUG.	CMT.	B.Scr
T5-01	Fully Lexical	51.88	73.16	43.55	76.04	86.89	96.74
T5-02	Delex26	62.44	77.93	47.50	82.18	89.47	97.47
T5-03	Delex1	62.73	77.85	47.41	81.76	88.96	97.37
T5-04	Lex+delex26	62.94	78.48	47.67	82.20	89.72	97.40
T5-05	Lex+delex1	62.72	78.09	47.30	82.06	88.95	97.43
T5-06	Lex+delex26+delex1(tst delex26)	63.54	78.85	47.91	82.47	89.72	97.60
T5-07	Lex+delex26+delex1(tst delex1)	**64.22**	**78.87**	**48.07**	**82.90**	**90.16**	**97.63**

specific applications such as DRS-to-text generation. In the subsequent sections comparing the neural DRS-to-Text generation model with LLMs (Section 3.7.2) and for error analysis (Section 3.7.3), we will utilize the text generated by our best-performing model, FT-byT5, evaluated on the test set without supersenses (see Table 3.15, T5-07).

3.7.2 Comparing Delexicalized Models with LLMs

In our evaluation, we sought to compare the effectiveness of our specialized delexicalized neural approach against widely used LLMs not specifically tailored for this task. We assessed the quality of text generated by our neural DRS-to-text systems alongside two prominent LLMs: ChatGPT 3.5 [OpenAI, 2023] and Claude 2.0 [Turpin et al., 2023]. Our investigation encompassed both few-shot and zero-shot prompting methodologies. Interestingly, the performance of ChatGPT remained consistent across both approaches, while Claude demonstrated significant improvement when utilizing few-shot learning compared to zero-shot techniques (as evidenced in Table 3.16).

Table 3.16: Evaluation of DRS-to-Text generation text for LLMs reporting scores for ChatGPT 3.5, Claude 2.0, the baseline (without delexicalization), and our best (FT-byT5) model. (Note: LLM = Large Language Model; MET. = METEOR; RUG. = ROUGE; CMT. = COMET; B.Scr = BERT Score)

LLM Type	Implementation Type	BLEU	chrF	MET.	RUG.	CMT.	B.Scr
Claude-2.0	Zero-shot learning	11.33	44.15	29.39	42.43	69.83	92.31
	Few-shot learning	27.25	58.72	38.58	64.25	87.17	95.37
ChatGPT-3.5	Zero-shot learning	9.82	43.69	27.91	39.80	68.80	91.98
	Few-shot learning	9.58	40.46	26.01	37.40	66.17	91.54
byT5	Fully lexical model	47.55	71.47	42.90	74.56	86.49	96.52
	FT-byT5 (our best model)	**61.00**	**75.96**	**45.70**	**80.02**	**88.52**	**97.29**

Our analysis focused on a subset of 215 examples from the test set, which were subjected to two evaluations: (1) using our top-performing DRS-to-text neural model, FT-byT5 (referenced in Table 3.15), and (2) using prompts for Claude 2.0 and ChatGPT 3.5 to generate comparable text outputs. We then employed automatic evaluation metrics to assess the quality of the generated text (results presented in Table 3.16). In

our findings, italicized scores denote the best results within each model category, while bold and italicized scores indicate superior performance across all model types, with our FT-byT5 model achieving this distinction. The empirical evidence strongly suggests that for complex, domain-specific applications like DRS-to-text generation, task-specific neural models are essential, as general-purpose LLMs, despite their broad capabilities, fall short in this specialized context.

3.7.3 Error Analysis of Delexicalization Results

To gain a deeper insight into our understanding of the experimental results, we conducted a manual inspection of the text generated by various models. This comparison involved outputs from fully lexical models, LLMs (specifically ChatGPT and Claude), and our FT-byT5 model. Table 3.17 presents examples illustrating various semantic and syntactic features, instances of missing or additional information, LLM hallucinations, and cases of flawless generation. Our evaluation criteria for the generated text included: (1) inaccurate information (marked in red), (2) extraneous, irrelevant content (underlined), and (3) semantically accurate but differently phrased representations (in blue).

Table 3.17: Error analysis of different model-generated examples with reference text.

Reference Text	**Model Type**	**Model Generated Text**
You can't live on that island.	Fully Lexical	Everybody can't live in the island.
	ChatGPT-3.5	If a person does not live on an island, it is possible.
	Claude-2.0	If x1 is a person, then x1 does not live on an island, which is presumed.
	FT-byT5	You cannot live on that island.
It will cost around 10000 yen.	Fully Lexical	It will cost about 1000 yen.
	ChatGPT-3.5	The cost of the entity referred to by x1 is presupposed to be now and is equal to 10,000 yen per unit.
	Claude-2.0	An entity costs 10000 yen now.
	FT-byT5	It will cost around 10000 yen.
Have you googled her?	Fully Lexical	You googled her.
	ChatGPT-3.5	The hearer (you), who is a female, is currently googling something at the present time.
	Claude-2.0	The hearer is currently googling a presumed female.
	FT-byT5	Have you googled her?

Table 3.17 further highlights three crucial aspects of natural language generation: negation, interrogatives, and quantification. The fully lexical model struggled to accurately capture the semantic information within the given examples (with completely inaccurate semantics highlighted in red). Additionally, this model faced challenges in precisely determining quantities and maintaining proper grammatical structures, as evidenced by the examples provided in Table 3.17.

ChatGPT and Claude both exhibited suboptimal performance in accurately translating the DRS examples. Analysis revealed that instead of providing exact translations, these models tended to explain the logical structure of the DRS (with additional irrelevant text underlined). We attribute this behavior to the absence of semantic or formal meaning representations in their training data. Moreover, the few-shot learning approach did not significantly enhance their generalization capabilities. For our manual

inspection of LLM-generated text, we selected samples from the most effective models — few-shot results for Claude and zero-shot results for ChatGPT (refer to Table 3.16 for LLM performance in few-shot and zero-shot scenarios).

Our top-performing model, while not achieving perfect replication of the test set information, demonstrated the most effective capture of semantic and grammatical representations. The minor variations in the model-generated text (highlighted in blue) are unlikely to impact human evaluation due to substantial word overlap between text pairs. However, these slight differences, despite preserving the intended meaning, semantics, and grammatical structure, result in lower scores in automatic evaluations due to the lack of exact matches.

3.8 Chapter Conclusion

In this chapter, we conducted a comprehensive investigation into data transformation techniques for neural DRS-to-Text generation, specifically focusing on data augmentation and data delexicalization. Utilizing the PMB dataset, we explored novel approaches to enhance the performance and generalization capabilities of various neural models. Our data augmentation experiments involved enriching lexical information in DRS for proper and common nouns while maintaining contextual similarity through the supersense approach on different in-context and out-of-context transformations. The results demonstrated significant improvements across character-level, word-level, and transformer-based models, validating the effectiveness and reliability of our proposed augmentation methods. Complementing these findings, our data delexicalization experiments employed WordNet supersenses and named entity-based lexical abstractions for common and proper nouns in the DRS. This approach, both independently and in conjunction with lexical data augmentation, enhanced the generalization abilities of the neural models and improved overall performance. Notably, our experiments with biLSTM and byT5 neural sequence-to-sequence models yielded promising results, with the fine-tuned byT5 model achieving the highest scores. A key observation from the delexicalization study was that this technique enabled the models to focus more effectively on the syntactic structure of complex meaning representations, leading to the generation of more accurate textual sequences. This is the reason that data augmentation applied through data delexicalization has the highest performance among all the experiments conducted through data augmentation and data delexicalization. Interestingly, we found that general-purpose LLMs such as ChatGPT and Claude tended to hallucinate and explain the DRS rather than generating correct textual sequences, underscoring the importance of task-specific models for complex domain-specific applications.

Limitations: Our data augmentation and data delexicalization approaches primarily stem from the reliance on WordNet-based supersenses, which, in our implementation, are restricted to nouns. While WordNet provides 26 supersense categories for nouns, it offers only 15 categories for verbs, and even fewer, often inconsistent categories for adjectives and adverbs. This limited categorization, particularly for verbs, adjectives, and adverbs, constrains our ability to generalize supersense-based transformations across all lexical categories present in the DRS in the PMB dataset. Additionally, supersense

categories for adjectives and adverbs do not originate from WordNet itself, creating further limitations in our application scope for these parts of speech.

Moreover, supersenses are strongly tied to English through the structure of the WordNet lexicon, which restricts the applicability of our method to other languages like Italian and Urdu. These languages lack comprehensive lexical resources and supersense taxonomies equivalent to English WordNet. This limitation hinders our ability to implement supersense-based data transformations effectively across multilingual DRS datasets, thereby impacting the potential for cross-linguistic generalization in DRS-to-text generation tasks.

Chapter 4

Improving Semantic Parsing and Text Generation through Multi-Faceted Multi-Lingual Data Augmentation: Working with Variable Free DRS

The advancement of language models in NLP necessitates the development of robust and generalizable data augmentation techniques, particularly in the domain of structured input representations. This chapter investigates innovative multi-faceted and multilingual data augmentation methodologies across three typologically diverse languages: English, Italian, and Urdu. The proposed approaches address the inherent complexities associated with structurally intricate inputs such as DRS, offering novel solutions in both semantic parsing (Text-to-DRS) and text generation (DRS-to-Text). We present a comprehensive augmentation framework that incorporates named entity augmentation, WordNet-based lexical substitutions, and grammatical transformations, each precisely adapted to the specific linguistic characteristics of the target languages.

In the context of English, our methodology expanded the PMB dataset by a factor of nine, resulting in significant performance improvements across multiple language models. Semantic parsing SMATCH scores increased by over 23%, while substantial gains were observed in BLEU, METEOR, ROUGE, COMET, chrF, and BERT scores for text generation tasks. For Italian, the application of external linguistic resources to augment low-resource data led to a 14% improvement in SMATCH scores, with comparable enhancements across other evaluation metrics. In the case of Urdu, where no annotated corpora previously existed, we developed a novel rule-based alignment method to transform English DRS into Urdu, complemented by lexical, grammatical, and named entity augmentations. This approach resulted in a substantial increase in training data and significant improvements in model performance across all key metrics.

The empirical evidence from this study demonstrates that strategically designed data augmentation techniques not only enhance the robustness of neural models but also facilitate the application of DRS in both high- and low-resource linguistic contexts. These findings provide a scalable approach for semantic processing tasks in multilingual environments, contributing to the broader goal of developing more versatile and effective NLP systems.

Chapter adapted from

1. Amin, Muhammad Saad, Luca Anselma, and Alessandro Mazzei, "Improving Semantic Parsing and Text Generation through Multi-Faceted Data Augmentation", submitted to IEEE Access.

2. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei, "Data Augmentation for Low-Resource Italian NLP: Enhancing Semantic Processing with DRS", in Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), Pisa, Italy, December 4-6, 2024 (F. Dell'Orletta, A. Lenci, S. Montemagni, and R. Sprugnoli, eds.), vol. 3878 of CEUR Workshop Proceedings, CEUR-WS.org, 2024.

3. Muhammad Saad Amin, Xiao Zhang, Luca Anselma, Alessandro Mazzei, and Johan Bos, "Semantic Processing for Urdu: Corpus Creation, Parsing, and Generation", Submitted to Language Resources and Evaluation.

4. Muhammad Saad Amin, Alessandro Mazzei, Luca Anselma, "Towards data augmentation for DRS-to-text generation", Proceedings of the Sixth Workshop on Natural Language for Artificial Intelligence (NL4AI 2022) co-located with 21st International Conference of the Italian Association for Artificial Intelligence (AI*IA 2022), Udine, Italy. CEUR WORKSHOP PROCEEDINGS, vol. 3287, pp. 141-152. CEUR-WS, 2022.

4.1 Introduction

Recent advancements in NLP, particularly in semantic parsing and text generation, have been largely driven by parallel corpora that align text with semantic representations such as DRS or AMR [Basile and Bos, 2011, Banarescu et al., 2013]. While these resources have enabled the development of robust models for high-resource languages like English [Yih et al., 2014, Zhong et al., 2020], low and mid resource languages continue to face significant challenges due to the scarcity of annotated datasets and language-specific complexities [Li et al., 2016, Abzianidze et al., 2020].

This chapter focused on the Simplified Box Notation (SBN), a variable-free representation of DRS (see Chapter 2, Section 2.1.3), which is applied to both semantic parsing and text generation tasks across English, Italian, and Urdu. By leveraging comprehensive data augmentation strategies, including lexical, grammatical, and named entities, we aim to address resource limitations for these languages. SBN offers a simplified yet powerful framework for handling a wide range of linguistic phenomena, making it well-suited for multilingual applications. Figure 4.1 displays multi-lingual textual representations for the text "I also like cake." for corresponding language-neutral DRS representations. The evaluation of DRS from Box notation (Figure 4.1(b)) to clause notation (Figure 4.1(c)) and then from clause notation to variable-free notation (SBN) (Figure 4.1(d)) is displayed.

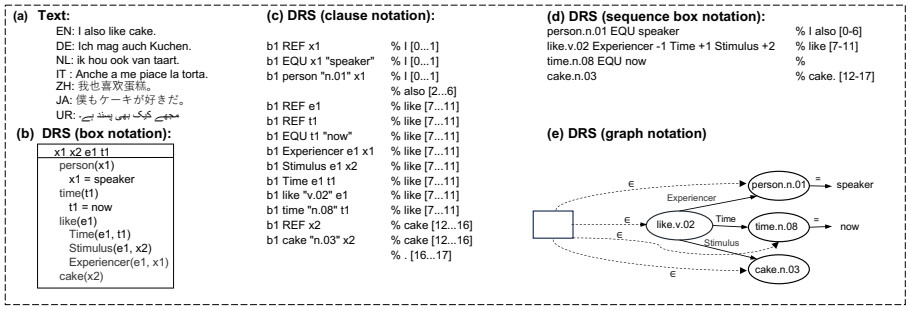


Figure 4.1: Different graphical representations of multi-lingual DRS for the text "I also like cake."

The ability to modify DRS-based meaning representations in a controlled manner is essential for evaluating the robustness of semantic parsers and generation systems [Wang et al., 2021a, Amin et al., 2024]. Systematically altering certain aspects of the DRS, such as entities, relations, or temporal expressions, how well these models handle variations can be assessed without altering the intended meaning. These controlled variations help to ensure that models are not overly sensitive to small changes and can generalize effectively across diverse input structures. It serves as a reliable test for the model's ability to process different forms of the same underlying meaning. Additionally, controlled modifications to DRS representations simulate its natural linguistic variability, allowing for the evaluation of how well systems manage different linguistic inputs that convey the same meaning. This is especially important for multilingual models that need to handle a wide range of linguistic phenomena across different languages, such as

varying grammatical structures or cultural nuances.

Moreover, controlled modification also plays a key role in error diagnosis. By isolating specific features or patterns within the DRS, problematic areas in parsing or generation can be identified easily. For example, if a model consistently fails when dealing with negation or complex noun phrases, controlled changes to the DRS can help pinpoint these weaknesses, facilitating more targeted improvements. In low-resource scenarios, controlled DRS modifications support data augmentation by generating diverse examples from limited datasets. This allows models to train on a broader spectrum of language without requiring extensive new annotations. Ultimately, this approach enhances the system's ability to generalize and increases robustness in real-world applications where language variability is the norm.

LLMs may unintentionally introduce noise during augmentation, particularly if they lack prior knowledge of the specific meaning representations [Jaszczolt and Jaszczolt, 2023, Imamura and Sumita, 2018, van Noord et al., 2019], potentially leading to examples that diverge from factual truth or grammatical correctness. The difficulties of applying data augmentation techniques are further compounded when dealing with languages that differ significantly from English in structure and resources. Italian, for instance, presents specific difficulties due to its flexible word order, complex verb conjugations, and the presence of grammatical gender [Jaszczolt and Jaszczolt, 2023]. These linguistic features, combined with the limited availability of semantically annotated corpora, position Italian as a mid-resource language in the specific domain of semantic processing [van Noord et al., 2018]. Traditional augmentation techniques, such as synonym replacement or back-translation, demonstrate reduced efficacy for Italian, necessitating innovative cross-lingual approaches that leverage resources from high-resource languages while preserving the linguistic integrity of the Italian language [Feng et al., 2021].

Low-resource languages like Urdu face additional hurdles. Despite being spoken by millions, Urdu lacks comprehensive semantic resources comparable to those available for English and other high-resource languages [Bögel et al., 2008]. The limited availability of parallel corpora, challenges in data annotation due to Urdu's calligraphic script, and the absence of detailed lexical ontologies significantly obstruct the creation of robust NLP systems for Urdu [Ahmed and Hautli, 2010, Hautli and Butt, 2011]. To address this gap, our work focuses on developing the "Urdu Meaning Bank" (UMB), the first semantic resource for Urdu that pairs sentences with formal meaning representations. We employ a combination of cross-lingual adaptation, machine translation, and data augmentation to build and enrich this resource, considering the unique morphosyntactic features of Urdu, such as verb conjugations, grammatical gender, and diglossic contextual dependencies.

This study explores the application of data augmentation techniques to semantic parsing and text generation across multiple languages, including English, Italian, and Urdu. Our objective is to enhance the robustness and adaptability of NLP models by generating diverse and contextually accurate training data, despite the linguistic challenges and resource limitations present in each language.

4.1.1 Research Objectives and Contributions

Our primary research objective is to explore and enhance the robustness of neural semantic parsing and text generation across three languages — English, Italian, and Urdu — through the development and evaluation of innovative data augmentation methodologies. We focus on augmenting data with modifications to named entities, lexical categories, and grammatical structures to improve the quality and effectiveness of semantic processing tasks. Specifically, our approach involves:

1. **English:** For English we investigated data augmentation techniques involving named entities (PER and GPE), lexical categories (such as nouns, verbs, adjectives, adverbs), and grammatical structures (such as tenses). Our methodology incorporates WordNet-based lexical substitutions for common nouns, verbs, adjectives, and adverbs, as well as grammatical transformations to generate new training sentences while preserving semantic integrity. We examined the impact of these techniques on the performance of neural semantic parsing and text generation models and explored the influence of external lexical knowledge on model functionality.

   This chapter extends the previous works on data augmentation (discussed in Chapter 3) by employing more systematic multi-faceted approaches (see Section 4.3) beyond the use of supersenses. Specifically, it transitions from the DRS clause format (see Chapter 2, Section 2.1.2) to the variable-free representation known as SBN (see Chapter 2, Section 2.1.3) and addresses both semantic parsing and text generation tasks in a multilingual context. Graphically the difference between DRS clause format (a) and variable-free format (b) is shown in Figure 4.2 for the text "Bill did not commit the crime."

2. **Italian:** For Italian, our study introduced a novel cross-lingual augmentation methodology that leverages English WordNet to enhance Italian semantic datasets. We provided empirical evidence of the effectiveness of this technique in improving performance scores for both DRS parsing and generation tasks in Italian. Additionally, we conducted a detailed analysis of how cross-lingual augmentation influences the handling of Italian-specific linguistic features, such as flexible word order, verb conjugations, and grammatical gender. This research also offers insights into the scalability of this approach to other low-resource languages within the semantic NLP domain.

3. **Urdu:** For Urdu, we aimed to develop the very first semantic corpus, freely available for research purposes. Our work involved developing semantic parsing and text generation models for Urdu and investigating effective data augmentation strategies tailored to Urdu's unique linguistic characteristics. We demonstrated the usefulness of these augmentation techniques in enhancing the generalization power of parsing and generation models for Urdu. Our evaluation compared Urdu's semantic processing performance to other languages, shedding light on the specific challenges and opportunities presented by Urdu in the context of semantic NLP.

```
b1 REF x1                       % Bill [0...4]
b1 Name x1 "bill"               % Bill [0...4]
b1 PRESUPPOSITION b2            % Bill [0...4]
b1 male "n.02" x1               % Bill [0...4]
b2 REF t1                       % did [5...8]
b2 TPR t1 "now"                 % did [5...8]
b2 time "n.08" t1               % did [5...8]
b4 Time e1 t1                   % did [5...8]
b2 NEGATION b4                  % not [9...12]
b4 REF e1                       % commit [13...19]
b4 Agent e1 x1                  % commit [13...19]
b4 Theme e1 x2                  % commit [13...19]
b4 commit "v.01" e1             % commit [13...19]
b3 REF x2                       % the [20...23]
b3 PRESUPPOSITION b4            % the [20...23]
b3 crime "n.01" x2              % crime [24...29]
                                % . [29...30]
```

(a)

```
male.n.02 Name "Bill"                % Bill [0-4]
time.n.08 TPR now                    % did not [5-12]
NEGATION <1                          %
commit.v.01 Agent -2 Time -1 Theme +1 % commit [13-19]
crime.n.01                           % the crime. [20-30]
```

(b)

Figure 4.2: Graphical representations of DRS clause format (a) and variable free DRS representation (b) for the text "Bill did not commit the crime."

Our main contributions in multi-lingual and multi-faceted implementation include:

Development of Data Augmentation Techniques: In the development of data augmentation techniques, we developed and tested innovative methods across English, Italian, and Urdu, including cross-lingual techniques and semantic data augmentation approaches. These methods are designed to create contextually appropriate training samples by modifying named entities, lexical categories, and grammatical structures.

Enhanced Performance Metrics: Our augmentation strategies resulted in significant improvements in performance scores for semantic parsing and text generation tasks. We quantified these improvements through detailed empirical analyses, demonstrating that our techniques can effectively enhance model accuracy and robustness. The implementation of these strategies led to measurable advancements in the quality and reliability of semantic processing across the target languages.

Cross-Lingual Insights and Scalability: We provided comprehensive insights into how cross-lingual augmentation methodologies impact semantic processing in different languages. This includes a thorough examination of the effects of augmenting Italian with English resources and an exploration of the scalability of these methods to other low-resource languages. Our findings contribute substantially to a better understanding of the potential applications and limitations of cross-lingual data augmentation, offering

valuable guidance for future research and development in this domain.

Creation of Semantic Resources: For Urdu, we successfully created the first semantically annotated corpus paired with formal meaning representations, providing a valuable and previously unavailable resource for future research. This corpus facilitates the development and evaluation of semantic parsing and text generation models specifically for Urdu, addressing a significant gap in linguistic resources for this language. The creation of this corpus represents a crucial step forward in enabling advanced NLP applications for Urdu.

Comparative Analysis: We conducted a comprehensive comparative analysis of the performance of semantic parsing and text generation models across English, Italian, and Urdu, highlighting the unique challenges and opportunities presented by each language. This comparative analysis helps identify best practices and potential areas for further research in multilingual semantic processing. By examining the differences and similarities in model performance across these languages, we have gained valuable insights into language-specific factors that influence semantic processing tasks.

Through these contributions, our research addresses key challenges in semantic parsing and text generation, offering novel solutions and valuable resources for advancing NLP technologies in English, Italian, and Urdu. The combination of innovative augmentation techniques, cross-lingual methodologies, and the creation of new linguistic resources provides a comprehensive approach to enhancing semantic processing capabilities across multiple languages, paving the way for more robust and versatile NLP systems.

The remaining chapter is organized as follows: in Section 4.2 we explain in detail the development of Urdu Meaning Bank (UMB)—the very first semantic resource for Urdu. Section 4.3 explains multi-faceted data augmentation approaches applied to linguistically diverse languages, i.e., English, Italian, and Urdu. Section 4.4 explains the experimental implementation and evaluation frameworks utilized in multi-lingual context. Results and discussion are presented in Section 4.5, and in-depth analysis of the examples through fine-grained evaluations is presented in Section 4.6. Section 4.7 emphasizes the behaviour of LLMs in the case of DRS representations, and error analysis is conducted in Section 4.8 with the conclusion in Section 4.9.

4.2 Creating a Meaning Bank for Urdu

The creation of a semantic resource for Urdu from meaning representations presents significant challenges due to the language's distinct linguistic features, which are characteristic of its language family. Operating under the assumption that the predicate-argument structure of linguistic meaning expressed in a DRS is language-neutral, we transformed English logic-text pairs from the PMB-5.0.0 into corresponding Urdu logic-text pairs. For the English-to-Urdu textual translation, we employed GoogleAPI, selected for its consistency in generating static translations compared to pre-trained generative models that produce variable outputs. These automatically translated examples went through manual annotation and correction. The primary challenge in establishing the Urdu se-

mantic resource lies in aligning DRS concepts with the word order in Urdu text. We address below the most prominent linguistic differences between English and Urdu pertaining to logical representation, along with the methodologies employed to construct an accurate semantic resource for Urdu.

These extensive dataset development efforts produced $1,200$ gold examples (fully manually annotated) and $6,857$ silver examples (partially manually annotated) for training, along with 900 gold examples for the development set and 900 gold examples for the test set. Additionally, the dataset was significantly expanded using multi-faceted data augmentation strategies, resulting in a ninefold increase in training examples (see Table 4.8). Detailed information about the multilingual dataset is provided in Table 4.5.

4.2.1 Cross-Lingual Adaptation through Named Entities

While the PMB is a multilingual corpus encompassing English, Italian, Dutch, and German variants of meaning representation, these languages adhere to Latin alphabets with left-to-right orthography, although with differing syntactic structures. A recent advancement to create a novel semantic resource from English for Chinese leveraged the similarities in writing direction (left-to-right) and syntactic structure (SVO) between the two languages. [Wang et al., 2021a] developed a Chinese semantic resource by substituting only English-named entities in DRS with Chinese counterparts and translating English text to Chinese. For Chinese semantic resource, no work was performed on syntactic structure reordering and word alignment with DRS concepts. In this chapter, we have not included Chinese results because of the following differences:

1. The authors performed experiments only for semantic parsing.
2. The neural architecture used was different i.e., LSTM.
3. Dataset size and example split were also very different from the languages reported in this chapter.

Our approach to creating Urdu semantic resources similarly began with the adoption of named entities and textual translations from English to Urdu. However, the development of a comprehensive semantic resource for Urdu necessitated additional steps, including syntactic structure reordering, alignment of DRS concepts with Urdu word order, and adjustments for grammatical gender.

4.2.2 Syntactic Structure and SBN Concept Alignment

The fundamental distinctions between English and Urdu are rooted in their syntactic structures. English follows a subject-verb-object syntactic pattern with left-to-right word order, whereas Urdu adheres to a subject-object-verb structure with right-to-left word order. Consequently, the logical concepts in DRS required realignment to correspond with Urdu's word order. In DRS, the sequence of logical concepts (nouns, verbs, adjectives, and adverbs) is intrinsically linked to the word order in textual representation. Thus, we manually transformed the English meaning representation to conform to the subject-object-verb pattern, ensuring alignment with Urdu's syntactic structure and textual word order. Figure 4.3 illustrates the concept and word alignment for English and

Urdu-based DRS-Text pairs, with color-coding to denote corresponding cross-lingual words and concepts.

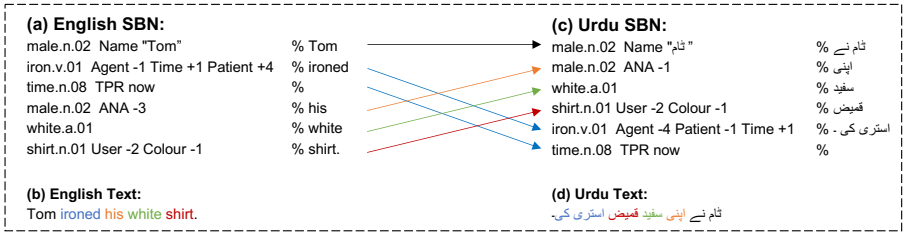


Figure 4.3: Comparing English and Urdu SBN along with their corresponding textual representations based on syntactic structure and surface alignment.

4.2.3 Grammatical Gender in Urdu

Urdu is characterized by grammatical gender, where nouns are inflected for masculine or feminine gender, and certain verbs (contingent on tense) agree with their subjects' gender. Unlike English pronouns, Urdu pronouns lack morphological or lexical features distinguishing gender. For instance, the pronoun "وہ" (woh) can denote both "he" and "she" in specific contexts. This contrasts with English meaning representations, which employ distinct concepts for male and female entities, rendering them unsuitable for direct application to Urdu. To annotate Urdu pronouns accurately, we utilized the general concepts "entity.n.01" or "person.n.01" when the antecedent's gender was indeterminate. In cases where gender could be inferred, such as from verbal inflections, we employed the more specific concepts "male.n.02" or "female.n.02" as appropriate.

4.3 Multi-faceted Data Transformation Approaches

The generation of semantically, pragmatically, and contextually accurate training data for neural semantic parsing and text generation presents significant challenges, particularly in the context of low-resource languages such as Italian and Urdu. When dealing with DRS paired with textual representations, it is imperative to meticulously adjust both components concurrently to maintain the precise mapping between meaning representations and their textual realizations, thereby ensuring data consistency and integrity. It is important to highlight that in our multi-lingual, multi-faceted data augmentation approaches, we directly applied augmentation methodologies to named entities, lexical entities, and grammatical structures in English, without facing challenges related to the absence of lexical databases such as WordNet or VerbNet. However, for Italian, the lack of lexical databases that support logical representations for Italian DRS posed a challenge for data augmentation. To address this, we leveraged the language-neutral nature of DRS for augmenting the Italian DRS. For text augmentation, we employed a machine translation approach, translating Italian text into English. We then applied the same

augmentation techniques as for English and translated the augmented data back into Italian.

For Urdu, the challenges were even greater due to the absence of an available dataset. As a first step, we developed the UMB (see Section 4.2). Subsequently, we applied a machine translation approach by translating Urdu to English, augmenting the English examples, and translating them back to Urdu for further experimentation. For back-and-forth translations, such as Italian to English or Urdu to English and vice versa, we utilized the Google API, which provides deterministic translations. This means that if the text is translated multiple times back and forth, the content of the translated text remains consistent. In contrast, when using pre-trained LLMs or fine-tuned machine translation models, the content and context of the information tend to vary with each translation attempt.

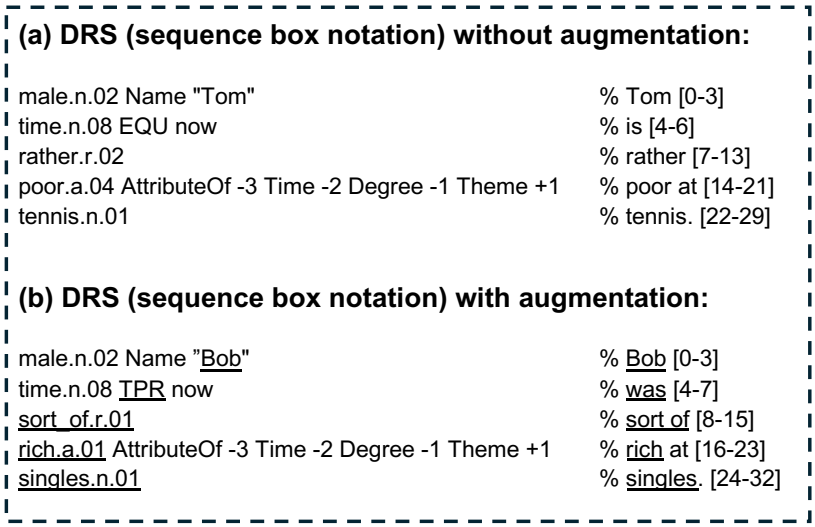


Figure 4.4: Graphical representations of DRS (a) without augmentation for the text "Tom is rather poor at tennis." and (b) with blending augmentation for the text "Bob was sort of rich at singles."

A graphical representation of English augmentation along with detailed examples are illustrated in Figure 4.4. For instance, a DRS-text pair for the sentence "Tom is rather poor at tennis." was transformed through blending augmentation into "Bob was sort of rich at singles." demonstrating alterations in proper nouns, common nouns, adjectives, adverbs, and verbal tenses. These augmentations were applied to both the gold (fully manually annotated) and silver (partially manually annotated) versions of the PMB dataset, with further analysis provided in subsequent sections. Table 4.1 illustrates English data augmentation examples for each type of transformation applied to the dataset. Italian and Urdu data transformation examples along with English translations are mentioned in Table 4.2 and Table 4.3 respectively.

Table 4.1: English named-entities, lexical, and grammatical data augmentation approaches for neural semantic parsing and text generation.

Augmentation Type	**Original Examples**	**Transformed Examples**
Named Entities	Tom asked Mary if she had been to Boston.	Bob asked Sarah if she had been to Cambridge.
Common Noun	Tom played with his dog.	Tom played with his puppy.
Verb	Tom thinks I stole the money.	Tom philosophizes I stole the money.
Adjective	He is ill.	He is well.
Adverb	The girl is deeply attached to her aunt.	The girl is profoundly attached to her aunt.
Tense	A girl is playing the flute.	A girl was playing the flute. A girl will be playing the flute. A girl has been playing the flute.
Blending	Tom is rather poor at tennis.	Bob was sort of rich at singles.

4.3.1 Named Entities Augmentation

The initial approach to dataset augmentation focused on named entity augmentation, specifically targeting the transformation of proper nouns, including person names (PER) and geopolitical entities (GPE) such as cities, states, countries, and islands. This rule-based methodology involved extracting named entities from both the textual content and the DRS while ensuring the elimination of potentially offensive content. To introduce lexical diversity and evaluate the model's capacity to handle novel lexical information, an outside-the-context strategy was employed, wherein existing named entities in the dataset were substituted with new entities not initially present. These substitutions were carefully executed within the same categorical constraints (e.g., replacing a person's name with another person's name, or a city's name with another city's name) to preserve the original semantic content of the sentences. Notably, the study refrained from utilizing public repositories for external lexical information. Instead, ChatGPT was employed to generate lists of globally frequent person names and GPEs, from which names already present in the dataset were subsequently filtered out. This methodology facilitated data augmentation without altering the underlying logical representation. For example, in the sentence "Berlin is the capital of Germany." "Berlin" was replaced with "Rome" and "Germany" with "Italy" maintaining the sentence's semantic integrity while introducing novel lexical entities.

The fundamental concept of applying named entity augmentation to the variable-free representation of DRS mirrors the methodologies detailed in Chapter 3, specifically regarding in-context and out-of-context named entity augmentation. The primary difference lies in the representation of the DRS. In Chapter 3, we worked with the clause-based format of DRS, which involves a more complex structure. Additionally, while experimenting with in-context and out-of-context replacements in Chapter 3, we observed that out-of-context replacements were more effective in enhancing model generalization. Therefore, in the named entity experiments presented in this chapter, we have directly applied the out-of-context approach.

Table 4.2: Italian named-entities, lexical, and grammatical DA approaches for neural semantic parsing and text generation. The English translation is mentioned in double-quotes.

Augmentation Type	**Original Examples**	**Transformed Examples**
Named Entities	Tom ha chiesto a Mary se fosse stata a Boston. "Tom asked Mary if she had been to Boston."	Bob ha chiesto a Sarah se fosse stata a Cambridge. "Bob asked Sarah if she had been to Cambridge."
Common Noun	Tom ha giocato con il suo cane. "Tom played with his dog."	Tom ha giocato con il suo cucciolo. "Tom played with his puppy."
Verb	Tom pensa che io abbia rubato i soldi. "Tom thinks I stole the money."	Tom filosofeggia che ho rubato i soldi. "Tom philosophizes I stole the money."
Adjective	Lui è malato. "He is ill."	Lui è bene. "He is well."
Adverb	La ragazza è profondamente legata a sua zia. "The girl is deeply attached to her aunt."	La ragazza è sinceramente legata a sua zia. "The girl is sincerely attached to her aunt."
Tense	Una ragazza suona il flauto. "A girl is playing the flute."	Una ragazza suonava il flauto. "A girl was playing the flute." Una ragazza suonerà il flauto. "A girl will be playing the flute." Una ragazza ha suonato il flauto. "A girl has been playing the flute."

4.3.2 Lexical Entities Augmentation

The lexical entities augmentation approach targets four specific lexical categories: common nouns, verbs, adverbs, and adjectives. This methodology involves modifying the content of these lexical items while ensuring the preservation of the sentence's contextual meaning. To guide the substitution process, WordNet synsets are utilized, which categorize words according to their senses and meanings. A critical aspect of this augmentation strategy is to avoid alterations that would fundamentally change the semantics of the sentence, such as substituting an animate agent with an inanimate one. The transformations applied to each of these four lexical categories are elucidated in detail in the subsequent sections.

Common Noun represents a lexical category with significant potential to influence sentence meaning, rendering their augmentation a complex task. To execute CN augmentation, a rule-based approach was implemented to extract CNs from the DRS, while the NLTK-based "WordNetLemmatizer" was employed for extraction from the corresponding text. The augmentation process involved the substitution of CNs with their hyponyms — more specific instances within the same semantic category — utilizing WordNet. This methodology facilitated the introduction of contextually relevant substitutions while preserving the original sentence's meaning. Although experiments were conducted with hypernyms (more general categories), the analysis revealed that hyponyms were more effective in maintaining the intended contextual sense, thus proving more suitable for the augmentation strategy. For instance, in the DRS, the logical

representation of a CN might be in the form 'lemma.n.xx', where 'lemma' denotes the common noun and 'xx' represents the sense number; in one example, "tennis.n.01" was replaced with "singles.n.01" to reflect a specific hyponym.

Using hyponyms in the context of DRS offers several advantages over hypernyms, particularly in terms of contextual relevance and greater specificity, especially in tasks related to semantic parsing and text generation. For instance, while the hypernym "animal" encompasses a wide range of creatures, the hyponym "dog" refers to a specific type of animal, as illustrated in the sentence "The golden retriever fetched the ball." This specificity allows for finer distinctions within semantic representations, resulting in more contextually accurate and detailed sentences. In DRS, this level of specificity is crucial for maintaining the precision of meaning, as it helps ensure that the generated text aligns closely with the intended semantics.

Verb augmentation presents a particular challenge due to the crucial role verbs play in conveying semantic phenomena and contextual meaning within sentences. To address this, WordNet-based troponyms — specific verb hyponyms — were employed to replace verbs with more precise, contextually similar alternatives. This approach facilitated the maintenance of semantic coherence while introducing lexical variety. In the DRS, verbs are represented as 'lemma.v.xx,' where 'lemma' denotes the verb and 'xx' indicates the sense number; for example, "study.v.01" might be replaced with "major.v.01". A custom rule-based technique, analogous to that used for common nouns, was applied to extract verbs from the DRS, while the NLTK-based "WordNetLemmatizer" was used for the corresponding natural language text.

Adverb augmentation necessitates contextually relevant lexical replacements to maintain the overall semantic integrity of a sentence. A WordNet-based synonym replacement approach was implemented for adverb augmentation. The rich synonym sets for adverbs provided by WordNet offered the flexibility to select alternatives with similar meanings, enabling the exploration of lexical variance while preserving semantic similarity. To ensure the consistency and semantic correctness of the substitutions, manual validation of the modified examples was performed. Consistent with the approach for other lexical categories, the adverb augmentation procedure was applied to both the DRS and the corresponding natural language text. In DRS, adverbs are represented as 'lemma.r.xx,' where 'lemma' denotes the adverb and 'xx' is the sense number; for instance, "really.r.01" might be replaced with "truly.r.01".

Adjective augmentation plays a critical role in shaping the overall meaning of a sentence. The augmentation process utilized WordNet-based antonyms to replace adjectives, facilitating the generation of new examples with diverse, yet contextually relevant meanings. This approach distinguished adjective augmentation from adverb substitution and enriched the dataset with a broader range of lexical variations. A careful analysis of the augmented data was conducted to ensure that antonym replacements did not introduce semantic errors or diminish contextual relevance. The augmentation process was consistently applied to both the adjective instances in the DRS and the corresponding natural language text. In DRS, adjectives are represented as 'lemma.a.xx,' where 'lemma' denotes the adjective and 'xx' is the sense number; for example, the adjective concept

"big.a.01" might be replaced with "small.a.01".

Table 4.3: Augmentation examples for Urdu semantic parsing and generation.

Transf Type	**Actual Example**	**Augmented English Text**	**Corresponding Urdu Text**
Proper Noun Aug	Tom asked Mary if she had been to Boston.	Ali asked Sarah if she had been to Cambridge.	علی نے سارہ سے پوچھا کہ کیا وہ کیمبرج گئی تھی۔
Common Noun Aug	Tom played with his dog.	Tom played with his puppy.	ٹام نے اپنے کتے کے ساتھ کھیلا۔
Verb Aug	Tom thinks I stole the money.	Tom philosophizes I stole the money.	ٹام کا فلسفہ ہے کہ میں نے پیسے چرائے ہیں۔
Adjective Aug	He is ill.	He is well.	وہ خیریت سے ہے۔
Adverb Aug	The girl is deeply attached to her aunt.	The girl is profoundly attached to her aunt.	لڑکی کو اپنی خالہ سے گہرا لگاؤ ہے۔
Tense Aug	A girl is playing the flute.	A girl was playing the flute.	ایک لڑکی بانسری بجا رہی تھی۔
		A girl will be playing the flute.	ایک لڑکی بانسری بجا رہی ہوگی۔
		A girl has been playing the flute.	ایک لڑکی بانسری بجا رہی ہے۔

4.3.3 Grammatical Augmentation

Grammatical augmentation encompasses modifications to the morpho-syntactic structure of sentences, with a primary emphasis on tense transformations. This methodology involves altering the temporal framework of events by shifting tenses among present, past, and future, rather than substituting contextually similar lexical items. In addition to tense alterations, the study explored a diverse range of grammatical transformations, including active and passive voice conversions, mood modifications (e.g., imperative), negation, number transformations (singular to plural), subject-object relationship adjustments, aspect variations (progressive and perfect), infinitive form implementations, first-person perspective shifts, and perfect participle applications. While transforming data examples, we also take care of other sub-types of tenses, e.g., continuous, perfect, and perfect continuous having the indefinite as the default tense type. But if the tense is already in indefinite form, we transform it into its perfect form. Table 4.4 lists different aspects of tense transformations utilized in this augmentation approach.

For tense augmentation specifically, a rule-based approach was applied to both the DRS and the corresponding text. Within the DRS, simple rules were employed to replace tense-marking logical predicates (e.g., "EQU" to "TPR/TSU" for present to past/future shifts). For the textual data, the Tenseflow API was utilized to implement these transformations. This dual-pronged approach ensured that the grammatical augmentation expanded the dataset with diverse syntactic variations while preserving the core semantic content, thereby enhancing the robustness of NLP models, particularly for tasks involving temporal reasoning and varied syntactic structures.

Table 4.4: All cases of tense change encountered in our implementation.

Conversion Type	Original Sentence	Converted Sentence
Present to Past & Future	I catch you	I caught you; I will catch you
Past to Present & Future	He cheated on me	He cheats on me; He will cheat on me
Future to Present & Past	I will love you	I love you; I loved you
First person	I said no	I say no
First person	He said no	He says no
Infinitive	I love to love	I will love to love
Ambiguous-POS	It was a thought	It will be a thought
Plural	The rabbits ran	The rabbits run
Plural	The rabbit ran	The rabbit runs
Third person singular	It will work	It works
Taking *will* as noun	The will says otherwise	The will said otherwise; The will will say otherwise
Perfect tense	He had walked to the store	He walks to the store; He will walk to the store
Continuous tense	I was going to the store	I am going to the store; I will be going to the store
Double tense change	I win because I have five cookies	I won because I had five cookies
Negation	I did not go	I do not go; I will not go
Future perfect	I will have been alive	I am alive; I was alive; I will be alive
Passive tenses	I am filled	I will be filled

4.4 Multi-lingual Experimentation and Evaluation Framework

The bidirectional transformation between semantic representations of meaning and natural language text presents significant challenges in language generation. A prominent research direction involves the utilization of neural networks to facilitate the translation between DRS and their corresponding textual realizations [Wang et al., 2021a, Amin et al., 2024, van Noord et al., 2020, Amin et al., 2022b, van Noord et al., 2018, Basile and Bos, 2011, Wang et al., 2023b]. However, prior to addressing semantic parsing and text generation via neural networks, it is imperative to preprocess the logical forms of the DRS. This preprocessing entails the linearization of logic-text pairs into sequences to ensure compatibility with neural model architectures.

4.4.1 Categorization of Augmented Datasets

PMB is a multilingual corpus comprising parallel translations across various languages e.g., English, Italian, German, Dutch, and Urdu. For the purposes of our investigation, we employed the English, Italian, and Urdu versions of PMB-5.0.0 for data augmentation and German and Dutch as a baseline in our experiments. The PMB dataset is stratified into three quality tiers: gold (fully manually annotated and verified), silver (partially manually annotated and verified), bronze (without manual inspection), and Copper (machine-translated version of English data examples). For **English**, the gold PMB dataset encompasses 9,057 train gold samples, 1,132 development, and 1,132 testing samples. The silver tier contributes an additional 143,731 training examples, culminating in a total of 152,788 gold and silver English training instances. For **Italian**, the distribution is as follows: 745 train gold, 555 development, 555 Test, and 4,316 train silver examples. The distribution of other low-resource languages includes: **German** comprising 1,206 train gold, 900 development, 900 test, and 6,862 train silver examples. **Dutch** consists of 586 train gold, 435 development, 435 test, and 1,646 Train

Silver examples. And for **Urdu**, we have 1,200 train gold, 900 development, 900 test, and 6,857 train silver examples. Table 4.5 lists dataset distribution states for multiple languages.

Table 4.5: Dataset split along with statistic numbers for multi-lingual baselines.

Languages	**Train Gold**	**Dev**	**Test**	**Train Silver**
Italian	745	555	555	4,316
German	1,206	900	900	6,862
Dutch	586	435	435	1,646
English	9,057	1,132	1,132	143,731
Urdu	1,200	900	900	6,857

The classification of the augmented dataset into discrete categories of named entities, lexical items, and grammatical transformations underscores the logical rationale behind the experiments delineated in this study. As previously elucidated, we have implemented six fundamental augmentation methodologies for modifying both the DRS and the corresponding text. However, we did not exhaustively explore all 2^6 combinations of different augmentation procedures in the experiments reported herein. This research does not aim to identify the optimal individual or blended augmentation approach in terms of its impact on model performance. Instead, we employed four distinct types of data combinations in our English experiments, corresponding to four neural models:

1. without augmentation — our baselines utilizing the original dataset examples.

2. individual augmentation — applying one type of augmentation procedure at a time.

3. blended augmentation — applying all possible augmentation strategies to a single example.

4. compound augmentation — concatenating all individually applied augmentation strategies.

We implemented baseline, individual, and compound augmentation strategies for the Italian and Urdu experiments, utilizing the best neural model identified through extensive experimentation with English data augmentation. Table 4.6 enumerates the augmentation types and the number of training examples for all combinations of augmentation methods utilized in English experiments. Table 4.7 provides detailed information on the types of augmentation, dataset sizes, and the number of training examples for individual and compound augmentation strategies employed in our Italian experiments. For Urdu, Table 4.8 provides detailed information on the augmentation type, dataset size, and the number of training examples corresponding to individual and compound augmentation strategies employed in our experiments. In our experimental implementation, we augmented the training set only, while the development and test sets remained unchanged.

Table 4.6 presents a detailed analysis of the impact of various augmentation strategies on the size of the dataset for English, comparing the original dataset with individual, blending, and compound augmentations. The original dataset, without augmentation, consists of 9,057 gold examples, 143,731 silver examples, and 152,788 combined gold-silver examples. The application of individual augmentation techniques, such as Named Entities, Common Nouns, Adjectives, Adverbs, and Verbs, doubles the dataset size, leading to 18,114 gold, 287,462 silver, and 305,576 gold-silver examples for each type of augmentation. Tense augmentation and blending augmentation both have a fourfold impact on the dataset size, increasing the total to 36,228 gold, 574,924 silver, and 611,152 gold-silver examples. Compound augmentation, which combines multiple strategies, results in the largest expansion, multiplying the dataset size by nine. This leads to a dataset comprising 81,513 gold examples, 1,293,579 silver examples, and a total of 1,375,092 gold-silver examples. The development and test sets remain unchanged at 1,132 examples each. This substantial increase in dataset size, particularly through compound augmentation, underscores the potential for generating more diverse and comprehensive training data, thereby enhancing the performance and generalization of neural models.

Table 4.6: Impact on the size of dataset examples for English without augmentation and with individual, blending, and compound augmentation.

Training Dataset Type	**Dataset Size**	**Gold Ex.**	**Silver Ex.**	**Gold-Silver Ex.**
Original (w/o Augmentation)	x1	9 057	143 731	152 788
Named Entities Augmentation	x2	18 114	287 462	305 576
Common Noun Augmentation	x2	18 114	287 462	305 576
Adjective Augmentation	x2	18 114	287 462	305 576
Adverb Augmentation	x2	18 114	287 462	305 576
Verb Augmentation	x2	18 114	287 462	305 576
Tense Augmentation	x4	36 228	574 924	611 152
Blending Augmentation	x4	36 228	574 924	611 152
Compound Augmentation	x9	81 513	1 293 579	1 375 092
Development Examples	–	1 132	–	–
Test Examples	–	1 132	–	–

Table 4.7 provides a comparative analysis of the number of Italian instances with and without augmentation, as well as those with individual and compound augmentations, demonstrating the impact of different augmentation methods on dataset size. The original dataset, without augmentation, comprised 5,061 gold-silver samples in total, consisting of 4,316 silver examples and 745 gold examples. Application of individual augmentations, including Named Entities, Common Noun, Adjective, Adverb, and Verb augmentations, doubles the dataset size, resulting in 1,490 gold, 8,632 silver, and 10,122 gold-silver examples for each augmentation type. Tense augmentation has an even more pronounced effect, quadrupling the dataset size to 2,980 gold, 17,264 silver, and 20,244 gold-silver examples. Compound augmentation yields the most substantial increase, expanding the dataset size ninefold to 6,705 gold, 38,844 silver, and 45,549 gold-silver

examples. Compound augmentation incorporates multiple augmentation strategies. The number of examples in both the development and test sets remains constant at 555. This significant expansion of the dataset size highlights the potential for more comprehensive and diverse training data, which can enhance the robustness and performance of neural networks.

Table 4.7: Impact on the size of Italian dataset examples without augmentation and with individual and compound augmentation. Note: Ex. = Examples; w/o = without.

Training Dataset Type	**Dataset Size**	**Gold Ex.**	**Silver Ex.**	**Gold-Silver Ex.**
Original (w/o Augmentation)	x1	745	4316	5061
Named Entities Augmentation	x2	1490	8632	10122
Common Noun Augmentation	x2	1490	8632	10122
Adjective Augmentation	x2	1490	8632	10122
Adverb Augmentation	x2	1490	8632	10122
Verb Augmentation	x2	1490	8632	10122
Tense Augmentation	x4	2980	17264	20244
Compound Augmentation	x9	6705	38844	45549
Development Examples	–	555	–	–
Test Examples	–	555	–	–

Table 4.8 provides an analysis of the impact of different augmentation techniques on the size of the Urdu dataset, highlighting the growth in dataset size as various augmentations are applied. The original dataset, without augmentation, consists of 1,200 gold examples, 6,857 silver examples, and 8,057 combined gold-silver examples. Individual augmentations, such as Adjective, Adverb, Verb, Proper Noun, and Common Noun augmentations, each result in a twofold increase in dataset size, bringing the total to 2,400 gold examples, 13,714 silver examples, and 16,114 combined examples for each augmentation type. Tense augmentation has an even more pronounced effect, quadrupling the dataset size to 4,800 gold examples, 27,428 silver examples, and 32,228 combined examples. Compound augmentation, which integrates multiple augmentation strategies, produces the most significant growth, expanding the dataset size ninefold to 10,800 gold examples, 61,713 silver examples, and 72,513 combined examples. The development and test sets remain fixed at 900 examples each, unaffected by augmentation. This considerable increase in dataset size, particularly through compound augmentation, indicates a substantial enhancement in data diversity, which is expected to contribute to improved performance and generalization in neural models trained on the augmented Urdu dataset.

4.4.2 Language Models used for Multi-lingual Semantic Parsing and Text Generation

For our experiments, we initially focused on English and subsequently expanded to low-resource languages i.e., Italian and Urdu. Given the state-of-the-art capabilities

Table 4.8: Urdu dataset size along with the number of examples corresponding to each augmentation flavor applied.

Training Dataset Type	Dataset Size	Gold Ex.	Silver Ex.	Gold-Silver Ex.
Original (w/o Augmentation)	x1	1200	6857	8057
Adjective Augmentation	x2	2400	13714	16114
Adverb Augmentation	x2	2400	13714	16114
Verb Augmentation	x2	2400	13714	16114
Proper Noun Augmentation	x2	2400	13714	16114
Common Noun Augmentation	x2	2400	13714	16114
Tense Augmentation	x4	4800	27428	32228
Compound Augmentation	x9	10800	61713	72513
Development Examples	–	900	–	–
Test Examples	–	900	–	–

of transformer-based neural architectures for NLG tasks, four distinct sequence-to-sequence transformer models (byT5, mT5, T5, and mBART) were used for English augmentation experiments. Our research not only focuses on overall performance metrics but also investigates the specific effects of various data augmentation approaches on the neural models' ability to differentiate between semantic representations and generated text.

The mBART model employs a bidirectional, autoregressive architecture optimized for specific natural language tasks such as summarization and machine translation [Liu et al., 2020]. Conversely, T5 [Liu et al., 2023] and mT5 [Xue et al., 2021] are monolingual and multilingual models, respectively, designed primarily for general text-to-text applications. The byT5 model offers finer control over its task-specific behavior through the prepending of task-specific prefix tokens (byte sequences) during the tokenization process [Xue et al., 2022]. We argue that the chosen model architecture and tokenization strategy have a substantial influence on model performance and its applicability to various NLP tasks and linguistic phenomena. Additionally, these architectural decisions have a direct impact on the representation of text data during both the input and output stages of processing.

For our experimental implementation, we adopted the conventional approach of fine-tuning pre-trained models on task-specific data. To this end, we implemented a two-stage fine-tuning process as described by [van Noord et al., 2020]. In the initial stage, we utilized augmented data for a limited number of epochs to fine-tune the model, thereby imparting preliminary knowledge about DRS. The second stage involved fine-tuning the model exclusively with gold data. We primarily adhered to the default hyperparameters for these models, with minor modifications including the use of AdamW as an optimizer, a polynomial learning rate decay of 1e-4, GeGLU as an activation function, a batch size of 8, and a maximum sequence length of 512. Table 4.9 lists some of the hyperparameters used in our experiments. We also used the same hyperparameter setting for our experiments with Italian and Urdu.

Table 4.9: Hyperparameter setting for experiments in English, Italian, and Urdu.

Parameter	Value
Optimizer	AdamW
Learning rate	1e-4
Batch size	32
Max length	512
Activation function	GeGLU
Epoch for fine-tuning stage 1	5
Epoch for fine-tuning stage 2	early stopping

4.4.3 Evaluation Metrics for Semantic Parsing and Generation

In this section, we report different types of evaluation measures that have been used to evaluate the quality of multi-lingual semantic parsing and text generation tasks.

Evaluation Metric for Semantic Parsing

The evaluation methodology for semantic parsing (text-to-DRS) comprises two principal stages [Poelman et al., 2022]. Initially, the DRS logical form is converted into Penman notation [Liu et al., 2015]. Subsequently, the system output is compared against the gold standard DRS by quantifying the overlap of triples using SMATCH [Cai and Knight, 2013] — a specialized tool designed for evaluating AMR parsing performance. The evaluation metric employed is the F1-Score, which represents the harmonic mean of precision and recall. This score ranges from 0 (lower bound) to 1 (upper bound), amalgamating precision (P) and recall (R) into a single, comprehensive metric. Crucially, the adoption of DRS as the semantic representation formalism enabled the analysis of semantic parsing results using the SMATCH rule-based evaluation tool. For all of our experimentation with English, Italian, and Urdu, we have evaluated semantic parsing through SMATCH (F1-Score).

For example, consider the sentence "The cat chased the mouse." The gold standard DRS for this sentence, expressed in Penman notation, is structured as follows: `( (cat x1) (mouse y1) (chase x1 y1) )`. Here, `x1` and `y1` are variables representing the entities involved in the action. If our semantic parser generates the DRS `( (cat a1) (mouse b1) (chase a1 b1) )`, we can evaluate its performance using SMATCH. First, both DRS representations are converted into sets of triples. For the gold standard, the triples are `(cat, x1)`, `(mouse, y1)`, and `(chase, x1, y1)`. For the parser output, the triples are `(cat, a1)`, `(mouse, b1)`, and `(chase, a1, b1)`. If all triples from the system match those in the gold standard, both precision and recall would equal 1, resulting in an F1-Score of 1, indicating perfect performance. However, if the parser fails to generate one of the triples, the precision might remain high if the output contains few incorrect triples, while recall would drop due to the missing triple, leading to a lower F1-Score.

Evaluation Metrics for Text Generation

To ensure alignment with human judgment in our evaluation process, we implemented a multi-faceted assessment of model-generated text quality for DRS-to-Text generation. This comprehensive approach incorporates:

1. n-gram-based automatic evaluation metrics, including BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], ROUGE [Lin, 2004], and chrF [Popović, 2015].

2. a neural framework-based evaluation measure, COMET [Rei et al., 2020].

3. pre-trained model-based measures such as BERTScore [Hanna and Bojar, 2021].

These metrics collectively evaluate the similarity between system-generated text and reference texts, providing a multidimensional evaluation of text generation performance. Furthermore, we have conducted human evaluation to elucidate the correlation between human and automatic evaluations.

4.5 Results and Discussion

In this section, we present a detailed analysis of the experimental results for English (see Section 4.5.1), Italian (see Section 4.5.2), and Urdu (see Section 4.5.3) across both semantic parsing and text generation tasks. The evaluation is based on a range of automatic metrics, providing a comprehensive assessment of model performance for each language and task. Through this analysis, we aim to highlight the effectiveness of our approaches and their adaptability across multiple languages.

4.5.1 English

Our multi-lingual multi-faceted data augmentation experiments start with English where we have segmented our experiments into three distinct phases. The first phase started with experiments without augmentation on the Gold version of PMB-5.0.0 (refer to Table 4.10), analyzing the performance of the four models. We observed that the performance of mBART was comparably low (underlined in Table 4.10) for both semantic parsing and generation tasks, leading to its exclusion from subsequent phases.

Table 4.10: Results of English semantic parsing and text generation for T5, mT5, mBART, and byT5 on the Gold version of PMB-5.0.0 without augmentation. Note: underline indicates the lowest scores and bold indicates the best scores among these models.

Model Type	PMB Type	Semantic Parsing	Text Generation				
		SMATCH (F1)	BLEU	METEOR	COMET	chrF	BERT Score
T5	Gold	74.35	**62.41**	49.57	**93.79**	**78.28**	**97.67**
mT5	Gold	42.13	43.83	42.79	88.32	70.39	96.29
mBART	Gold	40.70	33.27	39.40	86.93	67.76	95.64
byT5	Gold	**75.91**	60.60	**49.69**	92.15	78.19	97.45

The second experimental phase involved testing all individual augmentation approaches applied exclusively to Gold-PMB. Table 4.11 presents semantic parsing and generation results for six individual augmentation approaches applied to three models (experiments 1-6 for T5, 7-12 for mT5, and 13-18 for byT5). This phase revealed close competition between byT5 and T5, while highlighting the comparatively lower performance of mT5 for both semantic parsing and text generation tasks. Overall, all models demonstrated improved performance compared to the non-augmented baseline. However, for computationally intensive experiments involving gold-silver with augmentation (refer to Table 4.6 for training example counts), we sought to identify the model with the highest performance. After careful analysis of byT5 and T5 results, we selected byT5 for our third phase experiments, which involved gold-silver augmented data (results presented in Table 4.12).

Table 4.11: English experimental results of T5 (Exp. 1-6), mT5 (Exp. 7-12), and byT5 (Exp. 13-18) for individual augmentation experiments. Exp. 19-20 are the blending and compound augmentation flavors of byT5 (because it is the best among T5 and mT5). Underlined results represent the highest scores corresponding to one model type, while bold and underlined highlight the best among all models and augmentation flavors.

Exp.	Augmentation Type	Semantic Parsing	Text Generation				
		SMATCH (F1)	BLEU	METEOR	COMET	chrF	BERT Score
1	Adjective	80.23	64.42	50.63	93.77	80.02	97.80
2	Adverb	81.15	64.48	50.62	93.96	79.54	97.79
3	Common Noun	80.64	64.24	50.64	94.00	79.54	97.82
4	Proper Noun	81.12	63.78	50.86	93.71	80.05	97.79
5	Verb	80.84	65.46	51.43	93.83	80.96	97.85
6	Tense	82.24	64.76	51.04	93.98	80.55	97.81
7	Adjective	86.49	54.71	46.96	91.65	75.57	97.12
8	Adverb	73.97	58.31	47.85	92.40	76.79	97.44
9	Common Noun	75.62	58.13	48.30	92.79	77.47	97.45
10	Proper Noun	66.89	51.40	44.02	89.14	72.04	96.53
11	Verb	82.66	49.81	45.57	91.12	74.68	97.02
12	Tense	75.12	56.50	47.45	91.82	76.75	97.25
13	Adjective	81.74	63.77	50.54	93.28	79.94	97.70
14	Adverb	85.88	64.09	51.31	93.24	79.83	97.74
15	Common Noun	87.21	**65.77**	51.39	93.36	80.51	97.74
16	Proper Noun	87.91	64.30	51.07	93.15	80.28	97.72
17	Verb	86.52	64.80	51.38	**94.07**	80.40	97.83
18	Tense	88.84	65.09	**52.04**	93.74	**81.11**	97.86
19	Blending	89.12	64.78	51.60	93.70	80.31	**97.93**
20	Compound	**90.01**	64.67	50.97	93.53	80.15	97.82

The selection of byT5 for further experimentation was predicated on several factors:

1. Its inherent multilingual capabilities render it highly adaptable and versatile across a wide spectrum of natural language applications.

2. The model's byte- and character-level tokenization facilitates the capture of nuanced language features, diverse character sets and scripts, and semantic complexities.

3. byT5 exhibits significant robustness to noisy data, a critical feature for tasks sensitive to variations in spelling and pronunciation.

4. Its distinctive token-free approach, operating directly on raw UTF-8 text without reliance on sub-word or word vocabularies, represents a paradigm shift in NLP modeling.

5. byT5 demonstrated superior performance in individual augmentation experiments compared to T5 and mT5 (refer to Table 4.11), further supporting its selection for subsequent experimental phases.

Notably, byT5 has achieved state-of-the-art results on multilingual NLP benchmarks, as corroborated by recent studies [Belouadi and Eger, 2023, Stankevičius et al., 2022, Xue et al., 2022]. With this motivation, for out experiments with Italian and Urdu, we have used byT5 in our experiments.

Comparative analysis of overall model performance revealed that data augmentation significantly enhanced model generalization capabilities for both semantic parsing and generation tasks. Comparing first phase results—without augmentation (Table 4.10)—against individual augmentation outcomes (exp.1-18 in Table 4.11) demonstrates substantial improvements: a 13-point increase for byT5, with optimal performance on tense augmentation (exp.18 in Table 4.11, underlined); a 44-point increase for mT5, with best performance on adjective augmentation (exp.7 in Table 4.11, underlined); and an 8-point increase for T5, with best performance on tense augmentation (exp.7 in Table 4.11, underlined) for semantic parsing (SMATCH F1-score). Despite mT5's substantial improvement, byT5 maintained superior performance, particularly in tense augmentation (exp.18 in Table 4.11). In blending and compound augmentation scenarios for byT5, semantic parsing scores further improved by 2 points—yielding a cumulative 15-point improvement compared to non-augmented results—with optimal performance observed in compound augmentation (exp.20 in Table 4.11, bold and underlined).

Similarly, the DRS-to-text generation task exhibited enhanced model robustness through the application of data augmentation approaches. Comparative analysis of individual augmentation results for byT5, mT5, and T5 (Table 4.11) revealed significant improvements: a 5-point increase in BLEU scores for byT5, with optimal performance in common noun augmentation (exp.15 in Table 4.11, bold and underlined); a 15-point increase for mT5, with peak performance in adverb augmentation (exp.8 in Table 4.11, underlined); and a 3-point increase for T5, with best performance in verb augmentation (exp.5 in Table 4.11, underlined). Overall BLEU score comparisons indicated close competition between byT5 and T5, with byT5 marginally outperforming in common noun augmentation (exp.15 in Table 4.11, bold and underlined).

METEOR score comparisons revealed performance gains relative to non-augmented results (Table 4.10). ByT5 demonstrated a 3-point improvement, the highest among all models, for tense augmentation (exp.18 in Table 4.11, bold and underlined). MT5 and T5 showed 6-point and 2-point improvements, with optimal performance in common noun (exp.9 in Table 4.11, underlined) and verb augmentation (exp.5 in Table 4, underlined), respectively. COMET evaluations indicated 2-point and 1-point gains for byT5 and T5, with peak scores in verb augmentation (exp.17—bold and underlined, exp.5—underlined in Table 4.11), while mT5 exhibited a 4-point gain, excelling in common noun augmentation (exp.9 in Table 4.11, underlined). ChrF assessments showed improvements of 3, 7, and 2 points for byT5, mT5, and T5, with optimal performance in

tense (exp.18—bold and underlined), common noun (exp.9—underlined), and verb augmentation (exp.5—underlined) in Table 4.11, respectively. BERTScore evaluations, based on contextual embeddings of pre-trained models, also demonstrated improvements for tense (exp.18—underlined), common noun (exp.9—underlined), and verb augmentation (exp.5—underlined) for byT5, mT5, and T5, respectively (Table 4.11). BERTScore further improved with blending augmentation applied through byT5 (exp.19 in Table 4.11, bold and underlined), yielding the highest score.

Table 4.12: English experimental results of byT5 without augmentation on Gold and Silver PMB dataset (Exp. 1), individual augmentation (Exp. 2-7), and byT5 blending and compound augmentation (Exp. 8-9) experiments. Bold indicates the best scores. The underlined results indicate literature-based best scores. Bold represents our best scores and bold and underlined represents the best in all experiments in semantic parsing and text generation. Note: Aug. = Augmentation; S-Par. = Semantic Parsing; G = Gold; S = Silver; and B = Bronze version(s) of PMB. MET. = METEOR; CMT. = COMET; B_Scr. = BERTScore; C.N. = Common Noun; and P.N. = Proper Noun.

Experimentation Type	**Model Type**	**PMB Type**	**Aug. Type**	**S-Par. (F1)**	**Generation Results**				
					BLEU	**MET.**	**CMT.**	**chrF**	**B_Scr.**
[Amin et al., 2022b]	bi-LSTM	G	Tense	–	52.30	41.53	–	–	–
[Amin et al., 2024]	byT5	G	Nouns	–	57.15	45.90	–	–	97.02
[Wang et al., 2021b]	bi-LSTM	G+S	–	–	69.30	51.80	–	–	–
[Noord, 2019]	NeuDRS	G+S	–	84.50	–	–	–	–	–
[Amin et al., 2022b]	bi-LSTM	G+S	Tense	–	72.38	53.18	–	–	–
[Wang et al., 2023a]	mBART	G+S+B	–	94.70	74.50	55.00	102.90	–	–
[Wang et al., 2023b]	bi-LSTM	G+S	–	91.00	–	–	–	–	–
[Wang et al., 2021a]	bi-LSTM	G+S	–	88.10	–	–	–	–	–
[Zhang et al., 2024]	DRS-MLM	G+S	–	91.50	71.90	54.90	93.00	–	–
1	byT5	G+S	–	92.39	72.90	55.08	95.53	84.49	98.46
2	byT5	G+S	Adjective	93.48	74.12	55.99	95.94	85.23	98.58
3	byT5	G+S	Adverb	93.25	**74.98**	**56.12**	**95.98**	**85.61**	**98.61**
4	byT5	G+S	C.N.	92.62	73.92	55.95	95.77	85.21	98.58
5	byT5	G+S	P.N.	93.07	73.16	55.43	95.67	84.78	98.51
6	byT5	G+S	Verb	92.95	72.73	55.29	95.59	84.39	98.50
7	byT5	G+S	Tense	93.38	72.41	54.80	95.51	83.82	98.45
8	byT5	G+S	Blending	93.18	73.89	55.75	95.62	84.80	98.52
9	byT5	G+S	Compound	**93.56**	73.45	55.61	95.81	84.96	98.54

To optimize computational resources, we exclusively utilized byT5 in our phase three augmentation experiments, encompassing Gold and Silver data examples with all variants of augmentation (listed in Table 4.6). Furthermore, we conducted comparative analyses with literature-based implementations, considering dataset type, model utilized, and augmentation technique applied (Table 4.12).

We extended our implementation to encompass Gold and Silver data augmentation across four scenarios: (i) without augmentation (exp.1); (ii) individual augmentation (exp.2-7); (iii) blending augmentation (exp.8); and (iv) compound augmentation (exp.9), as presented in Table 4.12. Data augmentation consistently enhanced model generalization and robustness capabilities for computationally intensive semantic parsing and text generation experiments. With Gold and Silver data augmentation, byT5 exhibited optimal results for compound augmentation (exp.9 in Table 4.12, bold and underlined) in semantic parsing, representing the best performance across all experiments in this

study. Similarly, for the DRS-to-Text generation task, data augmentation consistently improved model performance, with adverb augmentation (exp.3 in Table 4.12, bold and underlined) yielding optimal results across all evaluation metrics. Comparing the overall performance of byT5 (our best-performing model), the SMATCH (F1) score for semantic parsing increased from 75.91 to 93.56. Text generation also demonstrated significant improvements, with BLEU scores rising from 60.60 to 74.98, METEOR from 49.69 to 56.12, COMET from 92.15 to 95.98, chrF from 78.19 to 85.61, and BERT scores from 97.45 to 98.61, transitioning from Gold to Gold and Silver variants of the PMB dataset.

Our research contributes four novel aspects to the field. First, we have implemented named entity, lexical, and grammatical data augmentation approaches for both semantic parsing and generation tasks. Previous studies primarily focused on exploring tenses [Amin et al., 2022b] or nouns [Amin et al., 2024] using supersenses and were limited to the DRS-to-text generation task. Second, we have developed a novel WordNet-based lexical substitution approach for data augmentation, which maximizes the preservation of the true contextual meaning of sentences. Third, our implementation emphasizes the variable-free representation of DRS, in contrast to earlier approaches that concentrated on the clause format of DRS. This shift underscores the fact that applications of variable-free DRS remain under-explored. Finally, we have conducted data augmentation experiments across four state-of-the-art transformer models: mBART, T5, mT5, and byT5. Previous research has been restricted to either a single transformer model type or outdated LSTM-based architectures. To our knowledge, no prior studies have comprehensively explored named entity, lexical, and grammatical data augmentation on meaning representation for both semantic parsing and generation tasks, beyond the aforementioned works on DRS-to-Text generation.

Comparing our individual augmentation results on the Gold-PMB dataset (Table 4.11) with literature-based implementations (Table 4.12), we highlight: (i) tense augmentation on the Gold-PMB dataset using a bi-LSTM-based model [Amin et al., 2022b]; and (ii) a byT5-based model on the Gold-PMB dataset with optimal results among various noun augmentation combinations [Amin et al., 2024] (see first two 'G' PMB Type results in Table 4.12). Our models demonstrate superior performance across all aspects of data augmentation. This comparison is based on training example size, model selection, and augmentation type applied to the training data. Both literature implementations utilize Gold-PMB, with tense and noun augmentation employing bi-LSTM and byT5 models respectively, thus we compare these results exclusively with our Gold-PMB experiments.

Our Gold-Silver implementation yields state-of-the-art results for both semantic parsing and generation tasks when compared to other literature-based models. We acknowledge that our results do not surpass the mBART-based model results [Wang et al., 2023a] in SMATCH F1-Score for semantic parsing and COMET score in text generation. However, we identified that a direct comparison between our best model and the mBART implementation is not equitable due to several factors:

1. The learning procedure for mBART differs from our implementation. mBART is initially trained from scratch in an unsupervised manner on DRS data using masked language modeling. This is followed by a fine-tuning phase on gold DRS data, tailored specifically for semantic parsing and generation tasks. In contrast,

our approach involves utilizing pre-trained models, which we then fine-tune with our augmented dataset.

2. The training dataset used in the mBART approach includes Gold, Silver, and Bronze variants across multiple languages, such as English, Italian, German, and Dutch. In contrast, we utilized only Gold and Silver data, which, after applying data augmentation, was still less than the combined Gold, Silver, and Bronze data used in the mBART approach.

3. The mBART training utilized the PMB-4.0.0 dataset release, which differs from PMB-5.0.0 in its distribution of training, development, and testing examples. The larger and more diverse test set in PMB-5.0.0 presents greater challenges.

In our experiments, we fine-tuned the model using our semantically correct and contextually similar augmented dataset, achieving superior results to mBART in BLEU and METEOR-based evaluation measures (see exp.3 in Table 4.12).

It is crucial to elucidate our decision against fine-tuning a model specifically trained on DRS-text pairs throughout our experimentation (as demonstrated in [Wang et al., 2023a]). Notably, [Zhang et al., 2024] explored fine-tuning the pre-trained mBART model, termed DRS-MLM, for semantic parsing and text generation tasks. However, DRS-MLM exhibited only marginal improvements of 0.1 in SMATCH F1-score for semantic parsing compared to the byT5 implementation. Moreover, DRS-MLM underperformed byT5 itself on the text generation task, as evidenced in Table 4.12. Interestingly, our approach of fine-tuning byT5 without data augmentation (see exp.1 in Table 4.12) outperformed both semantic parsing using DRS-MLM and text generation using byT5 (as reported by [Zhang et al., 2024]). Additionally, our empirical investigations revealed the limited effectiveness of fine-tuning the standard mBART model (see first-phase experiments in Table 4.10). This finding led us to adopt fine-tuning byT5 for all our experiments.

We also conducted a qualitative analysis to investigate the impact of data size and the crucial role of textual transformations applied to original examples. Our objective was to analyze whether performance gains in semantic parsing and text generation were attributable to increased training data volume or to the presence of contextually similar, semantically correct, and diverse training data. The analysis emphasized the significant role of data augmentation in enhancing model robustness, beyond mere dataset size expansion. If dataset size were the sole determinant, we would expect optimal results for compound augmentation in text generation tasks. However, we observed superior results for adverb augmentation (see exp.3 in Table 4.12) in text generation, despite having six times fewer training examples compared to compound augmentation. We further emphasize that a small amount of high-quality data can have a substantial impact on model performance compared to larger datasets that are imperfectly annotated or corrected. We achieved significant performance gains through data augmentation applied to Gold examples (compare baseline results in Table 4.10 with individual augmentation results in Table 4.11) relative to the gains achieved through Gold and Silver examples (compare exp.1 with exp.3 and exp.9 in Table 4.12) for both semantic parsing and generation tasks.

4.5.2 Italian

Based on the results of our English experiments, which identified byT5 as the best-performing model, we employed byT5 for the Italian experiments as well, utilizing a two-stage fine-tuning strategy: initial pre-fine-tuning with gold and silver (for exp.1–12), and gold, silver, and copper (for exp.13–21) data for 5 epochs to provide foundational DRS knowledge, followed by fine-tuning on gold data with an early stopping mechanism [van Noord et al., 2020]. We also conducted experiments with T5 specialized in Italian (IT5) [Sarti and Nissim, 2024], a model that had exhibited promising results in Italian language understanding and generation across various benchmarks (see exp.12 in Table 4.13).

The experimental results reported in Table 4.13 demonstrate the efficacy of diverse Data Augmentation (DA) strategies in enhancing semantic parsing and text generation tasks for Italian DRS. We utilized different variants of T5 (byT5 and IT5) models and evaluated performance on the PMB-5.0.0 dataset, employing SMATCH F1 for parsing and BLEU, METEOR, COMET, chrF, and BERTScore metrics for generation tasks.

Table 4.13: Italian semantic parsing and generation results of byT5 and IT5 with multi-lingual baselines and augmentation on PMB-5.0.0. The best results are **bold** and underlined. (Aug = Augmentation; Adj = Adjective; Adv = Adverb; PN = Proper Noun; CN = Common Noun; Comp = Compound; G = Gold; S = Silver; C = Copper).

Exp.	Implementation Type	Dataset	Parsing Results	Generation Results				
		Flavour	SMATCH (F1%)	BLEU	BERTScore	METEOR	COMET	chrF
1	German	G+S	73.00	34.14	88.24	30.07	59.53	53.72
2	Dutch	G+S	42.77	19.83	84.98	25.36	51.78	46.92
3	English	G+S	91.42	71.89	96.01	54.52	86.38	83.80
4	Italian without Aug	G+S	76.10	37.79	88.86	30.83	81.66	54.84
5	Adj Aug	G+S	80.86	42.48	90.02	33.19	84.56	58.95
6	Adv Aug	G+S	82.70	42.30	90.00	33.07	85.07	59.21
7	CN Aug	G+S	81.18	40.02	89.23	32.23	83.00	56.87
8	PN Aug	G+S	80.07	42.62	89.83	33.36	84.33	59.07
9	Verb Aug	G+S	80.15	39.99	89.48	31.90	83.10	57.04
10	Tense Aug	G+S	84.13	44.49	90.26	33.46	85.14	60.05
11	Comp Aug	G+S	85.98	45.12	90.56	34.54	85.66	61.66
12	IT5 with Comp Aug [Sarti and Nissim, 2024]	G+S	50.57	10.97	79.38	16.25	56.31	29.76
	byT5 [Zhang et al., 2024]	G+S+C	87.20	53.20	—	38.50	87.50	—
13	Italian without Aug	G+S+C	89.22	56.46	92.72	40.48	90.02	70.38
14	Adj Aug	G+S+C	89.46	56.77	92.90	40.49	90.02	70.66
15	Adv Aug	G+S+C	89.69	57.00	92.95	40.62	90.71	70.66
16	CN Aug	G+S+C	90.46	57.28	92.85	40.80	90.21	70.59
17	PN Aug	G+S+C	89.28	56.98	92.76	40.57	90.27	70.56
18	Verb Aug	G+S+C	**90.56**	56.15	92.80	40.49	90.10	70.46
19	Tense Aug	G+S+C	89.35	**57.48**	**92.97**	**40.95**	**90.97**	**70.88**
20	Comp Aug	G+S+C	89.44	56.58	92.79	40.87	90.21	70.63

In the multilingual baseline comparisons, Italian (76.10% SMATCH F1 for parsing) exhibits superior performance to Dutch (42.77%) and comparable results to German (73.00%), while expectedly trailing English (91.42%). For text generation, Italian achieves baseline scores of 37.79 BLEU, 30.83 METEOR, 81.66 COMET, 54.84 chrF, and 88.86 BERTScore, positioning it above Dutch and German across all metrics.

Individual augmentation strategies uniformly yield improvements over the baseline Italian model. For parsing tasks, tense augmentation demonstrates the highest efficacy among singular strategies, achieving 84.13% SMATCH F1. In generation tasks, tense augmentation emerges as the most effective individual strategy, attaining scores of 44.49 BLEU, 33.46 METEOR, 85.14 COMET, 60.05 chrF, and 90.26 BERTScore. These enhancements indicate that each augmentation type contributes uniquely to the

semantic understanding and generative capabilities of the neural model. The compound augmentation approach, which integrates all augmentation strategies, produces optimal results for the Gold+Silver (G+S) dataset. This comprehensive strategy achieves 85.98% SMATCH F1 for parsing and notable improvements across all generation metrics (45.12 BLEU, 34.54 METEOR, 85.66 COMET, 61.66 chrF, and 90.56 BERTScore), underscoring the synergistic benefits of combining diverse augmentation techniques. The performance of IT5 proved inadequate when applied to formal meaning representations, i.e., DRS. The model exhibited suboptimal results in both semantic parsing and text generation tasks subsequent to fine-tuning on the compound augmentation dataset.

Furthermore, comparisons with extant literature ([Zhang et al., 2024] in Table 4.13) reveal the superior performance of our proposed approach. The referenced study reports 87.20% SMATCH F1 for parsing and 53.20 BLEU, 38.50 METEOR, and 87.50 COMET for generation on the Gold+Silver+Copper (G+S+C) dataset. In contrast, our Italian model (G+S+C baseline) achieves 89.22% SMATCH F1, 56.46 BLEU, 40.48 METEOR, 90.02 COMET, 70.38 chrF, and 92.72 BERTScore on the same dataset, representing significant advancements across all metrics.

The most notable results are observed in the G+S+C dataset experiments. Verb Augmentation (Verb Aug) achieves the highest parsing score of 90.56% SMATCH F1, while Tense Augmentation leads in generation with scores of 57.48 BLEU, 40.95 METEOR, 90.97 COMET, 70.88 chrF, and 92.97 BERTScore. These results not only surpass previous benchmarks but also approach the performance metrics of English, a high-resource language, despite comparatively limited lexical resources for Italian. These experimental outcomes provide strong evidence that data augmentation can significantly enhance the performance of semantic parsing and text generation models for the Italian language.

4.5.3 Urdu

Table 4.14 presents Urdu semantic parsing and text generation results while comparing them with Italian, German, and Dutch as low-resource baselines and English serving as a high-resource language benchmark for our experimental evaluation (See Exp. 1–4 in Table 4.14).

We acknowledge that comparing Urdu with European low-resource baselines is not an equitable comparison due to their distinct linguistic families. However, our objective was to assess how closely we could approach languages with Latin-based writing systems and lexical databases supporting their concepts in meaning representation, similar to English WordNet. A comparison of Urdu with other low-resource languages that follow right-to-left writing patterns, such as Arabic, is not feasible due to the absence of DRS-Text pairs for these languages in the PMB dataset.

Experiment 5 of Table 4.14 presents Urdu semantic parsing and generation results without data augmentation. Our parsing results in Exp. 5 are comparable to German and Italian and surpass Dutch. Regarding generation results: BLEU scores exceed those of Italian, German, and Dutch; BERTScores are comparable to all three low-resource languages; METEOR scores surpass the low-resource languages; ROUGE scores outperform Italian and Dutch; and chrF scores are comparable to the low-resource languages. In our experimental design, we maintained the default PMB split adopted for Italian and Dutch rather than equalizing the number of examples in the development and test sets.

Table 4.14: Urdu semantic parsing and generation results of byT5 with multi-lingual baselines and augmentation on PMB-5.0.0. The best results for Urdu (compared with LRLs only) are underlined. † represents statistical significance with regard to the model without augmentation using *Wilcoxon Signed Rank Test* (Aug = Augmentation; Adj = Adjective; Adv = Adverb; PN = Proper Noun; CN = Common Noun; Comp = Compound).

Exp.	**Impl. Type**	**Parsing Results**	**Generation Results**				
		SMATCH (F1%)	BLEU	BERTScore	METEOR	ROUGE	chrF
1	Italian	69.49	30.57	87.21	28.02	53.90	50.47
2	German	73.00	34.14	88.24	30.07	59.53	53.72
3	Dutch	42.77	19.83	84.98	25.36	51.78	46.92
4	English	91.42	71.89	96.01	54.52	86.38	83.80
5	*Urdu*	67.12	45.85	85.49	42.92	56.22	41.47
6	Urdu Adj Aug	69.49	48.79	86.34	44.93	58.59	43.42
7	Urdu Adv Aug	70.46	48.82	86.46	44.94	59.27	43.85
8	Urdu Verb Aug	70.75	49.69	86.81	45.15	59.70	44.12
9	Urdu PN Aug	71.82	49.94	86.73	46.24	60.05	44.66
10	Urdu CN Aug	70.28	49.83	86.62	46.01	60.03	44.49
11	Urdu Tense Aug	74.29	50.21	86.64	46.76	60.41	45.18
12	Urdu Comp Aug	76.81†	52.14†	87.48†	49.21†	62.08†	47.65†

We posit that score variations may be influenced by the dataset size (see Table 4.8) and the absence of Urdu-specific evaluation metrics. Moreover, Urdu's morphologically rich nature affects the scores of automatic evaluation measures.

Given the data-intensive nature of neural networks, we implemented six distinct lexical, grammatical, and named-entity-based augmentation approaches to the Urdu dataset. Experiments 6–11 present evaluation scores for all augmentation approaches applied to Urdu data for both semantic parsing and generation tasks. To analyze the impact of data augmentation, we adopted two procedures: (1) application of individual augmentation types (e.g., adjective or adverb replacement) to assess their specific effects on model performance; and (2) combination of all augmentation approaches (see Exp. 12, referred to as compound augmentation) to evaluate the overall impact of augmentation on model performance.

The experimental results consistently demonstrate that data augmentation enhances model performance for both parsing and generation tasks. Comparing individual augmentation results (Exp. 6–11) with Urdu results without augmentation (Exp. 5) reveals consistent improvements in evaluation scores. The variation in scores among different augmentation results is attributed to the lexical alterations performed on the dataset examples. Compound augmentation (Exp. 12) yields the most substantial improvement in evaluation measures. Comparing no augmentation (Exp. 5) with compound augmentation (Exp. 12) reveals, for parsing, an increase of 9 points in SMATCH F1-score, and for generation, increases of 6 points in BLEU, 2 points in BERTScore, 6 points in METEOR, 5 points in ROUGE, and 6 points in chrF.

Comparing overall augmentation performance with multilingual low- and high-resource evaluations, the compound augmentation strategy achieved superior semantic parsing results for low-resource comparisons and approached the upper-bound results for English semantic parsing (see Exp. 1–4 and Exp. 12). For the generation task, we

achieved superior results in BLEU, METEOR, and ROUGE, and improved BERTScores relative to Italian and Dutch.

4.6 Analyzing Examples through Finer-Evaluation

To acquire a comprehensive understanding of data augmentation's impact on neural semantic parsing and text generation performance, we conduct an in-depth analysis of the parser's semantic graphs (Figure 4.5) through node-level and edge-level evaluations. Additionally, we assess the quality of generated text via human evaluation. Subsequently, this helps in identifying the common errors made by the best model (byT5) in both semantic parsing and text generation tasks.

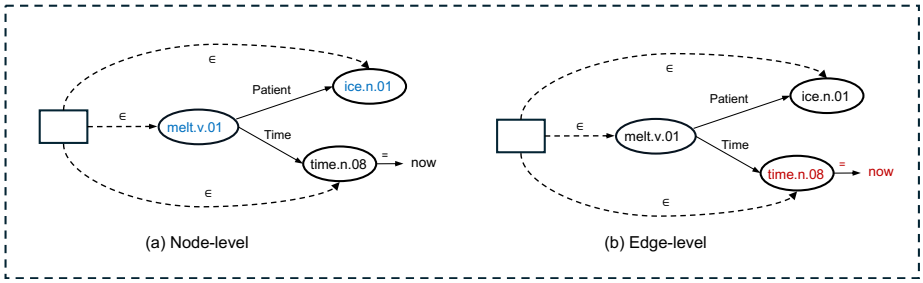


Figure 4.5: A graphical representation of the DRS for the sentence "The ice is melting." Two nodes, depicted in blue, represent individual concepts, while the edge, shown in red, is evaluated as a graph triples.

4.6.1 Fine-grained Analysis for Semantic Parsing

We refine our methodology by examining individual graph components for this semantic parsing graph analysis. This approach is based on previous evaluations of Chinese DRS parsing [Wang et al., 2023a], wherein nodes and edges within the graph were independently assessed. We calculate precision (P), recall (R), and F1-score (F1) for each type of parsed information at both node and edge levels.

Analyzing English Examples

As illustrated in Figure 4.5, individual nodes such as "ice.n.01" and "melt.v.01" are classified as Noun and Verb nodes, respectively, for the text "The ice is melting." For both node and edge-level evaluations, precision, recall, and F1 scores are computed based on the overlap between the reference and the model-generated entities. Edge-level evaluation, such as the triple "time.n.08 EQU now" in Figure 4.5, follows the same metrics by comparing the matching triples between the model's output and the Gold reference.

Table 4.15 presents a detailed comparison of semantic parsing task performance with and without augmentation on Gold-Silver-PMB. The results demonstrate a general

performance improvement across all edge and node types when data augmentation is applied. Notably, the parsing of Concepts — comprising Nouns, Verbs, Adjectives, and Adverbs — is significantly impacted by augmentation, particularly for adjectives and adverbs. These aspects pose challenges for DRS parsing due to the requirement of contextual awareness of WordNet and external lexical knowledge for accurate prediction.

Table 4.15: Fine-grained analysis of nodes and edges through semantic graph for English results with and without augmentation. All scores are listed in (%). Note: underline indicates that all values are improved.

Evaluation Type	Metric Type	Without Augmentation			With Augmentation		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score
Node Level	Names	84.30	82.30	83.30	84.50	82.60	83.50
	Negation	88.00	95.70	91.70	90.50	96.80	93.50
	Discourse	93.40	95.90	94.70	94.70	96.00	95.40
	Roles	93.40	93.00	93.20	94.40	93.70	94.10
	Members	99.00	98.80	98.90	99.30	98.90	99.10
	Concepts	87.40	87.20	87.30	88.20	87.80	88.00
	Noun Concept	90.50	90.10	90.30	91.30	90.60	90.90
	Adjective Concept	76.50	78.06	77.06	81.20	82.00	81.60
	Adverb Concept	78.90	83.30	81.10	82.10	86.80	84.40
	Verb Concept	78.70	78.60	78.65	78.80	78.70	78.75
Edge Level	Roles	88.00	87.90	87.90	88.90	88.60	88.80
	Name	89.20	87.10	88.20	89.90	88.00	89.00
	Members	94.10	94.00	94.10	94.60	94.20	94.40
	Operators	95.10	95.40	95.25	95.90	95.50	95.70
	Discourses	93.80	95.80	94.80	94.20	96.10	95.10

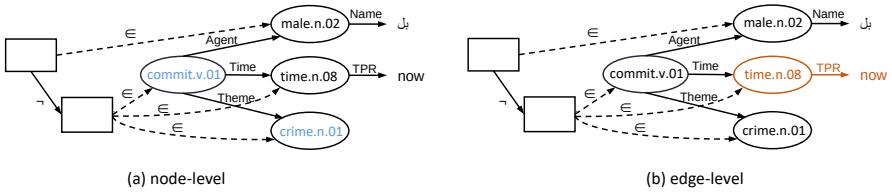
Node-level evaluation results indicate substantial improvements in F1-scores for correct concept prediction, particularly in Negation (1.8%), Roles (approximately 1%), and Concepts—specifically adjectives (4%) and adverbs (3.3%). Names, Discourse, Members, and Concepts with Nouns and Verbs also exhibited significant enhancements. Similarly, the edge-level evaluation revealed promising improvements in Roles-triple and Names-triple (approximately 1%), while Members-triple, Operator-triple, and Discourse-triple showed modest gains. We hypothesize that the limited improvement in these latter categories is due to the simplistic nature of these triples, exemplified by the predominance of basic operators such as "TPR", "TSU", or "EQU" in the dataset, simplifying their prediction.

Analyzing Urdu Examples

For fine-grained analysis of Urdu examples, Figure 4.6(a) illustrates node-level evaluation, where "crime.n.01" is categorized under Nouns and "commit.v.01" under Verbs. Edge-level evaluation metrics are determined based on the count of overlapping triples in the compared graphs, as exemplified by the orange triplets in Figure 4.6(b). For instance, time.n.08 TPR now constitutes an operator-triple. Figure 4.6 lists an example indicating ("بل نے جرم نہیں کیا۔" or "Bill didn't commit the crime.") for both the node-level and the edge-level graphical visualization and evaluation.

Table 4.16 presents fine-grained results comparing the Urdu parsing task with and without augmentation. The performance improves at both node-level and edge-level

Figure 4.6: Meaning representation of sentence "Bill didn't commit the crime." of fine-grained evaluation in node-level and edge-level. We highlight two examples in Nouns and Verbs in blue in (a) and one operator-triple in orange in (b).



when augmentation is incorporated. At the node-level, augmentation notably impacts Concepts, which encompass Nouns, Verbs, Adjectives, and Adverbs. These elements are particularly challenging in DRS parsing, as predicting these concepts from WordNet demands external knowledge and contextual information. The data reveals a 5 percentage point improvement in F1-score for concept predictions due to augmentation. For edge-level evaluation, the Roles-triple, Members-triple, and Discourse-triple categories all demonstrate notable improvement. The relative lack of growth in Operators-triple can be attributed to its simplicity, with most operators in our dataset being "TPR", "TSU", or "EQU", making their prediction straightforward. In contrast, the other three triple categories show an increase of at least 2.5 percentage points in F1-Score.

Table 4.16: Fine-grained analysis of nodes and edges based semantic parsing results with and without augmentation for Urdu.

Evaluation Type	**Metric Type**	**Without Augmentation**			**With Augmentation**		
		Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
Node-level	Names	75.11	73.12	74.11	76.61	76.03	76.31
	Discourse	44.74	58.61	50.72	56.51	55.34	55.91
	Roles	77.93	77.91	77.92	81.01	81.72	81.42
	Concepts	57.61	56.32	56.91	63.23	60.71	61.92
	Noun	69.01	67.14	68.03	73.42	69.42	71.32
	Adjective	16.53	15.33	15.93	26.02	26.51	26.32
	Adverb	30.64	28.94	29.74	41.92	34.64	37.93
	Verb	27.31	27.62	27.42	35.71	36.54	36.13
Edge-level	Roles-triple	58.31	59.01	58.65	64.02	65.03	64.52
	Members-triple	79.24	77.73	78.53	82.43	79.71	81.03
	Operators-triple	86.23	85.55	85.82	86.84	80.04	85.94
	Discourses-triple	42.61	56.83	48.64	60.85	61.54	61.12

4.6.2 Human Evaluation for Text Generation

Our evaluation of model-generated text is predicated on expert human review, wherein the text undergoes rigorous examination based on semantic, grammatical, and related phenomena-based criteria, culminating in a Robust Overall Semantic Evaluation (ROSE)

score. As defined by [Wang et al., 2021a], the ROSE score brings together the three binary evaluation metrics based on:

1. A semantic measure: assessing the generated text's accuracy in conveying the intended meaning relative to the gold reference. Suppose the gold reference sentence is "The cat is sitting on the mat." If the model generates "The cat is resting on the mat." this would be semantically correct because "resting" conveys a similar meaning to "sitting" in this context.

2. A grammatical measure ensuring the absence of orthographic or syntactic errors. Considering the same example, if the model generates "The cat is sit on the mat." this would be grammatically incorrect because "sit" should be "sitting" to agree with the auxiliary verb "is" A grammatically correct version would be "The cat is sitting on the mat." Grammatical correctness ensures that the text follows syntactic rules such as verb tense, subject-verb agreement, and sentence structure.

3. A phenomenon measure determining the presence of contextual control in comparison to the gold reference. If the gold reference sentence is "The cat sat on the mat before the dog arrived." and the model generates "The dog arrived before the cat and sat on the mat." this would violate phenomenon correctness because the generated sentence alters the sequence of events, which is critical in this context. Phenomenon correctness ensures that the generated text respects contextual and temporal aspects present in the gold reference, maintaining an accurate representation of events and their relationships.

The ROSE score is derived from the conjunction of these three metrics: a score of one is assigned if the text satisfies all criteria; otherwise, it receives a score of zero.

Evaluating English Examples

Table 4.17 presents different examples illustrating the assessment of semantics, grammar, and phenomena-based model-generated examples for English. These examples highlight different issues that arise in model-generated text and how they affect the overall quality of the output. In Example 1, "Was anybody killed?" is transformed into "Did they kill?" The generated sentence, while grammatically correct, changes the meaning of the question by shifting from a passive to an active construction, which alters the focus. This results in a semantic and phenomenon mismatch, leading to a 0 ROSE score. In Example 2, "I don't want pizza." becomes "I don't wanna want pizza." While the meaning is mostly retained, the generated sentence introduces a redundancy ("wanna want") that affects grammatical correctness. This grammatical issue, despite the correct phenomenon match, lowers the ROSE score to 0.

On the other hand, Example 3, "I think she's forty." is rendered as "I think that she is 40." which maintains both the meaning and context, while also being grammatically correct. This results in a perfect ROSE score of 1, as all aspects—semantic, grammatical, and phenomenon—are preserved. Similarly, in Example 4, "Add 3 spoonfuls of white wine." becomes "Add three spoonfuls of white wine." The only difference is the numeric format, but this does not affect the meaning or context, and both sentences are grammatically sound, leading to another perfect score. These examples illustrate how subtle

shifts in meaning, context, or grammar can significantly impact the evaluation scores, even when the generated text seems close to the reference.

Table 4.17: Semantics, Grammatical, and Phenomena-based binary evaluation of ROSE for model-generated examples concerning reference text. Note: Sem. = Semantics; Gram. = Grammatical; and Phen. = Phenomenon.

Example	Reference Text	Generated Text	Sem.	Gram.	Phen.	ROSE
1	Was anybody killed?	Did they kill?	0	1	0	0
2	I don't want pizza.	I don't wanna want pizza.	1	0	1	0
3	I think she's forty.	I think that she is 40.	1	1	1	1
4	Add 3 spoonfuls of white wine.	Add three spoonfuls of white wine.	1	1	1	1
5	He is not from Hokkaido.	He's not from Hokkaido.	1	1	1	1

Table 4.18 presents the mean ROSE scores for a subset of 50 randomly selected sentences from the test set, as evaluated by three reviewers.

The evaluation is based on a comparison of the performance between baseline model (byT5 without augmentation), our best model (byT5 with augmentation), and two leading LLMs, ChatGPT-3.5 and Claude-3.0, both in zero-shot configurations. In terms of Semantics, the byT5 model with augmentation achieves the highest score at 79.33%, outperforming the baseline byT5 without augmentation (76.66%) and significantly surpassing both ChatGPT (29.33%) and Claude (43.33%). This demonstrates that augmenting the data improved the model's ability to accurately capture the intended meaning of the text. For Grammatical Structure, all models, including byT5 without augmentation and ChatGPT, performed exceptionally well, with ChatGPT and Claude scoring 98.00% and the augmented byT5 and Claude achieving 100.00%. This indicates that all models generate grammatically sound sentences, with augmentation offering a slight improvement. When evaluating the Phenomenon (contextual alignment between generated and reference text), the augmented byT5 model again leads with 90.66%, while the baseline model scored 83.33%. ChatGPT and Claude showed low performance in this category, with ChatGPT achieving 35.33% and Claude 55.33%, highlighting the gap in contextual understanding compared to the fine-tuned models. The overall ROSE Score, which combines these three metrics, reveals that byT5 with augmentation provides the best overall performance at 78.00%, followed by the baseline model at 72.00%. In contrast, ChatGPT and Claude scored 29.33% and 43.33%, respectively, indicating that although LLMs perform well in grammatical accuracy, their semantic and contextual understanding lags behind the specialized models fine-tuned with augmented data. The results underscore the effectiveness of data augmentation in improving semantic parsing and text generation.

Evaluating Urdu Examples

Our evaluation of model-generated Urdu text focused on semantics and grammaticality only. Due to the morphologically rich nature of Urdu, for ROSE evaluation, we did not compute the phenomena measure. For ROSE analysis, we randomly selected a sample of 100 examples and compared the model-generated text with human reference text (gold test set). Perfect: indicating an exact match between model-generated text and human reference text. Given Urdu's morphologically rich nature, multiple Urdu words may

Table 4.18: Expert evaluation of English text based on Semantics, Grammatical Structure, and Phenomenon for byT5 without augmentation, LLMs (ChatGPT and Claude), and our best (augmented) model on a subset of 50 examples randomly chosen from the test set. Bold highlights the best scores. All scores are listed in (%).

Model Type	Semantics	Grammatical	Phenomenon	ROSE
byT5 without augmentation	76.66	98.00	83.33	72.00
ChatGPT-3.5 (zero-shot using prompt)	29.33	98.00	35.33	29.33
Claude-3.0 (zero-shot using prompt)	43.33	100.00	55.33	43.33
byT5 with augmentation (our best models)	**79.33**	**100.00**	**90.66**	**78.00**

reflect the same meanings. Our perfect score neglects this property, assigning a score of 1 only to text that exactly matches the human reference (see Table 4.19 for examples and evaluation scores). For ROSE analysis, we employ the following scoring criteria: (1) Semantics: assessing the preservation of the sentence's contextual meaning. A score of 1 is assigned for semantic relevance, 0 otherwise. (2) Grammaticality: evaluating the sentence's grammatical structure in terms of syntactic structure and word errors. A score of 1 is assigned if the sentence is grammatically correct and error-free (but possibly nonsensical), 0 otherwise. (3) ROSE: representing the product of semantics and grammaticality binary scores. A sentence correct in both aspects receives a score of 1, 0 otherwise.

In Table 4.20, Perfect and ROSE evaluations are based on manual analysis for the Urdu generation task. We have listed 4 different cases each reporting: (1) Perfect (P): all those examples that have the same model-generated text as listed in the gold examples; (2) Semantics (S): representing those examples that are semantically correct only; (3) Grammaticality (G): examples that are grammatically correct but not sustaining the same semantic information; and (4) ROSE (R): that is the product of semantic and grammatical evaluation scores. We also evaluated this sub-test-set through automatic evaluation measures to understand the correlation between human evaluation and automatic measures. Table 4.20 presents results for human evaluation and automatic evaluation of Urdu generation results with and without augmentation. It is important to note that our objective is not to surpass other low-resource languages in the dataset but to assess the model's performance for the Urdu meaning bank and the impact of data augmentation on model performance. Consequently, we have not conducted ROSE evaluation for other languages.

Table 4.20 demonstrates a more than 2.5-fold increase in perfect match scores through augmentation. Augmentation also improved semantic scores by 8 points, grammatical scores by 8 points, and ROSE evaluation scores by 7 points. The generated text without augmentation clearly exhibits errors resulting in lower scores. By providing more contextually similar and semantically correct examples through augmentation, model generalizability has been enhanced. These scores correlate with the automatic metrics (Table 4.20), thus indicating a reliable evaluation for the Urdu generation task.

Table 4.19: Perfect and ROSE evaluation based on manual analysis for Urdu generation task. We have listed 4 different cases each reporting: (1) *Perfect*(P): all those examples that have the same model generated text as listed in the gold examples; (2) *Semantics*(S): representing those examples that are semantically correct only; (3) *Grammaticality*(G): examples that are grammatically correct but not sustaining the same semantic information; and (4) *ROSE*(R): that is the product of semantic and grammatical evaluation scores. Note: for the first 2 columns, we have mentioned the English translations of the Urdu text (in double quotes) for understanding purposes.

Gold Example	**Model Generated Text**	**Manual Analysis**			
		P	S	G	R
کیا آپ نے اپنے آپ کو تکلیف پہنچائی؟ "Did you hurt yourself?"	کیا آپ کو آپ کے ساتھ درد کر رہا تھا؟ "Were you in pain with you?"	0	1	0	0
ٹام نے پاس ورڈ درج کیا۔ "Tom entered the password."	ٹام نے پاسورڈ کے ذریعے داخل ہوا۔ "Tom entered through the password."	0	1	0	0
میدان جنگلی پھولوں سے بھرے ہوئے ہیں۔ "The grounds are full of wild flowers."	اس درخت میں سبزیاں پھول رہی ہیں۔ "Vegetables are blooming in this tree."	0	0	1	0
اس کا کتا بہت تیز نہیں ہے۔ "His dog is not very fast."	اس کے کتے بہت سست نہیں ہیں۔ "His dogs are not very lazy."	0	0	1	0
خبر سن کر ہم سب حیران رہ گئے۔ "We were all shocked to hear the news."	اس خبر پر ہم سب سے حیران تھے۔ "We were all surprised at this news."	0	1	1	1
ٹام نے قہقہہ لگایا۔ "Tom laughed."	ٹام نے ہنسا۔ "Tom laughed."	0	1	1	1
ٹام نے گٹار بجایا اور مریم نے گایا۔ "Tom played the guitar and Mary sang."	ٹام نے گٹار بجایا اور مریم نے گایا۔ "Tom played the guitar and Mary sang."	1	1	1	1
میرا گولف بیگ کہاں ہے؟ "Where's my golf bag?"	میرا گولف بیگ کہاں ہے؟ "Where's my golf bag?"	1	1	1	1

Table 4.20: Analysis of generation results for Urdu (%) with and without augmentation.

		Human Evaluation			**Automatic Metrics**				
Impl. Type	Perfect	Sem.	Gram.	ROSE	BLEU	B-Score	METEOR	ROUGE	chrF
Urdu w/o aug	10	39	66	36	46.05	85.75	42.70	56.12	41.23
Urdu with aug	**26**	**47**	**74**	**43**	**49.66**	**86.83**	**47.20**	**59.57**	**45.87**

4.7 Comparing with LLMs

To further evaluate the efficacy of our semantic parsing and text generation models, we conducted a comparative analysis with two state-of-the-art general-purpose LLMs: (i) ChatGPT-3.5 by OpenAI [OpenAI, 2023]; and (ii) Claude-3-haiku by Perplexity Labs [Turpin et al., 2023]. The objective was to provide preliminary insights into the performance of our specialized models relative to general-purpose LLMs not specifically trained for semantic parsing and text generation tasks. We evaluated the LLMs using a zero-shot learning approach, which involved parsing and generation through prompting without any prior pre-training or fine-tuning of models. This analysis aimed to determine the relative performance capabilities of the various models. Since English demonstrated the highest performance among all language variants, we concentrated exclusively on English for comparison with LLMs and for conducting error analysis.

Table 4.21: English experimental results of byT5 Gold-Silver without augmentation, ChatGPT-3.5, Claude-3.0, and our best models (byT5 Compound Aug for Semantic Parsing and Adverb Aug for Generation on a subset of 100 examples from the test set. Bold indicates the best model. Note: MET. = METEOR and BERT = BERTScore.

Model Type	Semantic Parsing	Text Generation				
	SMATCH (F1)	BLEU	MET.	COMET	chrF	BERT
byT5 without augmentation	92.82	61.87	52.17	95.72	82.13	98.15
ChatGPT-3.5 (zero-shot using prompt)	15.87	1.87	21.50	67.23	32.78	90.75
Claude-3.0 (zero-shot using prompt)	63.25	0.74	19.08	46.97	23.72	88.12
byT5 with augmentation (our best models)	**93.83**	**67.22**	**55.64**	**95.86**	**83.54**	**98.49**

To gain insight into the performance of neural semantic parser and text generator models, we examined a sample of 100 sentences from the test set. Rather than selecting examples randomly, we chose the first 100 examples in the test set, which consisted of relatively short sentences. This selection was made to provide LLMs with simple and concise examples, facilitating their understanding of the complex DRS structure. Our investigation focused on the best-performing neural models, specifically: (i) byT5 without augmentation (first experiment in Table 4.21), and byT5 with augmentation — compound augmentation for semantic parsing and adverb augmentation for generation (fourth experiment in Table 4.21); and (ii) the outputs produced by ChatGPT-3.5 and Claude-3-haiku as zero-shot prompts in response to task-specific instructions. We evaluated the generated outputs using automatic evaluation measures, with results presented in Table 4.21. The experimental outcomes clearly demonstrate that while large language models like ChatGPT and Claude are powerful general-purpose tools, they underperform on low-resource, domain-specific tasks such as semantic parsing and text generation, particularly in the context of DRS. This underscores the necessity for neural models specifically developed and trained for semantic parsing (text-to-DRS) and natural language generation (DRS-to-text) tasks.

4.8 Analyzing Errors in Examples

To gain further insight into the evaluations, we conduct a comprehensive manual inspection of the DRS and text generated by various models. This process involves comparing identical samples generated by four distinct models: (i) fine-tuned byT5 without augmentation; (ii) ChatGPT; (iii) Claude; and (iv) our optimal byT5 model with augmentation. Table 4.22 presents errors based on the semantic parsing task, while Table 4.23 highlights textual errors generated by different comparable models for the DRS-to-text generation task.

4.8.1 Error Analysis for Semantic Parsing

During manual inspection of semantic parsing outputs, we categorize errors into two distinct types: (i) errors resulting in ill-formed DRS that cannot be converted into graph structures; and (ii) errors affecting the semantic content of the DRS. The former occurs when the parsed DRS produces an excessive number of tokens due to concept repetition,

resulting in an unnecessarily long sequence that does not align with the actual graph structure (underlined in Table 4.22). Another possibility arises from the insertion or omission of spaces, causing subsequent tokens to be incorrectly merged. The latter includes the generation of incorrect concepts, roles, indices, missing concepts or tokens (highlighted in brown), and the generation of extra irrelevant tokens (underlined) in the DRS.

Table 4.22 suggests that the model without augmentation underfits situations where information is based on very short sentences, e.g., "Liz Mohn", and begins adding or repeating incorrect concepts (highlighted in red). In some instances, the model struggles to predict exact sequence tokens and omits certain logical information (see concepts in brown).

Table 4.22: Error analysis of different model-parsed DRS concerning reference DRS. Reference DRS represents logical representation for the text: (a) Liz Mohn; (b) Is hexane toxic?; and (c) How's the dog? Note: underline indicates extra token; red color indicates wrong tokens; and brown color indicates missing tokens.

Reference DRS	**Model Type**	**Model Parsed DRS**
event.v.01 Participant +1 female.n.02 Name "Liz Mohn"	Without augmentation	event.v.01 Participant +1 female.n.02 Name "Liz Mohn" time.n.08 EQU now female.n.02 Name "Liz Mohn" Female -2 Time -1
	ChatGPT-3.5	event.v.01 Participant +1 female.n.02 Name "Liz Mohn"
	Claude-3.0	event.v.01 Participant +1 person.n.01 Name "Liz Mohn"
	With augmentation	event.v.01 Participant +1 female.n.02 Name "Liz Mohn"
time.n.08 EQU now hexane.n.01 toxic.a.01 Time -2 AttributeOf -1	Without augmentation	time.n.08 EQU now hexane_toxic.n.01 Time -1 AttributeOf -1
	ChatGPT-3.5	time.n.08 EQU now hexane.n.01 toxic.a.01 AttributeOf -2 Time -1
	Claude-3.0	time.n.08 EQU now hexane.n.01 toxic.a.01 Time -1 AttributeOf +2
	With augmentation	time.n.08 EQU now hexane.n.01 toxic.a.01 Time -2 AttributeOf -1
state.a.01 Time +1 EQU ? time.n.08 EQU now dog.n.01 Attribute -2	Without augmentation	state.a.01 Time +1 EQU ? time.n.08 EQU now dog.n.01 Attribute -2
	ChatGPT-3.5	state.a.01 Time +1 EQU ? time.n.08 EQU now dog.n.01 Attribute -2
	Claude-3.0	state.a.01 Time +1 EQU ? time.n.08 EQU now dog.n.01 Value -2
	With augmentation	state.a.01 Time +1 EQU ? time.n.08 EQU now dog.n.01 Attribute -2

General-purpose LLMs such as ChatGPT and Claude perform poorly in generating precise DRS sequences. For short sentences, these models underfit and fail to generate correct discourse referents (missing information is indicated in brown). In some cases, these models also struggle to generate correct indices for logical concepts (highlighted in red). Between these two models, ChatGPT performed particularly poorly, frequently neglecting most logical meaning representations. Claude, on the other hand, encounters problems with incorrect indices and roles (highlighted in red). Our best-augmented model performs accurately for short and medium-length DRS sequences, while for longer sequences, it occasionally faces challenges in missing or generating

tokens, misinterpreting exact indices, or swapping roles in the DRS.

4.8.2 Error Analysis for Text Generation

During manual inspection of generated text, we present examples of various properties such as semantics, grammar, omissions of information, LLM hallucinations, extraneous information, and flawlessly created instances in Table 4.23. Our evaluation of the model-generated text is based on the following criteria: (i) errors (marked in red); (ii) irrelevant additional information (underlined); and (iii) semantically valid modifications with unique textual expressions (marked in blue).

Table 4.23: Error analysis of different model-generated text concerning reference text. Note: underline indicates extra irrelevant information; red indicates wrong information; and blue indicates semantically correct but not exactly the same information.

Reference Text	Model Type	Model Generated Text
We're not scared of anybody.	Without augmentation	We're not scared of people.
	ChatGPT-3.5	The person, who is speaking or the subject of discussion, is currently experiencing fear at this moment.
	Claude-3.0	The speaker is not scared at the present time.
	With augmentation	We are not scared of anybody.
I owed him a thousand dollars.	Without augmentation	I owed him about 100 dollars.
	ChatGPT-3.5	The person (the speaker) owes one thousand dollars to a male individual at the present moment.
	Claude-3.0	The speaker owes someone a thousand dollars at the present time.
	With augmentation	I owed him a 1000 dollars.
Are you 17 years old?	Without augmentation	Is you seventeen years old?
	ChatGPT-3.5	At the present moment, the hearer is seventeen years old.
	Claude-3.0	The hearer is 17 years old at the present time.
	With augmentation	Are you seventeen years old?

Table 4.23 contains examples demonstrating three critical components of NLG: negation, question construction, and quantification. The textual responses in Table 4.23 indicate that the model without augmentation struggled to effectively represent the intended semantics of the sentences, with some instances exhibiting entirely incorrect interpretations (marked in red). Furthermore, the unaugmented model struggled to manage quantification while maintaining grammatical precision within phrases.

Both ChatGPT and Claude performed poorly on this task, as neither model was capable of producing appropriate translations for examples specified in the DRS formalism. Analysis of the samples revealed that, rather than producing exact translations, the models attempted to explain the logical representation of the DRS, resulting in irrelevant information (underlined). This behavior is likely attributable to the absence of semantic or formal meaning representation during the training of these large language models (LLMs).

The best augmentation model accurately captured the semantic and grammatical complexities of the examples but encountered some difficulty in generating the exact information presented in the test set. These relatively minor modifications (highlighted in blue) to the model-generated text do not significantly impact human evaluation, as

the generated text retains the identical meaning, semantics, and grammatical structure of the phrases (refer to examples in Table 4.18 for ROSE evaluation). However, these changes may result in lower scores in automatic evaluations due to the lack of exact word-for-word matching between the original and generated text pairs.

4.9 Chapter Conclusion

This investigation has empirically demonstrated the efficacy of multi-faceted data augmentation techniques in enhancing the performance of semantic parsing and text generation tasks across three typologically distinct languages: English, Italian, and Urdu. By implementing named entity augmentation, lexical substitutions, and grammatical transformations—each carefully tailored to the linguistic characteristics of the target languages—we observed significant improvements in the generalization capacity and robustness of the neural models. Our methodological approach, encompassing out-of-context named entity augmentation, WordNet-based lexical substitutions, and tense-oriented grammatical transformations, has proven efficacious in preserving semantic fidelity and contextual coherence, thereby augmenting the models' capacity to assimilate complex grammatical, semantic, pragmatic, and world knowledge constructs.

In the English language context, our methodologies yielded substantial performance enhancements in both semantic parsing and text generation tasks, underscoring the critical role of high-quality, contextually appropriate data in neural model training. For Italian, our cross-lingual data augmentation strategy, leveraging resources such as English WordNet, successfully mitigated the challenges inherent in low-resource NLP, culminating in the state-of-the-art performance in DRS parsing and generation. The Italian-specific augmentations, including the accommodation of syntactic flexibility and grammatical nuances, exemplify the efficacy of our approach in facilitating semantic knowledge transfer to this language.

In the case of Urdu, we devised a novel semantically annotated corpus through the transformation of English-aligned DRS into Urdu-aligned DRS, further enriched by lexical, grammatical, and named-entity-based augmentation techniques. This innovative approach facilitated significant performance gains, elevating Urdu closer to the benchmarks established for high-resource languages such as English, and surpassing those of other low-resource European languages. Our findings suggest that these augmentation techniques not only enhance model performance but also hold promise for application in other low-resource languages, particularly those characterized by SOV word order and right-to-left orthography.

Furthermore, through this research, we underscore the imperative for specialized neural models tailored to specific tasks such as semantic parsing and text generation. While large-scale language models like ChatGPT and Claude offer versatile capabilities, they exhibit limitations in generating precise formal meaning representations, such as DRS, when evaluated in zero-shot scenarios. This work establishes a foundation for further exploration in enhancing meaning representations and developing linguistic resources for underrepresented languages in the field of natural language processing.

Limitations: While this study demonstrates substantial advancements in semantic pars-

ing and text generation through data augmentation techniques, several constraints warrant acknowledgment. Primarily, the experimental focus was limited to fine-tuning models pre-trained for general-purpose or task-specific applications, excluding those explicitly designed for DRS-based semantic parsing or text generation. The observed outcomes suggest that more substantial performance enhancements could potentially be realized through the development and implementation of neural models pre-trained specifically on DRS-text pairs prior to task-specific fine-tuning.

Furthermore, although in this study we successfully generated a semantically annotated corpus and exhibited significant performance improvements, the inherent limitations associated with low-resource languages persist. These constraints include the lack of comprehensive lexical resources and the complexities arising from unique syntactic structures, which continue to impede the full realization of the models' potential. Addressing these challenges necessitates further development of linguistic resources and the formulation of more sophisticated augmentation strategies tailored explicitly to the distinctive characteristics of similar low-resource languages. Such advancements would contribute to a more comprehensive and nuanced approach to natural language processing across diverse linguistic contexts.

Chapter 5

Evaluating Structural and Linguistic Quality in DRS-based Parsing and Generation through Bidirectional Evaluation

Evaluating Discourse Representation Structure (DRS)-based semantic parsing and text generation systems presents complex challenges, as traditional metrics often fall short in capturing both structural and linguistic fidelity. This chapter introduces two complementary bidirectional evaluation methodologies designed to provide a holistic assessment of DRS-based systems. The Parse-Generate (Pars-Gen) approach enhances traditional parsing evaluations by generating text from parsed DRSs, which enables the identification of linguistic phenomena that structure-focused metrics like SMATCH might miss. In parallel, the Generate-Parse (Gen-Pars) approach mitigates limitations of surface-level text generation metrics—such as BLEU, METEOR, and BERTScore—by transforming generated text back into DRS representations, allowing for a semantic assessment that captures structural and relational integrity. Using the Parallel Meaning Bank dataset, our evaluation spans English, Italian, and Urdu, uncovering unique insights into the interplay between structural and linguistic measures across languages. Findings highlight that, while SMATCH captures role-based structural overlaps, it may overlook finer details of linguistic expression, and generation metrics often miss core semantic equivalences. This chapter advances DRS evaluation by offering a dual approach that more accurately reflects system performance, providing robust tools for comprehensive assessments in both semantic parsing and text generation.

Chapter adapted from

1. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2025. Evaluating Structural and Linguistic Quality in Urdu DRS Parsing and Generation through Bidirectional Evaluation. In Proceedings of the First Workshop on Natural Lan-

guage Processing for Indo-Aryan and Dravidian Languages, pages 33–43, Abu Dhabi. Association for Computational Linguistics.

5.1 Introduction

DRS is central to advanced semantic processing, providing a framework for capturing nuanced linguistic elements beyond simple text interpretation [Kamp and Reyle, 1993]. Being language-neutral, DRS models complex semantics like negation and quantification [Kamp and Reyle, 1993, Jaszczolt and Jaszczolt, 2023] and is used in processing various languages without language-specific constraints. This flexibility makes DRS essential for building multilingual NLP systems that need a unified way of representing meaning across various languages.

Semantic parsing [van Noord et al., 2018, Noord, 2019, van Noord et al., 2019] and generation [Wang et al., 2021b, Amin et al., 2022b, Liu et al., 2021, Amin et al., 2024] are the core applications of the DRS processing. A distinguishing feature of these tasks is their reversibility: the output of semantic parsing can serve as the input for text generation and vice versa. This bidirectional relationship between parsing and generation opens avenues for evaluation, suggesting that one process's output quality could be validated through its reverse operation. However, while the reversibility implies unique evaluation potential, it also underscores challenges in current methodologies, as traditional evaluations often lack the capability to capture the complete spectrum of structural and linguistic intricacies in these tasks, thus necessitating innovative evaluation paradigms.

Semantic parsing evaluation typically relies on structural metrics like SMATCH, which assesses roles or concepts-based overlaps between predicted and reference DRS graphs [Kamp et al., 2010]. While valuable for evaluating structural accuracy, this metric often miss essential linguistic subtleties and penalize the overall evaluation. For instance, for the text "Thirteen people were arrested.", two DRS representations like "quantity.n.01 EQU 13 person.n.01 Quantity -1 arrest.v.01 Patient -1 Time +1 time.n.08 TPR now" and "quantity.n.01 EQU 30 person.n.01 Quantity -1 arrest.v.01 Patient -3 Time +1 time.n.08 TPR now" with minor structural divergences, such as `Quantity` and `Index`, obtained a significantly low SMATCH scores despite near-identical semantics (see example 4 in Table 5.2). Such distinctions illustrate how structural metrics alone may fall short in capturing the semantic nuances, coherence, and pragmatic meaning crucial to linguistic representation. This limitation inspired the development of the PARS/PARSGEN approach, which leverages text generation to assess parsing quality, highlighting linguistic phenomena that structural metrics might otherwise overlook.

Text generation from DRS also poses unique evaluation challenges. Traditional metrics like BLEU, METEOR, and even recent metrics like COMET and BERTScore prioritize surface-level similarities between generated and reference texts. However, given the diversity of natural language, there can be multiple valid expressions for the same meaning. For example, with the sentences "A room" and "The room," traditional surface-level evaluations don't see these as a perfect match and therefore give them low scores. However, when we convert them into their meaning representation, such as DRS, they receive an SMATCH score of 1, recognizing them as a perfect match both linguistically and structurally (see example 1 in Table 5.3 and Table 5.9). Furthermore, two outputs with different syntactic structures or vocabularies might convey the same meaning, yet traditional metrics may penalize these legitimate variations. This issue is especially apparent in cases where syntactic variations preserve semantic equivalence,

as with active-passive transformations or paraphrasing. To address this, we propose the GEN/GEN-PARS paradigm, which evaluates generated text by parsing it back into DRS, offering a structural evaluation perspective that complements surface-level metrics.

The majority of existing research on semantic parsing and generation has focused primarily on English, with separate evaluation approaches for each language. This singular focus has limited our understanding of metric efficacy across languages with distinct structural properties. Although advances in pre-trained language models have significantly enhanced NLP capabilities, evaluation in semantic parsing and text generation has not fully capitalized on these developments, especially in integrating structural and linguistic elements within meaning representation [Amin et al., 2024]. Despite recent progress, both DRS-based semantic parsing and text generation remain intricate and error-prone [Wang et al., 2023a], necessitating more comprehensive evaluation frameworks for a thorough understanding of model performance.

These complexities are magnified in cross-linguistic semantic representation. While DRS strives for a language-neutral approach, assessing its accuracy across diverse languages demands methods that account for both language-specific structural and linguistic variations. This need becomes crucial when working with languages that diverge significantly in syntactic structure and semantic expression—such as English, Italian, and Urdu. The proposed PARS/PARS-GEN and GEN/GEN-PARS evaluation paradigms aim to address this by providing a framework that evaluates both structural accuracy and linguistic adequacy across languages, yielding insights into how semantic representation quality varies across linguistic boundaries.

To further enhance our evaluation approach, we conducted human assessments using Robust Overall Semantic Evaluation (ROSE), which provides a binary evaluation of semantic accuracy, grammatical correctness, and linguistic phenomena in the generated text. Examples that meet all criteria are assigned a score of 1 (see chapter 4, Section 4.6.2 for detailed description of ROSE). Additionally, we performed a comprehensive correlation analysis, using Pearson correlation (for PARS/PARS-GEN and GEN/GEN-PARS relationships), point-biserial correlation, and Spearman correlation between ROSE and GEN scores. These statistically significant correlations, together with human evaluation, further validate the use of alternative evaluation methods that account for both structural and linguistic qualities in assessing DRS-based semantic parsing and text generation.

5.1.1 Research Objectives and Contributions

Our research addresses fundamental challenges in evaluating DRS-based systems through several interconnected objectives. The main goal is to develop evaluation frameworks that bridge the gap between structural and linguistic assessments of DRS-based systems. This entails designing evaluation methods that capture both structural accuracy in semantic representations and their effectiveness in maintaining linguistic meaning. We particularly aim to explore the impact of semantic parsing accuracy on text generation quality across different languages, recognizing that structural precision may influence linguistic output in language-specific ways.

A core objective is to develop novel evaluation paradigms that leverage the bidirectional relationship between parsing and generation. With PARS/PARS-GEN, we aim to evaluate parsing quality by examining the linguistic coherence of text generated from

parsed DRS, uncovering insights beyond those provided by structural metrics. Conversely, GEN/GEN-PARS enables the assessment of generation quality by analyzing the semantic consistency of the parsed structure derived from generated text, providing a deeper perspective than surface-level metrics alone.

Our research also seeks to establish new cross-linguistic evaluation methodologies that consider both structural and semantic dimensions. This objective is particularly significant, given that different languages may represent the same meaning through various syntactic patterns and lexical choices. By examining these patterns across English, Italian, and Urdu, we aim to understand the performance of evaluation metrics in linguistically diverse contexts.

This research addresses several important questions aimed at enhancing the evaluation of semantic parsing and text generation. First, we explore how to move beyond existing structural and surface-level metrics to more accurately assess DRS-based semantic parsing and generation tasks. We investigate the relationship between structural accuracy in semantic parsing and the resulting linguistic quality in text generation, examining how structural fidelity impacts overall language output. Another key question focuses on the variation in evaluation challenges and error patterns across different languages, seeking to understand the unique difficulties presented by each linguistic context. Additionally, we exploit whether the reversible nature of semantic parsing and text generation can be leveraged to refine evaluation methods. Finally, we examine which types of errors our proposed approaches most effectively identify or address within each language, aiming to uncover patterns that could guide more targeted improvements.

This research makes significant contributions to semantic processing evaluation:

1. It introduces novel evaluation paradigms, PARS/PARS-GEN and GEN/GEN-PARS, which reveal unique insights into the language's syntactic variability and complex semantic structures that traditional metrics often overlook.

2. The PARS/PARS-GEN paradigm uses linearized text to mitigate non-optimal outcomes in SMATCH's greedy search algorithm, enabling a more intuitive and human-centered approach to parsing evaluation.

3. Through the GEN/GEN-PARS evaluation, it identifies semantic and syntactic issues at a node level, examining lexical DRS concepts like nouns, verbs, adjectives, and adverbs within the generated DRS to provide a granular view of the generation quality, ultimately facilitating a balanced metric that captures both structural and linguistic fidelity.

4. It proposes a detailed Pearson correlation analysis between PARS/PARS-GEN, GEN/GEN-PARS. The observed statistically significant correlations underscore the robustness of our approach and demonstrate the effectiveness of combining structural and linguistic assessments in DRS-based semantic processing.

Through these contributions, we presents a preliminary evaluation framework designed to enhance the assessment of DRS-based semantic parsing and text generation tasks. The methods and analyses serve as an initial exploration into bidirectional evaluation, setting the stage for more refined approaches in subsequent research.

Our cross-linguistic analysis provides valuable insights into the challenges of semantic parsing and generation across languages. By evaluating English, Italian, and Urdu, we clarify how evaluation metrics function across languages with differing syntactic structures and semantic representation patterns. This examination helps identify which evaluation aspects are language-specific and which are universally applicable. The practical implications of our work are substantial. Our findings offer actionable recommendations for improving semantic parsing and text generation systems. By identifying critical error types and exploring the optimal weighting or combination of metrics, we enhance system assessment accuracy. Our evaluation frameworks also provide new tools for debugging and optimizing semantic processing systems across different languages.

The remaining chapter is organized as follows: Section 5.2 discusses the limitations of current evaluation approaches in detail. Section 5.3 presents our novel evaluation methodologies and describes the experimental setup and implementation details. Section 5.4 presents our analysis through correlation and human evaluation. Finally, Section 5.5 concludes with implications and future directions.

5.2 Limitations of Current Evaluation Approaches

The evaluation of semantic parsing and text generation systems presents unique challenges that conventional metrics often struggle to address comprehensively. This section examines these limitations in detail and establishes the motivation for our proposed evaluation approaches.

5.2.1 Limitations of Semantic Parsing Evaluation

Semantic parsing, particularly with DRS, poses complex challenges that traditional evaluation metrics often struggle to fully address. Evaluation measures like SMATCH [Cai and Knight, 2013], SMATCH++ [Opitz, 2023], and SemBLEU [Song and Gildea, 2019] have laid a foundation for assessing structural similarities between predicted and reference DRS representations. However, these structural measures have notable limitations that prevent them from accurately evaluating the depth of parsing quality, especially as models become more adept at handling complex semantic relationships and processing multilingual data.

The most commonly used metric, SMATCH [Kamp et al., 2010], assesses semantic overlap by using a greedy hill-climbing algorithm to map nodes between predicted and reference DRS structures. This method does not ensure optimal alignment due to the NP-complete nature of the node mapping problem, often resulting in suboptimal solutions where node mappings lack optimal semantic or structural correspondence. For instance, in our analysis, we observed cases where SMATCH assigns a score of 00.00 to semantically valid DRS, such as the simple entity representation of "Liz Mohn," where the core semantic content is preserved but structural variations lead to complete penalization—see Example 1 in Table 5.1.

Moreover, SMATCH—and similar metrics like SMATCH++— approach DRS nodes as isolated entities, often overlooking the deeper semantic content and relationships between them. This limitation becomes evident in cases like "Nancy is very cute, isn't she?"

(see Example 3 in Table 5.1) despite capturing the core entity and attributes, SMATCH assigns a low score of 38.46. The metric struggles to properly assess the semantic equivalence when handling complex linguistic phenomena such as degree modifiers ("very") and negative tag questions, even though the essential meaning is preserved in the parsed output.

The limitations of structural metrics like SMATCH become more apparent when assessing a system's ability to represent linguistic nuances accurately. Consider the sentence "I woke up at eleven," where SMATCH assigns a score of 00.00 despite both the parsed and gold DRS containing the same temporal information and core event structure—referring to Example 2 in Table 5.1. The only difference lies in the subtle variation of verb sense (wake_up.v.01 vs. wake_up.v.02) and temporal representation, yet this results in complete penalization, highlighting how structural metrics can be overly sensitive to minor variations that do not significantly impact meaning.

Table 5.1: Structural overlap-based evaluation measures: highlighting limitations of SMATCH.

Ex. No.	Gold Text	PARS (DRS)	Gold DRS	PARS (SMATCH)
1	Liz Mohn	female.n.02 Name "Liz Mohn" time.n.08 EQU now female.n.02 Name "Liz Mohn" Female -2 Time -1	event.v.01 Participant +1 female.n.02 Name "Liz Mohn"	00.00
2	I woke up at eleven.	person.n.01 EQU speaker wake_up.v.01 Agent -1 Time +1 Time +2 Time +3 time.n.08 TPR now time.n.08 ClockTime 11:00	person.n.01 EQU speaker wake_up.v.02 Agent -1 Time +1 Time +2 time.n.08 TPR now time.n.08 ClockTime 11:00	00.00
3	Nancy is very cute, isn't she?	female.n.02 Name "Nancy" time.n.08 EQU now very.r.01 cute.a.01 AttributeOf -3 Time -2 Degree -1	NEGATION <1 NEGATION <1 female.n.02 Name "Nancy" time.n.08 EQU now very.r.01 cute.a.01 Time -2 Degree -1 Value + AttributeOf -3 NEGATION <2 NEGATION <1 very.r.01 cute.a.01 Degree -1 Value + Time +1 AttributeOf +2 time.n.08 EQU now female.n.02	38.46
4	Plant trees!	event.v.01 Participant +1 plant_tree.n.01	person.n.01 EQU hearer plant.v.01 Agent -1 Theme +1 tree.n.01	44.44
5	Is this appropriate?	be.v.01 Time +1 Theme +2 Co-Theme +3 time.n.08 EQU now entity.n.01 appropriate.n.01	time.n.08 EQU now entity.n.01 appropriate.a.01 Time -2 AttributeOf -1	64.00

These examples demonstrate the need for evaluation methods that can assess linguis-

tic quality in parsed outputs, especially when structural differences have minimal impact on meaning. The case of "Plant trees!" illustrates this clearly, where SMATCH assigns a score of 44.44 despite both representations capturing the core imperative action (see Example 4 in Table 5.1). The difference in scores stems primarily from variations in how the agent and theme roles are structured, rather than fundamental semantic differences.

Another major limitation of current evaluation metrics is their inability to effectively handle cross-linguistic variations. Metrics like SMATCH were developed primarily for English, limiting their adaptability to languages with differing syntactic or semantic structures. For example, languages like Italian and Urdu often encode temporal, causal, or syntactic relationships in ways that differ from English. Applying metrics like SMATCH to these languages may fail to capture valid semantic differences, leading to an unjust penalization of models that produce contextually appropriate but structurally distinct outputs.

This challenge is particularly significant for multilingual semantic parsing systems, which must reconcile the structural requirements of DRS frameworks with the linguistic intricacies of diverse languages. Our analysis of examples across different syntactic constructions—from simple entity mentions to complex questions and imperatives—reveals how current evaluation approaches consistently struggle to capture these linguistic nuances, resulting in evaluations that overlook a system's actual cross-linguistic parsing capabilities.

Table 5.2: Structural overlap-based evaluation measures: highlighting limitations of SMATCH for Urdu parsing. English translations are mentioned in brackets. PARS scores are in %.

Ex. No	Gold Text	Gold (DRS)	PARS (DRS)	PARS (SMATCH)
1	ٹام نے ایک نیا پک اپ خریدا۔ ("Tom bought a new pickup.")	male.n.02 Name "ٹام" new.a.05 AttributeOf +1 pickup.n.01 buy.v.01 Agent -3 Theme -1 Time +1 time.n.08 TPR now	male.n.02 Name "ٹام" new.a.01 AttributeOf +1 pick_up.n.01 buy.v.01 Agent -4 Beneficiary -3 Theme -1 Time +1 time.n.08 TPR now	00.00
2	ٹام مریم کو اپنے کتے کی تصویر دکھاتا ہے۔ ("Tom shows Mary a picture of his dog.")	male.n.02 Name "ٹام" female.n.02 Name "مریم" male.n.02 ANA -2 dog.n.01 Owner -1 picture.n.01 Topic -1 show.v.04 Agent -5 Recipient -4 Theme -1 Time +1 time.n.08 TPR now	male.n.02 Name "ٹام" female.n.02 Name "مریم" female.n.02 ANA -1 dog.n.01 Owner -1 photo.n.01 Creator -1 show.v.01 Agent -4 Recipient -3 Recipient -1 Time +1 time.n.08 TPR now	69.23
3	آج میری گردن میں درد ہے۔ ("Today I have a pain in my neck.")	day.n.03 TCT now time.n.08 TIN -1 person.n.01 EQU speaker neck.n.01 pain.n.01 Location -1 have.v.16 Time -4 Experiencer -3 Stimulus -1 Time +1 time.n.08 EQU now	person.n.01 EQU speaker neck.n.01 hurt.v.01 Patient -2 Patient -1 Time +1 time.n.08 EQU now	60.00
4	تیرہ افراد کو گرفتار کر لیا گیا۔ ("Thirteen people were arrested.")	quantity.n.01 EQU 13 person.n.01 Quantity -1 arrest.v.01 Patient -1 Time +1 time.n.08 TPR now	quantity.n.01 EQU 30 person.n.01 Quantity -1 arrest.v.01 Patient -3 Time +1 time.n.08 TPR now	00.00
5	میرے پاس بہت پیسہ ہے۔ ("I have a lot of money.")	person.n.01 EQU speaker money.n.01 Quantity + get.v.01 Pivot -2 Theme -1 Time +1 time.n.08 TPR now	person.n.01 EQU speaker have.v.01 Pivot -1 Theme +2 Time +3 quantity.n.01 EQU +1 quantity.n.01 EQU +1 money.n.01 Quantity + time.n.08 EQU now	57.89

For **Urdu**, SMATCH assigns a zero score to the DRS representation for ("*Tom bought a new pickup*"), despite the semantic content being essentially equivalent in both gold and predicted DRS. The low score is due to minor structural differences, underscoring a limitation of SMATCH's focus on structural alignment rather than semantic equivalence (Example 1, Table 5.2).

Additionally, SMATCH's handling of semantic relationships is limited, as it treats DRS nodes as isolated entities. This limitation is evident in Example 2, ("*Tom shows Mary*

a picture of his dog"), where differences in role modifiers like ("`Topic`" and "`Creator`") for "picture" result in a SMATCH score of 69.23. The metric's penalty for these isolated structural variations, without accounting for the underlying semantic alignment, highlights its tendency to overlook contextually equivalent expressions when modifiers are altered or substituted. This penalization is further illustrated in Example 3, ("*Today I have a pain in my neck*"), where SMATCH deducts points based on minor discrepancies in verb sense, yielding a score of 60.00 despite the overall message being well-preserved across both DRS.

In Example 4, ("*Thirteen people were arrested*"), SMATCH once again assigns a score of zero, this time due to an inconsistency in the numerical value between gold (13) and predicted (30) DRS. This significant deduction overlooks that the core event—people being arrested—is accurately conveyed. Example 5, ("*I have a lot of money*"), further emphasizes SMATCH's limitations, where minor numerical and role discrepancies lead to a score of 57.89, despite the intended meaning being largely retained. These examples collectively underscore that SMATCH's sensitivity to structural changes can cause unfairly low scores even when semantic content is mostly preserved.

Given these limitations, there is a pressing need for more comprehensive evaluation methods that extend beyond structural similarity, incorporating measures of semantic accuracy and linguistic diversity. Our PARS-GEN approach addresses this gap by combining structural metrics with a generation-based evaluation. By converting parsed DRS outputs back into natural language and comparing them with reference texts, PARS-GEN can assess both the structural and semantic quality of parsing outputs, capturing the system's effectiveness in conveying intended meaning rather than merely replicating reference structures.

In practice, PARS-GEN can identify when a model has generated a semantically valid DRS despite structural differences. For instance, in cases like "Liz Mohn" and "I woke up at eleven" where SMATCH assigns zero scores, PARS-GEN reveals the semantic equivalence through text generation and subsequent evaluation using metrics such as chrF, METEOR, and BERTScore. This evaluation measure enables the quantification of parsing quality dimensions that structural metrics overlook, particularly in cases where syntactic variation does not impact semantic meaning.

A significant advantage of PARS-GEN lies in its ability to make DRS evaluation more accessible and interpretable. Traditional DRS evaluation requires deep expertise in understanding complex logical structures, making it challenging to identify whether parsing errors stem from structural misalignment or genuine semantic differences. For instance, in the "Nancy is very cute, isn't she?" example, determining why SMATCH assigns a score of 38.46 requires carefully analyzing the intricate interplay of negation operators, attribute relationships, and temporal anchoring in both the parsed (PARS-DRS) and gold DRS. Similarly, for "Is this appropriate?", one must understand how predicative adjectives versus nominal concepts are represented in DRS to comprehend why different yet semantically equivalent structures receive a SMATCH score of 64.00. PARS-GEN simplifies this analysis by converting DRS back into natural language text—equivalent textual representation of the DRS, allowing evaluators to directly observe whether semantic equivalence is preserved. When the parsed DRS for "I woke up at eleven" generates text that conveys the same temporal and eventive meaning as the original, it becomes immediately apparent that the SMATCH score of 00.00 is overly punitive, despite variations in

predicate sense tagging (wake_up.v.01 versus wake_up.v.02). This transformation from logical forms to text makes the evaluation process more transparent and enables even those without extensive DRS expertise to identify parsing issues and assess semantic accuracy.

Our findings indicate that the PARS-GEN approach provides a more nuanced evaluation that accurately reflects a system's ability to parse meaning from text. It captures critical linguistic phenomena that structural metrics like SMATCH often miss, including subtle variations in verb senses, named entities, and temporal expressions. The examples analyzed in Table 5.1 demonstrate how PARS-GEN can reveal the true semantic capabilities of parsing systems, particularly in cases where structural metrics fail to acknowledge valid variations in meaning representation.

Table 5.3: Semantic overlap-based evaluation measures: highlighting limitations of automatic evaluation metrics for text generation. Note: B_Scr. = BERTScore.

Ex. No.	Gold DRS	GEN (Text)	Gold Text	GEN				
				BLEU	METEOR	COMET	chrF	B_Scr.
1	event.v.01 Participant +1 room.n.01	The room.	A room.	55.03	62.50	84.85	54.69	99.88
2	person.n.01 EQU speaker NEGATION <1 time.n.08 TSU now return.v.01 Theme -2 Time -1	Let's not return again.	I'll never return.	12.70	19.61	77.84	27.25	92.23
3	person.n.01 Sub speaker buy.v.01 Agent -1 Time +1 Theme +2 time.n.08 EQU now cd.n.04	We buy some CDs.	We buy CDs.	30.21	91.46	93.88	56.82	96.73
4	NEGATION <1 time.n.08 EQU now entity.n.01 cute.a.01 Time -2 AttributeOf -1	Nothing is that cute.	Ain't that cute?	21.36	37.50	64.08	46.01	88.40
5	entity.n.01 time.n.08 TSU now cost.v.01 Theme -2 Time -1 Value +3 quantity.n.01 BOT +1 quantity.n.01 EQU 100 peso.n.03 Quantity -2	This will cost at least 100 pesos.	It'll cost at least a hundred pesos.	25.85	54.34	94.06	51.89	98.44

5.2.2 Limitations of Traditional Text Generation Metrics

Metrics such as BLEU and ROUGE, which rely on n-gram overlaps, fall short of capturing deeper semantic alignment beyond lexical matches. This limitation is strikingly evident in our analysis of DRS-to-text generation examples. For instance, the text "The room" generated from the DRS receives a BLEU score of 55.03 when compared to the gold text "A room," despite being semantically equivalent renderings of the underlying concept `room.n.01`—see Example 1 in Table 5.3. Similarly, for the expression of quantity in "It'll cost at least a hundred pesos" versus "This will cost at least 100 pesos", BLEU assigns a low score of 25.85, penalizing valid variations in number expression and determiner choice that preserves the core meaning (refer to Example 5 in Table 5.3). SMATCH in these cases, indicates full alignment with a score of 1, demonstrating the semantic fidelity gap in overlap-based measures.

Enhanced metrics such as METEOR and chrF incorporate linguistic features like stemming, synonym matching, and character-level analysis to address some limitations.

Our analysis reveals varying degrees of success: for "We buy CDs" versus "We buy some CDs," METEOR shows better performance (91.46) compared to BLEU (30.21), demonstrating its ability to handle minor determiner variations—Example 3 in Table 5.3. However, when evaluating "Let's not return again" against "I'll never return," METEOR score drops to 19.61, struggling with different expressions of negation despite semantic equivalence (see Example 2 in Table 5.3). This highlights METEOR's limitations in handling diverse ways of expressing the same semantic content. Character-based chrF, more adept at handling morphological differences, is valuable for languages with complex morphology but may over-penalize minor variations without significant meaning changes, posing challenges for languages where slight morphological shifts impact semantics.

Recent neural-based metrics like COMET and BERTScore aim to measure semantic similarity by leveraging pre-trained language models. While these metrics show promise in capturing semantic relationships, our examples reveal interesting patterns. Consider "Ain't that cute?" versus "Nothing is that cute," where BERTScore (88.40) and COMET (64.08) show higher correlation with semantic similarity compared to traditional metrics (BLEU: 21.36, METEOR: 37.50)—see Example 4 in Table 5.3. However, these scores still do not fully capture the semantic equivalence of these different syntactic constructions expressing the same sentiment. For example, BERTScore may prioritize fluency over semantic accuracy, assigning high scores to fluent but semantically incorrect generations if they align with patterns learned during training. Additionally, these neural metrics are computationally expensive, limiting their practicality for large-scale evaluations or real-time assessments. Finally, the opaque nature of neural models complicates error analysis—when a neural metric assigns a low score, it is often difficult to determine why, making it challenging to trace the issue back to specific aspects of the generated text that require improvement.

The limitations of these metrics become more pronounced when evaluating text generation across different languages. Many metrics rely on language-specific resources, creating a disparity in evaluation reliability across languages. Our analysis of examples spanning different linguistic phenomena—from simple entity descriptions to complex quantified expressions—reveals how current metrics struggle to account for cross-linguistic variations in expressing the same semantic content. Moreover, languages vary widely in how they express semantic content. Structural differences, such as word order in English versus morphological markers in Urdu, can lead to valid but different surface forms for the same underlying meaning. Current metrics struggle to account for these cross-linguistic variations, often penalizing semantically equivalent expressions simply because they differ lexically or structurally from the reference text. Cultural and pragmatic differences further complicate evaluation, as the same semantic content might be naturally expressed differently depending on the cultural context, and existing metrics are ill-equipped to handle such variations.

For **Urdu**, BLEU assigns a score of 16.67 to the generated translation ("*I am not ashamed yet*") compared to the gold reference ("*I'm not shy yet*") (Example 1, Table 5.4). While the generated text conveys the same core meaning, the BLEU score is low due to slight lexical variations in the choice of words like "ashamed" vs. "shy." This highlights BLEU's emphasis on lexical overlap over capturing the overall meaning of the sentence.

Similarly, for the translation (“*I bought two dozen pencils*”) compared to (“*I bought* 24 *pencils*”), both sentences convey the same meaning but are penalized due to different representations i.e., “two dozen” vs. “24.” This exemplifies the metric’s failure to acknowledge acceptable paraphrases or equivalent expressions in the target language, further underscoring its limitations in multilingual contexts. In both cases, SMATCH indicates complete semantic alignment with a score of 1.0, highlighting the gap in traditional metrics’ sensitivity to semantic fidelity.

METEOR, which improves on BLEU by considering synonym matching and stemming, does provide higher scores for the same example (43.31 vs. BLEU’s 49.12), but it is not immune to limitations. METEOR still struggles with capturing fine-grained semantic differences, as seen in Example 5 in Table 5.4, where the score of 37.04 fails to distinguish between (“*He was born in* 198*X*”) and (“*He was born in the eighties*”). Despite both sentences being semantically similar, METEOR’s score is lower because it does not consider the subtleties of temporal expressions in Urdu and fails to fully match the corresponding time entities. The chrF score (which focuses on character-level n-gram overlap) in this context, with scores ranging from 21.95 (Example 1) to 39.79 (Example 2), similarly fails to capture the underlying semantic similarity. While chrF can be somewhat more effective for languages with complex morphology, such as Urdu, it still penalizes minor differences in word structure and morphology, even when the generated text accurately conveys the intended meaning. In Example 1, “ashamed” vs. “shy” show small morphological differences that affect chrF’s performance, despite the generated text being semantically correct.

Table 5.4: Semantic overlap-based evaluation measures: highlighting limitations of automatic evaluation metrics for Urdu text generation. Note: B_Scr. = BERTScore.

Ex. No.	Gold DRS	Gold Text	GEN Text	GEN Scores				
				BLEU	METEOR	ROUGE	chrF	B_Scr.
1	person.n.01 EQU speaker ashamed.a.01 Experiencer -1 Time +1 NEGATION <1 time.n.08 EQU now	میں شرمندہ نہیں ہوں۔ (“I am not ashamed.”)	میں ابھی تک شرمندہ نہیں ہوں۔ (“I’m not shy yet.”)	16.67	11.90	19.99	21.95	78.96
2	person.n.01 EQU speaker quantity.n.01 EQU 24 pencil.n.01 Quantity -1 buy.v.01 Agent -3 Theme -1 Time +1 time.n.08 TPR now	میں نے دو درجن پنسلیں خریدیں۔ (“I bought two dozen pencils.”)	میں نے 24 پنسلیں خریدیں۔ (“I bought 24 pencils.”)	49.12	43.31	54.54	39.79	92.09
3	NEGATION <1 person.n.01 NEGATION <1 leave.v.01 Theme -1 Time +1 time.n.08 TPR now	سب چلے گئے۔ (“Everyone left.”)	اب سب چلے گئے ہیں۔ (“All have now left.”)	50.00	32.25	57.14	29.38	88.74
4	male.n.02 Name “جان” female.n.02 Name “مریم” money.n.01 give.v.03 Agent -3 Recipient -2 Theme -1 Time +1 time.n.08 TPR now	جان نے مریم کو کچھ پیسے دیے۔ (“John gave Mary some money.”)	جان نے رقم مریم کو دی۔ (“John gave the money to Mary.”)	56.43	57.52	61.54	48.84	89.99
5	male.n.02 time.n.08 YearOfCentury ’198X’ bear.v.02 Patient -2 Time -1 Time +1 time.n.08 TPR now	وہ اسی کی دہائی میں پیدا ہوئے۔ (“He was born in the eighties.”)	وہ اسّی میں پیدا ہوئے۔ (“He was born in 198X.”)	42.32	37.04	44.15	34.12	79.26

BERTScore, which attempts to measure semantic similarity using pre-trained language models, is better suited for capturing the deeper semantic relationships between words. However, even this metric struggles when dealing with syntactic and morphological variations in Urdu. For instance, in Example 3, (“*Everyone left*”) and (“*All have now left*”) exhibit a difference in tense and aspect, yet the meaning remains intact. BERTScore performs better here with scores of 88.74, but still faces challenges when evaluating mi-

nor syntactic differences that do not affect the overall meaning.

These examples clearly demonstrate the need for evaluation methods that go beyond surface-level comparisons. In DRS-to-text generation, where preserving semantic fidelity is crucial, traditional metrics often fail to capture the true quality of the generated text. For instance, the variation in expressing negation ("Let's not return again" vs. "I'll never return") receives low scores across traditional metrics (BLEU: 12.70, chrF: 27.25) despite maintaining semantic equivalence.

The GEN-PARS approach addresses these limitations by evaluating generated text in a semantically grounded manner. Rather than relying solely on surface similarities, it assesses whether the generated text preserves the underlying semantic content encoded in the original DRS. Notably, for all examples in our analysis where traditional metrics show significant variations in scores (e.g., BLEU ranging from 12.70 to 55.03), parsing the generated texts back to DRS representations yields perfect SMATCH scores of 1.00, confirming their semantic equivalence despite surface differences. This demonstrates GEN-PARS's ability to recognize valid semantic preservation even when traditional generation metrics suggest otherwise.

One of the most significant advantages of GEN-PARS is its ability to handle linguistic variations effectively. For instance, in cases like "We buy CDs" versus "We buy some CDs," where traditional metrics show inconsistent scores (BLEU: 30.21, METEOR: 91.46, chrF: 56.82), GEN-PARS confirms their semantic equivalence through perfect SMATCH scores. Similarly, even for more divergent surface realizations like "Let's not return again" versus "I'll never return," which receive very low traditional metric scores (BLEU: 12.70, METEOR: 19.61), GEN-PARS validates their semantic equivalence through DRS comparison. This pattern holds true across all our analyzed examples, demonstrating GEN-PARS's robustness in recognizing meaning-preserving variations regardless of surface form differences.

The GEN-PARS approach is particularly valuable for multilingual evaluation tasks. Whether evaluating simple entity descriptions like "The room" and "A room" or complex constructions involving negation and quantification, GEN-PARS focuses on semantic preservation rather than surface form. This makes it especially suitable for assessing generation quality across languages with different structural properties. Moreover, GEN-PARS provides interpretable feedback that can directly inform system improvement. By analyzing the differences between the DRS of the generated text and the original DRS, we can identify specific areas where the system fails to capture the intended meaning. This level of interpretability is crucial for improving DRS-to-text generation systems, particularly in cases where traditional metrics might miss semantic discrepancies or overly penalize valid variations.

5.3 Methods and Results

Our study introduces novel perspectives on evaluating DRS parsing and text generation systems. We present two evaluation methodologies: (1) evaluating parsing through generation capabilities (PARS-GEN), and (2) assessing generation through semantic parsing (GEN-PARS). Unlike traditional approaches that rely solely on structural metrics or surface-level text similarities, our work presents evaluation methodologies that provide

a comprehensive assessment of both structural and linguistic aspects of semantic processing systems.

For our experiments, we utilized state-of-the-art byT5-based models for both generation (DRS-to-Text) and parsing (Text-to-DRS), fine-tuned on a fully augmented PMB dataset. We evaluated our approaches (PARS/PARS-GEN and GEN/GEN-PARS) across three typologically distinct languages—English (EN), Italian (IT), and Urdu (UR)—using standard test sets. This cross-linguistic evaluation design allows us to assess how our evaluation methods perform across languages with different syntactic structures and semantic expression patterns.

Our cross-task evaluation approaches demonstrate varying performance across languages. For English, the PARS approach achieved a high SMATCH F1-score of 93.56, while the corresponding PARS-GEN evaluation revealed strong linguistic quality through multiple generation metrics (BLEU: 70.13, METEOR: 86.95, COMET: 95.34, chrF: 84.14, and BERTScore: 98.41). The GEN approach showed similarly impressive results (BLEU: 71.01, METEOR: 87.67, COMET: 95.81, chrF: 84.97, and BERTScore: 98.54), with GEN-PARS yielding a strong SMATCH score of 93.63. These results demonstrate robust semantic preservation across both parsing and generation tasks for English, indicating that structural accuracy strongly correlates with linguistic quality in this language.

Table 5.5: Experimental results of PARS and PARS-GEN on standard test sets for English, Italian, and Urdu. Underlined is the PARS-GEN metric that has the highest correlation with PARS. Note: S-F1 = SMATCH F1-Score; B_Scr. = BERTScore.

Language	**PARS**	**PARS-GEN**					
Type	**S-F1**	**BLEU**	**METEOR**	**COMET**	**chrF**	**B_Scr.**	**ROUGE**
EN	<u>93.56</u>	70.13	86.95	<u>95.34</u>	84.14	98.41	–
IT	<u>90.56</u>	<u>56.74</u>	73.32	90.15	71.69	92.80	–
UR	<u>79.77</u>	45.48	41.39	–	40.57	<u>85.36</u>	49.55

The evaluation results for Italian and Urdu reveal interesting cross-linguistic patterns. Italian maintained strong performance with PARS achieving a 90.56 SMATCH score, while PARS-GEN showed good generation metrics (BLEU: 56.74, METEOR: 73.32). The GEN/GEN-PARS approach for Italian yielded comparable results (BLEU: 56.76, SMATCH: 89.88), suggesting consistent semantic preservation across both directions. Urdu, with its complex morphological structure, showed more variation: PARS achieved a 79.77 SMATCH score with lower generation metrics in PARS-GEN (BLEU: 45.48, METEOR: 41.39), while GEN/GEN-PARS demonstrated somewhat different parsing performance (SMATCH: 74.83). These results highlight how morphological complexity and syntactic differences can impact both structural and linguistic evaluation metrics.

The comprehensive evaluation through our cross-task approaches (PARS/PARS-GEN and GEN/GEN-PARS) reveals important insights into semantic parsing and generation capabilities across languages. English consistently demonstrates robust performance in both structural and linguistic metrics, suggesting strong semantic preservation capabilities. Italian shows good but slightly reduced performance, while Urdu exhibits more significant variations, particularly in generation tasks. These patterns indicate that

Table 5.6: Experimental results of GEN and GEN-PARS approaches on standard test sets for English, Italian, and Urdu. Underlined is the GEN metric that has the highest correlation with GEN-PARS evaluation. Note: S-F1 = SMATCH F1-Score; B_Scr. = BERTScore.

Language	GEN						GEN-PARS
Type	BLEU	METEOR	COMET	chrF	B_Scr.	ROUGE	S-F1
EN	71.01	87.67	95.81	84.97	98.54	–	93.63
IT	56.76	72.67	89.97	70.59	92.85	–	89.88
UR	53.31	53.07	–	51.49	88.33	59.40	74.83

while our evaluation methods effectively capture semantic relationships, their effectiveness varies with linguistic complexity, suggesting the need for language-specific considerations in evaluation frameworks, particularly for morphologically rich languages. English, Italian, and Urdu results about PARS/PARS-GEN and GEN/GEN-PARS evaluations are listed in Table 5.5 and Table 5.6.

5.4 Analysis and Discussion

To further emphasize the usefulness of the reversible evaluation approaches that take into account both semantic and structural information, we have analyzed examples present in Table 5.1 and Table 5.3 by performing the reverse evaluations i.e., PARS through PARS-GEN and GEN through GEN-PARS as explained in Section 5.4.1. Furthermore, we have performed a correlation analysis on the evaluation measures by taking into account Pearson, Spearman, and point-biserial correlation evaluations as listed in Section 5.4.2.

5.4.1 Reversible Evaluation Measures

While previous sections, Section 5.2.1 and Section 5.2.2, highlighted the limitations of traditional parsing and generation metrics individually, in this section we present the cases where our proposed evaluation approaches (PARS-GEN and GEN-PARS) provide complementary evidence of semantic and structural preservation. Through detailed case studies, we demonstrate how low scores in one type of evaluation (PARS or GEN) can be counter-verified by evaluating it in the reverse direction, revealing semantic equivalences that would have been missed.

Evaluating PARS with reversible PARS-GEN Metrics

In analyzing DRS with structural overlap metrics like SMATCH, certain limitations in capturing the full semantic equivalence between the gold standard and generated DRS are evident. Table 5.1 highlights this issue through examples where PARS (DRS) scores do not adequately reflect semantic alignment despite the intended meaning being correctly represented. These examples underscore a critical drawback of relying solely on structural metrics, as they may fail to capture essential meaning alignment between generated and gold structures.

To address these limitations, PARS-GEN (text generation from DRS) evaluations in Table 5.7 supplement structural assessments with semantic overlap metrics, including BLEU, METEOR, COMET, chrF, and BERTScore, which provide a finer-grained view of how well the generated text aligns with the gold text. For Example 1, PARS-GEN achieves a BERTScore of 91.12 and METEOR of 78.12, capturing the semantic fidelity of the phrase "Liz Mohn is Liz Mohn." Although SMATCH did not register structural similarity, the text-based evaluations in PARS-GEN reveal a strong overlap in meaning. Similarly, Example 2 achieves nearly perfect PARS-GEN scores across all metrics (BLEU: 1.00, METEOR: 99.76, COMET: 98.49, chrF: 1.00, BERTScore: 1.00), demonstrating that despite SMATCH's inability to capture semantic alignment, PARS-GEN accurately reflects the intended message that the speaker woke up at eleven.

Table 5.7: Semantic overlap-based evaluation measures: highlighting limitations of automatic evaluation metrics for text generation. Note: B_Scr. = BERTScore.

Ex. No.	PARS (DRS)	PARS (SMATCH)	PARS-GEN (Text)	Gold Text	PARS-GEN				
					BLEU	METEOR	COMET	chrF	B_Scr.
1	female.n.02 Name "Liz Mohn" time.n.08 EQU now female.n.02 Name "Liz Mohn" Female -2 Time -1	00.00	Liz Mohn is Liz Mohn.	Liz Mohn.	16.23	78.12	78.89	65.98	91.12
2	person.n.01 EQU speaker wake_up.v.01 Agent -1 Time +1 Time +2 Time +3 time.n.08 TPR now time.n.08 ClockTime 11:00	00.00	I woke up at eleven.	I woke up at eleven.	1.00	99.76	98.49	1.00	1.00
3	female.n.02 Name "Nancy" time.n.08 EQU now very.r.01 cute.a.01 AttributeOf -3 Time -2 Degree -1	38.46	Nancy is very cute.	Nancy is very cute, isn't she?	36.70	36.70	87.63	57.70	93.49
4	event.v.01 Participant +1 plant_tree.n.01	44.44	Plant trees.	Plant trees!	55.03	62.50	92.66	80.38	96.18
5	be.v.01 Time +1 Theme +2 Co-Theme +3 time.n.08 EQU now entity.n.01 appropriate.n.01	64.00	Is that appropriate?	Is this appropriate?	35.35	63.88	98.21	66.53	97.54

Furthermore, Example 3 in Table 5.1 highlights a nuanced challenge where SMATCH (38.46) underestimates the semantic alignment due to complex relational and sentiment-bearing expressions. Here, the DRS encodes the phrase "Nancy is very cute," yet this overlap is inadequately represented by the structural metric. In contrast, PARS-GEN scores in Table 5.7, with a BERTScore of 93.49, provide a closer approximation of the intended meaning, thereby validating the DRS from a semantic standpoint. Similarly, Example 5 also demonstrates this phenomenon, where a SMATCH score of 64.00 misses subtle lexical differences in phrases like "Is that appropriate?" vs. "Is this appropriate?" PARS-GEN BLEU (35.35) and BERTScore (97.54) confirm semantic equivalence, which structural evaluation alone failed to capture.

Analyzing **Urdu** examples, Table 5.2 highlights this issue through examples where PARS (DRS) scores do not adequately reflect semantic alignment despite the intended meaning being correctly represented. These examples underscore a critical drawback of relying solely on structural metrics, as they may fail to capture essential meaning alignment between generated and gold structures.

To address these limitations, PARS-GEN (text generation from DRS) evaluations in

Table 5.8 supplement structural assessments with semantic overlap metrics, including BLEU, METEOR, ROUGE, chrF, and BERTScore, which provide a finer-grained view of how well the generated text aligns with the gold text. In Example 1, PARS-GEN achieves a BERTScore of 97.30 and METEOR of 69.14, capturing the semantic fidelity of the phrase (*"Tom bought a new pickup"*). Although SMATCH did not register structural similarity, the text-based evaluations in PARS-GEN reveal a strong overlap in meaning. Similarly, Example 2 achieves a perfect PARS-GEN scores across all metrics (BLEU: 1.00, METEOR: 99.93, ROUGE: 99.99, chrF: 1.00, BERTScore: 1.00), demonstrating that, despite SMATCH's inability to capture semantic alignment, PARS-GEN accurately reflects the intended message that the `Owner` showed `Recipient` a picture of dog.

Furthermore, Example 3 in Table 5.2 highlights a nuanced challenge where SMATCH (60.00) underestimates the semantic alignment due to complex relational and sentiment-bearing expressions. Here, the DRS encodes the phrase (*"My neck still hurts"*) yet this overlap is inadequately represented by the structural metric. In contrast, PARS-GEN scores in Table 5.8, with a BERTScore of 87.11, provide a closer approximation of the intended meaning, thereby validating the DRS from a semantic standpoint. Similarly, Example 5 also demonstrates this phenomenon, where a SMATCH score of 57.89 misses subtle lexical differences in phrases like (*"I have a lot of money"*), PARS-GEN BLEU (83.33) and BERTScore (97.63) confirm semantic equivalence, which structural evaluation alone failed to capture.

Table 5.8: Evaluating Urdu PARS through PARS-GEN by taking examples from Table 5.2. Note: B_Scr. = BERTScore.

Ex. No.	PARS DRS	PARS (SMATCH)	PARS GEN Text	Gold Text	GEN Scores				
					BLEU	METEOR	ROUGE	chrF	B_Scr.
1	male.n.02 Name "ٹام" new.a.01 AttributeOf +1 pick_up.n.01 buy.v.01 Agent -4 Beneficiary -3 Theme -1 Time +1 time.n.08 TPR now	00.00	ٹام نے ایک نیا پک اپ خریدا. ("Tom bought a new pickup.")	ٹام نے ایک نیا پک اپ خریدا. ("Tom bought a new pickup.")	71.43	69.14	71.43	64.14	97.30
2	male.n.02 Name "ٹام" female.n.02 Name "مریم" female.n.02 ANA -1 dog.n.01 Owner -1 photo.n.01 Creator -1 show.v.01 Agent -4 Recipient -3 Recipient -1 Time +1 time.n.08 TPR now	69.23	ٹام مریم کو اپنے کتے کی تصویر دکھاتا ہے. ("Tom shows Mary a picture of his dog")	ٹام مریم کو اپنے کتے کی تصویر دکھاتا ہے. ("Tom shows Mary a picture of his dog.")	1.00	99.93	99.99	1.00	1.00
3	person.n.01 EQU speaker neck.n.01 hurt.v.01 Patient -2 Patient -1 Time +1 time.n.08 EQU now	60.00	میری گردن میں اب بھی درد ہے. ("My neck still hurts.")	آج میری گردن میں درد ہے. ("Today I have a pain in my neck.")	57.14	61.47	61.54	48.07	87.11
4	quantity.n.01 EQU 30 person.n.01 Quantity -1 arrest.v.01 Patient -3 Time +1 time.n.08 TPR now	00.00	تیس افراد کو گرفتار کر لیا گیا. ("Thirty people were arrested.")	تیرہ افراد کو گرفتار کر لیا گیا. ("Thirteen people were arrested.")	56.43	54.35	61.54	61.72	94.55
5	person.n.01 EQU speaker have.v.01 Pivot -1 Theme +2 Time +3 quantity.n.01 EQU +1 quantity.n.01 EQU +1 money.n.01 Quantity + time.n.08 EQU now	57.89	میرے پاس بہت پیسہ ہے. ("I have a lot of money.")	میرے پاس بہت پیسہ ہے. ("I have a lot of money.")	83.33	80.66	83.33	76.08	97.63

This analysis reveals that PARS-GEN complements structural metrics by providing a more robust measure of semantic fidelity in text generation tasks. By using both PARS and PARS-GEN, we gain a comprehensive understanding of meaning overlap, particularly in cases where linguistic nuances or variations may obscure the structural alignment but are nonetheless captured through text-based evaluations. Together, PARS and PARS-GEN offer a dual approach that effectively bridges the gap between structural and semantic overlap, enhancing the accuracy and reliability of DRS evaluation.

Evaluating GEN with reversible GEN-PARS Metric

The evaluation of generated text against gold DRS (after performing GEN-PAR) using semantic overlap metrics reveals critical insights into the limitations of traditional automatic metrics for text generation. Table 5.3 outlines these issues, showcasing several examples where semantic alignment is assessed through automatic word-overlap-based measures e.g., BLEU, METEOR, COMET, chrF, and BERTScore. This discrepancy suggests that, while BLEU and METEOR focus on n-gram matching, they may not adequately capture the semantic richness and structural sequences represented in the DRS.

Transitioning to Table 5.9, which focuses on structural overlap metrics, we observe the implementation of GEN-PARS, which assesses the generated text against the original DRS. Notably, all examples (1-5) yield a perfect SMATCH score of 1.00, signifying that the generated structures align perfectly with the gold DRS. For example, in Example 1, the transition from "The room" in GEN to the corresponding GEN-PARS representation maintains the event structure intact, reinforcing the idea that the generated text retains all necessary elements for a correct DRS encoding.

Table 5.9: Structural overlap-based evaluation measures: highlighting limitations of SMATCH.

Ex. No.	**GEN (Text)**	**GEN-PARS (DRS)**	**Gold DRS**	**GEN-PARS (SMATCH)**
1	The room.	event.v.01 Participant +1 room.n.01	event.v.01 Participant +1 room.n.01	1.00
2	Let's not return again.	person.n.01 EQU speaker NEGATION <1 time.n.08 TSU now return.v.01 Theme -2 Time -1	person.n.01 EQU speaker NEGATION <1 time.n.08 TSU now return.v.01 Theme -2 Time -1	1.00
3	We buy some CDs.	person.n.01 Sub speaker buy.v.01 Agent -1 Time +1 Theme +2 time.n.08 EQU now cd.n.04	person.n.01 Sub speaker buy.v.01 Agent -1 Time +1 Theme +2 time.n.08 EQU now cd.n.04	1.00
4	Nothing is that cute.	NEGATION <1 entity.n.01 time.n.08 EQU now cute.a.01 AttributeOf -2 Time -1	NEGATION <1 time.n.08 EQU now entity.n.01 cute.a.01 Time -2 AttributeOf -1	1.00
5	This will cost at least 100 pesos.	entity.n.01 time.n.08 TSU now cost.v.01 Theme -2 Time -1 Value +3 quantity.n.01 BOT +1 quantity.n.01 EQU 100 peso.n.03 Quantity -2	entity.n.01 time.n.08 TSU now cost.v.01 Theme -2 Time -1 Value +3 quantity.n.01 BOT +1 quantity.n.01 EQU 100 peso.n.03 Quantity -2	1.00

Furthermore, Example 3 and Example 4 reveal similar patterns. Both examples demonstrate that the generated text aligns seamlessly with the DRS structure, as evidenced by the SMATCH scores of 1.00. The transformation from "We buy some CDs" to its DRS representation encapsulates the essential semantic components, reinforcing the effectiveness of GEN-PARS in maintaining structural integrity while providing a high-quality semantic output.

For **Urdu** Table 5.4 outlines these issues, showcasing several examples where semantic alignment is assessed through automatic word-overlap-based measures, e.g., BLEU, METEOR, ROUGE, chrF, and BERTScore. This discrepancy suggests that, traditional evaluation metrics for Urdu text focus on n-gram matching, they may not adequately capture the semantic richness and structural sequences represented in the DRS.

Transitioning to Table 5.10, which focuses on structural overlap metrics, we observe the implementation of GEN-PARS, which assesses the generated text against the original DRS. Notably, all examples (1-5) yield a perfect SMATCH score of 1.00, signifying that the generated structures align perfectly with the gold DRS. For instance, in Example 1, the transition from (*"I'm not shy yet"*) in GEN to the corresponding GEN-PARS representation maintains the event structure intact, reinforcing the idea that the generated text retains all necessary elements for a correct DRS encoding.

Furthermore, Example 3 and Example 4 reveal similar patterns. Both examples demonstrate that the generated text aligns seamlessly with the DRS structure, as evidenced by the SMATCH scores of 1.00. The transformation from (*"All have now left"*) and (*"John gave the money to Mary"*) to their DRS representations encapsulates the essential semantic components, reinforcing the effectiveness of GEN-PARS in maintaining structural integrity while providing a high-quality semantic output.

Table 5.10: Evaluating Urdu GEN through GEN-PARS by taking examples from Table 5.4. Note: GPARS = GEN-PARS

Ex. No.	GEN Text	GEN-PARS (DRS)	Gold DRS	GPARS (SMATCH)
1	میں ابھی تک شرمندہ نہیں ہوں. ("I'm not shy yet.")	person.n.01 EQU speaker ashamed.a.01 Experiencer -1 Time +1 NEGATION <1 time.n.08 EQU now	person.n.01 EQU speaker ashamed.a.01 Experiencer -1 Time +1 NEGATION <1 time.n.08 EQU now	1.00
2	میں نے 24 پنسلیں خریدیں. ("I bought 24 pencils.")	person.n.01 EQU speaker quantity.n.01 EQU 24 pencil.n.01 Quantity -1 buy.v.01 Agent -3 Theme -1 Time +1 time.n.08 TPR now	person.n.01 EQU speaker quantity.n.01 EQU 24 pencil.n.01 Quantity -1 buy.v.01 Agent -3 Theme -1 Time +1 time.n.08 TPR now	1.00
3	اب سب چلے گئے ہیں. ("All have now left.")	NEGATION <1 person.n.01 NEGATION <1 leave.v.01 Theme -1 Time +1 time.n.08 TPR now	NEGATION <1 person.n.01 NEGATION <1 leave.v.01 Theme -1 Time +1 time.n.08 TPR now	1.00
4	جان نے رقم مریم کو دی. ("John gave the money to Mary.")	male.n.02 Name "جان" female.n.02 Name "مریم" money.n.01 give.v.03 Agent -3 Recipient -2 Theme -1 Time +1 time.n.08 TPR now	male.n.02 Name "جان" female.n.02 Name "مریم" money.n.01 give.v.03 Agent -3 Recipient -2 Theme -1 Time +1 time.n.08 TPR now	1.00
5	وہ امّی میں پیدا ہوئے. ("He was born in 198X.")	male.n.02 time.n.08 YearOfCentury 198X bear.v.02 Patient -2 Time -1 Time +1 time.n.08 TPR now	male.n.02 time.n.08 YearOfCentury '198X' bear.v.02 Patient -2 Time -1 Time +1 time.n.08 TPR now	1.00

This analysis elucidates the necessity of integrating both semantic (GEN) and structural (GEN-PARS) evaluations in understanding the quality of generated texts. While GEN metrics highlight the challenges posed by conventional evaluations in capturing semantic nuances, GEN-PARS effectively illustrates how generated structures can align with DRS, thus ensuring that the meaning is preserved. By leveraging both sets of metrics, we can obtain a more nuanced view of the strengths and limitations of text-generation processes, fostering improvements in model training and evaluation methodologies.

5.4.2 Correlation Analysis

While evaluating DRS-based systems, robust analysis is crucial for understanding both quantitative performance measures and qualitative aspects, such as the system's alignment with human expectations and the preservation of underlying semantic content. Traditional metrics provide a foundation, but a deeper exploration through correlation analysis can offer more nuanced insights into how closely automatic evaluations reflect human judgments and whether these metrics capture meaningful variations in generated outputs. By analyzing correlations between automated measures like PARS/PARS-GEN and GEN/GEN-PARS, and comparing them with human evaluations, we aim to assess the reliability of these metrics across different evaluation approaches. Correlation analysis helps reveal whether observed patterns in generated text align with intended semantic accuracy, facilitating a holistic view of system performance across both machine-driven and human-centered evaluations.

To better understand the relationship in-between automated evaluation metrics and also with human judgment, we conducted a correlation analysis using both Pearson and point-biserial correlations. Each of these methods provides distinct insights into the performance alignment across different evaluation dimensions.

Pearson correlation was used to assess the linear relationship between automated evaluation metrics, specifically PARS/PARS-GEN and GEN/GEN-PARS scores. By quantifying the degree to which these metrics correlate, Pearson correlation enables us to determine if changes in one metric (e.g., the accuracy of parse-generation cycles) are mirrored by changes in another (e.g., the quality of the generated text). A high positive correlation would indicate consistency between these metrics, implying that they capture similar aspects of semantic fidelity and structural coherence within the generated text.

For human evaluation, we used point-biserial correlation to examine the relationship between the binary nature of human judgments (e.g., ROSE) and the continuous scores provided by automated metrics. This analysis provides insight into whether high scores from automated metrics are predictive of human-perceived accuracy, bridging the gap between quantitative measures and qualitative evaluations. A significant point-biserial correlation suggests that automated metrics may reliably reflect the human-perceived quality of the generated text, reinforcing their suitability for evaluating DRS-to-text systems.

Together, these correlation measures allow us to assess the reliability of both automated and human evaluations, facilitating a comprehensive understanding of model performance across different evaluation dimensions.

PARS/PARS-GEN Correlation

To validate our PARS/PARS-GEN evaluation approach, we conducted a comprehensive correlation analysis across three languages, examining the relationship between parsing performance (SMATCH scores) and various generation metrics. The analysis employed Pearson correlation coefficients to measure the strength and direction of relationships, along with p-values to determine statistical significance.

For **English**, the Pearson correlation analysis revealed interesting patterns across dif-

ferent evaluation metrics. The strongest Pearson correlation was observed between parsing performance (PARS) and COMET scores (r = 0.3776, p < 1.12e-39) i.e., a PARS-GEN evaluation measure, indicating a moderate positive relationship with very high statistical significance. BERTScore also showed moderate correlation (r = 0.3316, p < 1.85e-30), while other metrics such as BLEU (r = 0.1867), METEOR (r = 0.1980), and chrF (r = 0.2137) demonstrated weaker but statistically significant correlations (all p-values < 1e-10). These results suggest that neural-based metrics (COMET and BERTScore) capture parsing quality more effectively than traditional surface-level metrics.

In **Italian**, we observed more uniform correlation patterns across all metrics. The strongest correlation was found with BLEU scores (r = 0.2928, p < 1.95e-12), closely followed by BERTScore (r = 0.2888, p < 3.97e-12). Other metrics showed similar correlation strengths: METEOR (r = 0.2812), COMET (r = 0.2746), and chrF (r = 0.2720), all with high statistical significance (p < 1e-10). The consistency in correlation values suggests that for Italian, different generation metrics provide comparable insights into parsing quality.

For **Urdu**, despite its morphological complexity, we observed significant correlations across all metrics. BERTScore showed the strongest correlation (r = 0.2832, p < 4.55e-18), while BLEU (r = 0.2318), ROUGE (r = 0.2023), chrF (r = 0.2042), and METEOR (r = 0.1949) demonstrated weaker but highly significant correlations. The stronger correlation with BERTScore suggests that contextual embeddings may better capture semantic relationships in morphologically rich languages.

Comparing across languages, several notable patterns emerge. First, correlations are consistently positive and statistically significant across all languages and metrics, validating the fundamental premise of our PARS-GEN evaluation approach. Second, the strength of correlations varies by language, with English showing the strongest correlations, particularly for neural-based metrics. This variation might reflect the different challenges posed by each language's linguistic structure and the varying capabilities of different evaluation metrics to capture these nuances.

GEN/GEN-PARS Correlation

Following our analysis of PARS/PARS-GEN correlations, we examined the relationships between generation metrics and parsing performance in the reverse direction through our GEN/GEN-PARS approach. This bidirectional analysis provides insights into how well generation quality predicts parsing performance across languages.

For **English**, the GEN/GEN-PARS correlations showed generally weaker relationships compared to the PARS/PARS-GEN approach. COMET demonstrated the strongest correlation (r = 0.2146, p < 2.91e-13), while traditional metrics showed very weak correlations: BLEU (r = 0.1058, p = 0.0003) and METEOR (r = 0.1075, p = 0.0002). BERTScore and chrF showed slightly stronger correlations (r = 0.1894 and r = 0.1637 respectively) but remained in the weak range. These results suggest that neural-based metrics like COMET may be more reliable indicators of semantic preservation in the generation-to-parsing direction.

Italian exhibited more consistent correlation patterns across metrics in the GEN/GEN-PARS evaluation. METEOR showed the strongest correlation (r = 0.2053, p < 1.09e-6), followed closely by BLEU (r = 0.2001, p < 2.01e-6). COMET, chrF, and BERTScore

Table 5.11: Correlation (Pearson) Results for PARS and PARS/GEN across English, Italian, and Urdu. Underlined correlation values represent the highest (strongest) correlation of PARS with PARS-GEN evaluation measures.

Sr. No	PARS vs. PARS/GEN	English			
		Corr-val	P-val	Correlation	Significance
1	Pars vs BLEU	0.1867	2.42e-10	weak	statistically significant
2	Pars vs METEOR	0.1980	1.76e-11	weak	statistically significant
3	Pars vs COMET	0.3776	1.12e-39	moderate	very highly significant
4	Pars vs chrF	0.2137	3.63e-13	weak	statistically significant
5	Pars vs BERTScore	0.3316	1.85e-30	moderate	highly significant
		Italian			
		Corr-val	P-val	Correlation	Significance
1	Pars vs BLEU	0.2928	1.95e-12	weak	highly significant
2	Pars vs METEOR	0.2812	1.51e-11	weak	highly significant
3	Pars vs COMET	0.2746	4.58e-11	weak	highly significant
4	Pars vs chrF	0.2720	7.17e-11	weak	highly significant
5	Pars vs BERTScore	0.2888	3.97e-12	weak	highly significant
		Urdu			
		Corr-val	P-val	Correlation	Significance
1	Pars vs BLEU	0.2318	1.87e-12	weak	highly significant
2	Pars vs METEOR	0.1949	3.69e-9	weak	highly significant
3	Pars vs ROUGE	0.2023	9.12e-10	weak	highly significant
4	Pars vs chrF	0.2042	6.25e-10	weak	highly significant
5	Pars vs BERTScore	0.2832	4.55e-18	weak	highly significant

showed slightly weaker but still significant correlations (r = 0.1822, 0.1947, and 0.1674 respectively). The relatively uniform correlation strengths suggest that for Italian, different generation metrics provide similar predictive power for parsing quality.

Interestingly, **Urdu** showed the strongest correlations in the GEN/GEN-PARS evaluation among all three languages. BERTScore demonstrated the highest correlation (r = 0.4073, p < 2.75e-37), followed by BLEU (r = 0.3414, p < 5.36e-26). Other metrics including ROUGE (r = 0.3043), chrF (r = 0.2987), and METEOR (r = 0.2936) also showed significant correlations. These stronger correlations for Urdu are particularly noteworthy given its morphological complexity.

Comparing the correlations across both evaluation directions (PARS/PARS-GEN and GEN/GEN-PARS) reveals several key insights:

1. The PARS/PARS-GEN approach generally shows stronger correlations for English and Italian, suggesting that parsing quality might be a better predictor of generation performance than vice versa.

2. Urdu demonstrates an interesting pattern where GEN/GEN-PARS correlations are notably stronger than PARS/PARS-GEN correlations, particularly for BERTScore and BLEU.

3. Neural-based metrics (COMET and BERTScore) consistently show stronger correlations compared to traditional metrics across both approaches, indicating their superior capability in capturing semantic relationships.

4. All correlations remain positive and statistically significant across both approaches, validating the bidirectional relationship between parsing and generation quality.

Table 5.12: Correlation (Pearson) Results for GEN and GEN/PARS across English, Italian, and Urdu. Underlined correlation values represent the highest (strongest) correlation of GEN with GEN-PARS evaluation measures.

Sr. No	GEN vs. GEN/PARS	English			
		Corr-val	P-val	Correlation	Significance
1	BLEU vs Gen-Pars	0.1058	0.0003	very weak	statistically significant
2	METEOR vs Gen-Pars	0.1075	0.0002	very weak	statistically significant
3	COMET vs Gen-Pars	0.2146	2.91e-13	weak	highly significant
4	chrF vs Gen-Pars	0.1637	2.99e-08	weak	statistically significant
5	BERTScore vs Gen-Pars	0.1894	1.31e-10	weak	statistically significant
		Italian			
		Corr-val	P-val	Correlation	Significance
1	BLEU vs Gen-Pars	0.2001	2.01e-6	weak	highly significant
2	METEOR vs Gen-Pars	0.2053	1.09e-6	weak	highly significant
3	COMET vs Gen-Pars	0.1822	1.55e-5	weak	highly significant
4	chrF vs Gen-Pars	0.1947	3.79e-6	weak	highly significant
5	BERTScore vs Gen-Pars	0.1674	7.37e-5	weak	highly significant
		Urdu			
		Corr-val	P-val	Correlation	Significance
1	BLEU vs Gen-Pars	0.3414	5.36e-26	moderate	highly significant
2	METEOR vs Gen-Pars	0.2936	2.30e-19	weak	highly significant
3	ROUGE vs Gen-Pars	0.3043	9.82e-21	weak	highly significant
4	chrF vs Gen-Pars	0.2987	5.25e-20	weak	highly significant
5	BERTScore vs Gen-Pars	0.4073	2.75e-37	moderate	highly significant

Human Evaluation (ROSE) Correlation with GEN

To provide a more comprehensive validation of our evaluation approaches, we conducted a human evaluation using the ROSE framework for English language samples[1]. ROSE implements a binary evaluation scheme incorporating semantic, grammatical, and phenomena-based assessments, where a positive score (1) requires satisfaction of all three criteria. We analyzed the relationship between ROSE scores and automatic metrics using point-biserial correlation for binary-continuous relationships and Spearman correlation for ranked relationships.

The point-biserial correlation analysis revealed strong relationships between human judgment and automatic metrics. BERTScore demonstrated the strongest correlation with ROSE ($r = 0.5392$, $p < 2.21e\text{-}86$), indicating its superior ability to capture human-perceived quality. METEOR and COMET also showed substantial correlations ($r = 0.5153$ and $r = 0.4819$ respectively), while BLEU and chrF demonstrated moderate correlations ($r = 0.4239$ and $r = 0.4424$ respectively). The Spearman correlation analysis largely confirmed the patterns observed in the point-biserial analysis, with some interesting nuances. BERTScore maintained its position as the strongest correlate with

[1]We are working on the ROSE evaluation for Italian and Urdu.

human judgment (r = 0.51, p < 0.01), while COMET and METEOR showed comparable correlations (r = 0.47 and r = 0.46 respectively). BLEU and chrF demonstrated medium correlations (r = 0.42 and r = 0.44).

These human evaluation results provide several crucial insights:

1. Pre-trained model-based metrics (BERTScore) align most closely with human judgment, suggesting their effectiveness in capturing semantic and grammatical nuances that humans consider important.

2. The moderate to strong correlations of neural-based metrics (COMET) and sophisticated linguistic metrics (METEOR) with human judgment validate their use in automatic evaluation pipelines.

3. The strong agreement between point-biserial and Spearman correlations reinforces the reliability of these findings across different statistical approaches.

These findings have important implications for our evaluation methodology. While our GEN approach provides valuable surface-level insights, the stronger correlations between automatic metrics and human judgment suggest that a hybrid evaluation approach might be most effective. Such an approach would combine structural analysis with neural-based metrics that better align with human perception of text quality.

Table 5.13: Human Evaluation Results for ROSE: Point-Biserial and Spearman Correlations for English. Underlined correlation values represent the highest (strongest) correlation of ROSE with GEN evaluation measures.

Sr. No	**ROSE vs. GEN**	**Point-Biserial Correlation**			
		Corr-val	P-val	Correlation	Significance
1	ROSE vs BLEU	0.4239	1.36e-50	moderate	highly significant
2	ROSE vs METEOR	0.5153	8.18e-78	moderate to strong	strong statistical significance
3	ROSE vs COMET	0.4819	6.84e-67	moderate to strong	highly significant
4	ROSE vs chrF	0.4424	1.92e-55	moderate	highly significant
5	ROSE vs Bert_Scr	<u>0.5392</u>	2.21e-86	stronger	very highly significant
		Spearman Correlation			
		Corr-val	P-val	Correlation	Significance
1	ROSE vs BLEU	0.42	0.00	medium	significant
2	ROSE vs METEOR	0.46	0.00	medium	significant
3	ROSE vs COMET	0.47	0.00	medium	significant
4	ROSE vs chrF	0.44	0.00	medium	significant
5	ROSE vs Bert_Scr	<u>0.51</u>	0.00	high	significant

5.5 Chapter Conclusion

This research advances the evaluation of DRS-based semantic parsing and text generation systems by addressing key limitations of traditional evaluation metrics, which often fail to capture the full spectrum of structural and linguistic fidelity required for accurate assessment. Through the development of two novel, complementary evaluation methodologies—Parse-Generate (Pars-Gen) and Generate-Parse (Gen-Pars)—we provide

a bidirectional framework that integrates both structural and linguistic perspectives to more holistically evaluate DRS-based systems.

Pars-Gen enables a nuanced assessment of semantic parsing quality by generating natural language text from parsed DRSs, identifying linguistic phenomena that purely structural metrics, such as SMATCH, may overlook. Conversely, Gen-Pars transforms generated text back into DRS, offering a structural and semantic measure of text generation quality that goes beyond surface-level metrics, including BLEU, METEOR, and BERTScore. Applying these methodologies to the PMB dataset across English, Italian, and Urdu has revealed language-specific and universal patterns in semantic representation, shedding light on the interplay between structural accuracy and linguistic coherence in multilingual contexts.

Our findings underscore the need for integrated evaluation approaches in semantic processing, offering practical insights for system debugging and optimization. By highlighting critical error types and proposing balanced metric weighting strategies, this research provides notable recommendations for advancing DRS-based semantic parsing and text generation systems.

Limitations: The cross-task evaluations conducted for DRS parsing and generation offer a foundational approach to assessing the structural and linguistic quality of multilingual semantic processing comprehensively. However, the transformation process from DRS to text and text to DRS relies heavily on the capabilities of the underlying pre-trained language models. These models must demonstrate sufficient generalizability and robustness to achieve accurate and high-quality data transformations between DRS and text formats. Model biases or limitations in the pre-trained architecture may adversely impact performance, potentially resulting in evaluations that deviate from gold-standard outputs. This reliance on model quality underscores the need for continued refinement and bias mitigation in pre-trained models to ensure reliable and unbiased semantic transformation and evaluation.

Chapter 6

Exploiting Task Reversibility of DRS Parsing and Generation: Challenges and Insights from Multi-lingual Perspective

Semantic parsing and text generation exhibit reversible properties when utilizing Discourse Representation Structures (DRS). However, both processes—text-to-DRS parsing and DRS-to-text generation—are susceptible to errors. In this chapter, we exploit the reversible nature of DRS to explore both error propagation, which is commonly seen in pipeline methods, and the less frequently studied potential for error correction. We investigate two pipeline approaches: Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG), utilizing pre-trained language models where the output of one model becomes the input for the next. Using the Parallel Meaning Bank dataset, we evaluate these pipelines across English, Italian, and Urdu, revealing complex dynamics between error propagation/amplification and mitigation/correction. Our findings show that, while pipeline approaches often amplify errors, surprisingly they occasionally perform error correction, with varying impacts across languages. Through detailed analysis across multiple dimensions—including sentence length, structural complexity, sentence type, polarity, and voice—we identified specific patterns in error behavior. English demonstrates the most stability in pipeline processing, while Italian shows moderate variations, and Urdu exhibits the highest sensitivity. Further, our findings suggest that these pipeline methods support the development of more linguistically balanced datasets to enable a comprehensive assessment. This cross-linguistic investigation contributes to our understanding of error dynamics i.e., propagation or mitigation in semantic processing and highlights both the capabilities and limitations of reversible pipeline approaches across diverse languages.

Chapter adapted from

1. Muhammad Saad Amin, Luca Anselma, and Alessandro Mazzei. 2025. Exploiting Task Reversibility of DRS Parsing and Generation: Challenges and Insights from a Multi-lingual Perspective. In Proceedings of the First Workshop on Language Models for Low-Resource Languages, pages 268–286, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

6.1 Introduction

DRS has demonstrated its versatility through applications in various NLP tasks, including semantic parsing [van Noord et al., 2018, Noord, 2019, van Noord et al., 2019], and text generation [Wang et al., 2021b, Amin et al., 2022b, Liu et al., 2021, Amin et al., 2024]. A particularly intriguing property of these tasks is their reversible nature—the output of one can serve as the input of the other, creating a theoretical cycle of semantic processing. However, existing research has typically approached semantic parsing and generation as isolated tasks, focusing primarily on English. This approach necessitates building separate models for each task and language, a strategy significantly constrained by data availability and resource requirements.

While recent years have witnessed significant advances in NLP through pre-trained language models, semantic parsing and text generation have faced unique challenges in fully leveraging these advancements. The primary obstacle lies in the explicit representation of meaning, which is not inherently integrated into the training objectives of these models [Amin et al., 2024]. Consequently, despite recent progress, both DRS semantic parsing and text generation remain complex and error-prone tasks [Wang et al., 2023a]. Parsing errors can result in incorrect or incomplete meaning representations, while generation errors manifest as disfluent or semantically inconsistent text [Wang et al., 2021b]. Understanding and addressing these error patterns becomes crucial for improving the reliability of DRS-based systems. Conventional approaches to improving performance in these domains typically involve resource-intensive retraining of models on larger datasets or the implementation of more complex architectural designs.

In this work, we propose an approach leveraging the reversible nature of semantic parsing and text generation to investigate error propagation and mitigation across multiple languages without additional model training. Our method utilizes pre-trained language models in two pipeline setups: 1) Parse-Generate-Parse (PGP), where input text is parsed, used to generate text, and then parsed again; and 2) Generate-Parse-Generate (GPG), where a DRS is used to generate text, which is parsed and then used to regenerate text.

We conduct our evaluation using the PMB dataset, a comprehensive benchmark for DRS-based semantic processing [Abzianidze et al., 2017]. Our analysis spans three typologically diverse languages: English, Italian, and Urdu. This cross-linguistic approach provides unique insights into how different linguistic structures and complexities influence the behavior of our pipeline approaches, offering a more comprehensive understanding of error dynamics in semantic processing.

This chapter addresses several research questions aimed at exploring the impact of reversible semantic parsing and text generation pipelines on error dynamics across multiple languages. Specifically, it examines how the reversible nature of DRS in semantic parsing and text generation influences error propagation and mitigation across different languages. Additionally, the chapter investigates whether language models can effectively uncover and analyze error dynamics when applied in a structured pipeline approach without further model training. A key focus is to measure performance changes introduced by the reversible PGP and GPG pipelines compared to baseline models, assessing these shifts across linguistic contexts. Furthermore, the chapter seeks to identify which types of errors are more effectively mitigated or amplified by the PGP and GPG

pipelines in each language. Lastly, the research questions probe the capabilities and constraints of these reversible pipeline approaches within diverse linguistic frameworks, providing insights into their adaptability and limitations.

The key contributions of this chapter are threefold:

1. We propose a novel method for investigating error dynamics in DRS-based NLP tasks by exploiting the inherent reversibility of semantic parsing and generation. This approach provides a new perspective on error analysis without requiring additional training resources.

2. Through extensive experimentation, we demonstrate the varied effects of pipeline approaches across multiple languages, revealing how linguistic differences influence error patterns and correction possibilities.

3. We present a comprehensive cross-linguistic error analysis framework that examines various linguistic dimensions, providing insights into both the capabilities and limitations of reversible pipeline approaches in different linguistic contexts.

Through these contributions, we explore the preliminary aspects of task reversibility within DRS parsing and generation. By examining pipeline-based approaches such as PGP and GPG, this work provides an initial investigation into how reversibility can impact error dynamics in multilingual DRS tasks. These early findings lay the groundwork for more detailed studies on reversible pipeline processing in semantic parsing.

The remaining chapter is organized as follows: in Section 6.2, we present language models and our proposed reversible pipeline approaches. Our multi-lingual experimentation is described in detail in Section 6.3. A detailed analysis of the results and a discussion of the findings are listed in Section 6.4. In Section 6.5, we exploit the error patterns in the pipeline approaches. In Section 6.6, we reveal the causes of pipeline success and failure, and in Section 6.7, we conclude the chapter.

6.2 Language Model and Reversible Pipelines

Our study departs from the standard rule-based and neural network-based methods for DRS parsing and text generation. We offer a novel perspective that takes advantage of the DRS reversible capabilities that do not require any explicit design of rules or external tools, in contrast to rule-based systems like Boxer or the more recent multilingual DRS parser which rely on hand-crafted rules and commercial dependency parsers Bos [2008], Poelman et al. [2022]. Instead, our work presents a pipeline-based approach for semantic parsing and text generation that takes advantage of the complementary benefits offered by pre-trained language models. Our approach cascades these reversible processes into two different pipelines, Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG), so as to identify error mitigation or amplification that might occur in the generation or parsing phase, without requiring extra rule engineering or model training.

In our PGP and GPG pipelines, we employed byT5 [Xue et al., 2022], a pre-trained language model fine-tuned on fully augmented DRS-Text pairs, achieving state-of-the-art performance in both semantic parsing and text generation tasks. To evaluate the impact of the pipeline approach, we utilized SMATCH for semantic parsing [Cai and Knight, 2013], while BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], ROUGE [Lin, 2004], COMET [Rei et al., 2020], chrF [Popović, 2015], and BERTScore [Hanna and Bojar, 2021] were applied to assess text generation outcomes.

6.2.1 Text-To-Text Transfer Transformer Model

In our experimentation, we employed the standard transformer model belonging to the Text-To-Text Transfer Transformers (T5) family [Unanue et al., 2023], specifically the byT5 [Xue et al., 2022] variant, due to its superior performance compared to other T5 variants, including mT5 [Xue et al., 2021] and T5 [Unanue et al., 2023] itself. Our approach deviates from traditional experimental methods as the conventional methods can be computationally expensive and time-consuming requiring frequent pre-training or fine-tuning of language models for task-specific applications. On the other hand, through this PGP or GPG pipeline approach, we do not require any additional model pre-training or fine-tuning.

6.2.2 Parse-Generate-Parse (PGP) Pipeline

The PGP pipeline is designed to identify error dynamics — mitigation or amplification — in the semantic parsing task by propagating the input text through three stages: parsing, generation, and parsing again. The pipeline operates as follows: (1) The input text is first processed by the parser model, which generates a DRS. (2) The generated DRS is then passed to the generator model, which produces a text output based on the DRS representation. (3) Finally, the generated text is fed into the same parser model, resulting in a new DRS representation. Figure 6.1 displays the graphical representation of the proposed PGP pipeline.

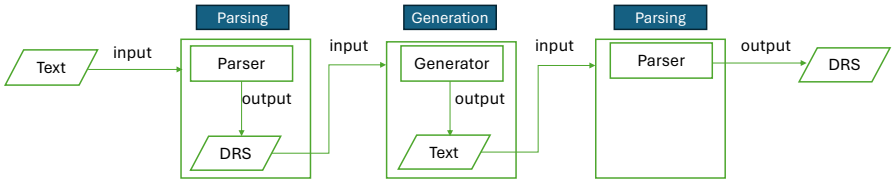


Figure 6.1: Graphical representation of PGP pipeline.

6.2.3 Generate-Parse-Generate (GPG) Pipeline

Similarly, the GPG pipeline is designed to identify error dynamics in the text generation task by propagating the input DRS through three stages: generation, parsing, and gener-

ation again. The pipeline operates as follows: (1) The input DRS is first processed by the generator model, which produces a text output. (2) The generated text is then passed to the parser model, resulting in a new DRS representation. (3) Finally, the parsed DRS is fed into the same generator model, producing a new text output. Graphically, the GPG pipeline is shown in Figure 6.2.

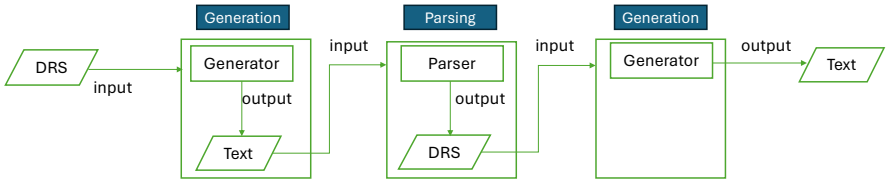


Figure 6.2: Graphical representation of GPG pipeline.

By iteratively propagating the data through these reversible pipelines, errors introduced in the initial parsing (generation) stage can be potentially analyzed in the subsequent generation (parsing) and parsing (generation) stages, leveraging the complementary strengths of the pre-trained models.

6.3 Multi-Lingual Experimentation

For our experiments, we have used two state-of-the-art models: a generator (DRS-to-Text) and a parser (Text-to-DRS), both based on byT5 and fine-tuned on a fully augmented PMB dataset. These models were used without any additional pre-training or fine-tuning. We evaluated our pipelines on three distinct languages—English (EN), Italian (IT), and Urdu (UR)—using the standard test set from the dataset. The results reveal complex patterns of performance changes across languages and metrics.

6.3.1 PGP Evaluation

The PGP pipeline was evaluated using SMATCH, an overlap-based metric typically used in AMR parsing [Cai and Knight, 2013], which computes the F1-score of matched triples between system-generated and gold standard DRS representations. The results indicate that the PGP pipeline generally retains parsing accuracy across multiple languages, but with variations depending on language complexity.

For English, the pipeline performed deterministically, with only a marginal decrease in SMATCH F1-score from 93.56 to 93.06, a mere 0.5% decrease. This demonstrates that the pipeline introduces minimal errors for semantic parsing tasks in a rich-resourced language i.e., English. For Italian, a slight decrease in the F1-score (from 90.56 to 89.19) was observed, representing a 1.37% decrease. While Italian's more complex sentence structure and grammar present challenges, the PGP pipeline still performs admirably, showing promise for further language-specific improvements. In Urdu, the F1-score decreased more noticeably, from 79.77 to 76.42 (a 3.35% drop), reflecting the greater chal-

Table 6.1: Experimental results of parsing and generation with and without pipeline approach. Underline represents the best results. Note: S-Par. = Semantic Parsing; S-F1 = SMATCH F1-Score; MET. = METEOR; CMT. = COMET; B_Scr. = BERTScore; RUG. = ROUGE.

Experimentation Type	Lang. Type	S-Par.	Generation Results					
		S-F1	BLEU	MET.	CMT.	chrF	B_Scr.	RUG.
without pipeline	EN	93.56	71.01	87.67	95.81	84.97	98.54	–
with pipeline		93.06	69.25	86.73	95.33	83.77	98.35	–
without pipeline	IT	90.56	56.76	72.67	89.97	70.59	92.85	–
with pipeline		89.19	53.06	69.68	88.53	67.54	91.88	–
without pipeline	UR	79.77	55.31	53.07	–	51.49	88.33	59.40
with pipeline		76.42	48.72	45.98	–	44.87	86.27	53.07

lenges posed by its rich morphology and syntax. Despite these challenges, the pipeline holds potential even without extensive pre-training or fine-tuning, suggesting that further adaptation could yield improved results for morphologically complex languages. Table 6.1 lists multi-lingual semantic parsing results comparing the PGP pipeline with the standalone parser.

The parsing performance breakdown (see Table 6.2) further highlights language-specific trends. For English, out of 1132 examples, 49 (4.33%) improved, 975 (86.13%) remained the same, and 108 (9.54%) showed decreased performance. Italian demonstrated similar trends with 29 (5.23%) improvements, 446 (80.36%) unchanged examples, and 80 (14.41%) showing decreased performance out of 555 examples. Urdu, however, showed the most variability, with 114 (12.66%) examples showing improvement, 449 (49.88%) remaining the same, and a notable 337 (37.44%) showing decreased performance out of 900 examples.

Table 6.2: Performance metrics of multi-lingual semantic parsing and generation indicating the total number of examples, with the number and percentage of improved, same, and decreased categories. (Note: Lang. = Language; and Ex. = Example)

Lang.	Imp. Type	Ex. Testset	Ex. Improved	Ex. Same	Ex. Decreased
English	Parsing	1132	49 (+4.33%)	975 (86.13%)	108 (-9.54%)
	Generation		35 (+3.09%)	1015 (89.66%)	82 (-7.24%)
Italian	Parsing	555	29 (+5.23%)	446 (80.36%)	80 (-14.41%)
	Generation		24 (+4.32%)	438 (78.92%)	93 (-16.76%)
Urdu	Parsing	900	114 (+12.66%)	449 (49.88%)	337 (-37.44%)
	Generation		114 (+12.66%)	401 (44.55%)	385 (-42.77%)

6.3.2 GPG Evaluation

For the GPG pipeline, we evaluated text generation performance using both rule-based (BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], chrF [Popović, 2015], ROUGE [Lin, 2004]), neural model-based (COMET [Rei et al., 2020]), and pre-trained model-based (BERTScore [Hanna and Bojar, 2021]) metrics to assess the quality

of generated text compared to reference text across English, Italian, and Urdu. The GPG pipeline maintains strong performance, especially for English text generation, with only minor declines across BLEU (71.01 to 69.25), METEOR (87.67 to 86.73), and chrF (84.97 to 83.77), indicating that the generated text remains highly comparable to the original output.

For Italian, although there was a slight decrease in BLEU (56.76 to 53.06), METEOR (72.67 to 69.68), and chrF (70.59 to 67.54), the GPG pipeline still performed commendably, demonstrating its capability to handle more linguistically diverse languages. In Urdu, despite its morphological complexity, the pipeline still captures the essence of sentence structure. However, larger declines in BLEU (55.31 to 48.72), METEOR (53.07 to 45.98), chrF (51.49 to 44.87), and ROUGE (59.40 to 53.07) indicate the need for further optimization in handling morphologically rich languages like Urdu. Table 6.1 list multi-lingual text generation results across different evaluation measures.

The generation performance breakdown complements these metric-based results. For English, 35 (3.09%) out of 1132 examples showed improvement, 1015 (89.66%) remained unchanged, and 82 (7.24%) showed decreased performance. In Italian, 24 (4.32%) out of 555 examples showed improvement, 438 (78.92%) remained the same, and 93 (16.76%) showed decreased performance. Urdu displayed the most variation, with 114 (12.66%) examples showing improvement, 401 (44.55%) remaining unchanged, and 385 (42.77%) showing decreased performance out of 900 examples—see Table 6.2 for percentage distribution across different languages.

In the broad spectrum of evaluation, both the PGP and GPG pipelines demonstrate potential for handling multilingual semantic parsing and text generation tasks. For English, the pipelines preserve much of the original performance with only minor fluctuations, underscoring their robustness. Even for Italian and Urdu, where challenges due to linguistic complexity are more pronounced, the pipelines provide a strong foundation for further improvements. The decrease in performance, particularly for Italian and Urdu, underscores areas for improvement but is balanced by the pipelines' overall effectiveness in multilingual contexts. The results indicate that with minimal language-specific adaptations, especially for Urdu, the pipeline is capable of generating multi-lingual high-quality results. These experiments pave the way for further exploration into how reversible semantic parsing and text generation can be leveraged to enhance semantic processing in a multilingual context.

6.4 Analysis and Discussion

To understand why our pipeline approaches (PGP and GPG) often result in error amplification rather than mitigation, we conducted a systematic analysis across five linguistic dimensions: sentence length (Section 6.4.1), sentence types (Section 6.4.2), structural complexity (Section 6.4.3), polarity (Section 6.4.4), and voice (Section 6.4.5). This multi-faceted and multi-lingual analysis aims to identify specific linguistic phenomena that may contribute to pipeline performance degradation.

6.4.1 Analyzing Impact of Sentence Length

To analyze the impact of sentence length on pipeline performance, we categorized sentences into three classes based on token count: short (0-4 tokens), medium (5-8 tokens), and long (9+ tokens). For token classification, we have adopted a rule-based custom tokenization strategy to split the sentences. Our analysis reveals significant distributional disparities between training and test sets across all three languages, which partially explains the suboptimal performance of our pipeline approaches. Table 6.3 shows the sentence splits corresponding to different sentence lengths based on tokens/words per sentence.

Table 6.3: Sentence length distribution by language and data type.

Lang.	**Data Type**	**Total Ex.**	**Sentence Splits (words/tokens)**		
			Short (%) (0–4)	**Medium (%)** (5–8)	**Long (%)** (9–)
English	Train	152788	3.81	44.48	51.72
	Test	1132	14.75	69.70	15.55
Italian	Train	5061	13.50	61.49	25.01
	Test	555	25.77	70.27	3.96
Urdu	Train	9057	13.07	68.12	18.80
	Test	900	18.33	66.78	14.89

In **English**, while the training data shows a natural distribution skewed towards longer sentences (51.72% long, 44.48% medium, 3.81% short), the test set exhibits a markedly different distribution with a strong bias towards medium-length sentences (69.70%) and notably higher representation of short sentences (14.75%). This distributional mismatch appears to impact pipeline effectiveness, as evidenced by consistent performance degradation across all metrics and length categories. The impact is particularly pronounced in short sentences, where the SMATCH score drops from 90.89 to 89.69, suggesting that the pipeline struggles with concise expressions where each token carries significant semantic weight.

Italian displays an even more pronounced distributional shift between training and test sets. The test data is heavily concentrated in the medium-length category (70.27%) with a notable overrepresentation of short sentences (25.77%) compared to training. This imbalance appears to particularly affect the pipeline's performance on long sentences, where we observe the most substantial degradation across metrics (e.g., BLEU score drops from 47.98 to 41.68). The scarcity of long sentences in the test set (3.96%) compared to training (25.01%) suggests that the model may not have developed robust handling of complex, lengthy expressions.

Urdu presents the most concerning performance degradation among the three languages, with substantial drops across all metrics and length categories. The medium-length sentences, despite being the most represented in both training (68.12%) and test (66.78%) sets, show a significant performance decline in pipeline processing (SMATCH drops from 81.32 to 77.40). This suggests that beyond distributional mismatches, structural characteristics of Urdu, such as its SOV word order and complex morphology, may be amplifying errors through the pipeline stages.

A cross-linguistic analysis reveals that medium-length sentences consistently achieve

the best baseline performance across all three languages, but also suffer from notable degradation in pipeline processing. This pattern suggests that, while these sentences contain enough information for robust semantic parsing, the pipeline's sequential nature introduces compounding errors that overwhelm any potential error correction benefits. The performance degradation is most pronounced in metrics that evaluate structural similarity and semantic accuracy (SMATCH, METEOR) rather than surface-level similarity (BLEU), indicating that the pipeline is particularly vulnerable to semantic drift during multiple transformation steps.

These findings suggest that the factors that contribute to the underperformance of the pipeline apparoch are: (1) distributional mismatches between training and test sets across sentence lengths, (2) language-specific structural characteristics that amplify errors through multiple transformations, and (3) the inherent challenge of maintaining semantic consistency through sequential processing steps. The consistent degradation across all metrics and languages indicates that our current pipeline architecture may need fundamental modifications to achieve effective error mitigation. Table 6.4 lists multilingual results with the utilization of the impact of sentence length on the performance of pipeline approaches.

Table 6.4: Impact of sentence length on evaluation results with and without pipeline for EN, IT, and UR. Bold indicates the better results.

Lang.	Imp. Type (ex.)	Pipeline	SMATCH (F1)	BLEU	METEOR	COMET	ROUGE	chrF	BERTScore
EN	Short (167)	Without	**90.89**	**71.04**	**84.13**	**95.47**	–	**82.25**	**98.46**
		With	89.69	68.17	83.04	94.43	–	80.50	98.06
	Medium (789)	Without	**94.85**	**71.89**	**88.66**	**96.34**	–	**85.85**	**98.65**
		With	94.47	70.29	87.78	95.99	–	84.77	98.52
	Long (176)	Without	**90.32**	**66.99**	**86.62**	**93.70**	–	**83.61**	**98.09**
		With	89.89	65.35	85.54	93.25	–	82.39	97.91
IT	Short (143)	Without	**90.52**	**53.61**	**63.14**	**87.62**	–	**63.44**	**90.27**
		With	89.48	49.32	60.15	85.65	–	59.89	89.14
	Medium (390)	Without	**90.66**	**58.41**	**76.22**	**90.98**	–	**73.29**	**93.79**
		With	89.14	55.07	73.39	89.86	–	70.55	92.95
	Long (22)	Without	**89.22**	**47.98**	**71.58**	**87.02**	–	**69.23**	**92.90**
		With	88.21	41.68	65.73	83.56	–	61.56	90.91
UR	Short (165)	Without	**79.14**	**52.17**	**49.20**	–	**56.90**	**49.60**	**87.43**
		With	76.46	44.86	40.93	–	49.17	41.59	85.34
	Medium (601)	Without	**81.32**	**57.38**	**55.29**	–	**60.99**	**52.97**	**88.87**
		With	77.40	51.35	48.97	–	55.49	47.20	87.07
	Long (134)	Without	**73.06**	**49.91**	**47.88**	–	**55.36**	**47.20**	**86.97**
		With	71.96	41.63	38.78	–	47.00	38.47	83.82

6.4.2 Performance Impact on Sentence Types

A systematic analysis of sentence type distributions reveals significant disparities between training and test sets across English, Italian, and Urdu. This imbalance manifests distinctly in each language, affecting the pipeline's error mitigation capabilities in different ways. Table 6.5 lists 4 different types of sentences present in the English, Italian, and Urdu data examples. We have used spaCy (https://spacy.io) to extract these sentence types from the dataset.

In **English**, the training data is heavily dominated by declarative sentences (86.76%),

Table 6.5: Sentence structure type distribution in training and test sets (EN, IT, UR).

Sentence Type	EN		IT		UR	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Declarative	86.76	61.31	87.39	87.57	93.82	87.22
Exclamatory	2.26	6.27	1.90	2.52	0.71	3.00
Imperative	2.06	0.80	0.57	0.18	0.76	0.89
Interrogative	8.91	31.63	10.14	9.73	4.71	8.89

while the test set shows a more balanced distribution with declarative sentences comprising 61.31%. This imbalance is further highlighted in interrogative sentences, where the test set proportion (31.63%) significantly exceeds the training representation (8.91%). The impact of this disparity is evident in the pipeline's performance: declarative sentences show performance degradation from baseline SMATCH of 93.44% to 92.98% with the pipeline. Interrogative sentences, despite their underrepresentation in training, maintain relatively robust performance with a modest SMATCH decline from 93.94% to 93.37%. Notably, exclamatory sentences, though comprising only 2.26% of training data, achieve the highest baseline SMATCH score (94.97%) but still experience degradation through the pipeline (94.23%).

Table 6.6: Impact of sentence type on evaluation results with and without pipeline for EN, IT, and UR. Bold indicates the better results.

Lang.	Sent. Type (ex.)	Pipeline	SMATCH (F1)	BLEU	METEOR	COMET	ROUGE	chrF	BERTScore
EN	Declarative (694)	Without	**93.44**	**72.75**	**89.58**	**95.93**	–	**86.10**	**98.73**
		With	92.98	70.75	88.66	95.39	–	84.87	98.54
	Exclamatory (71)	Without	**94.97**	**56.22**	**71.55**	**95.95**	–	**76.40**	**97.52**
		With	94.23	54.74	70.27	95.65	–	75.08	97.32
	Imperative (9)	Without	76.09	**77.40**	**95.38**	**96.28**	–	**87.25**	**99.33**
		With	**76.83**	74.04	94.49	96.11	–	84.97	**99.35**
	Interrogative (358)	Without	**93.94**	**70.39**	**86.98**	**95.53**	–	**84.41**	**98.34**
		With	93.37	68.98	86.06	95.14	–	83.34	98.17
IT	Declarative (486)	Without	**90.91**	**57.48**	**73.76**	**90.00**	–	**70.96**	**93.22**
		With	89.45	54.31	71.08	88.54	–	67.99	92.25
	Exclamatory (14)	Without	**91.92**	**47.27**	**59.42**	91.44	–	**65.80**	**86.95**
		With	88.66	46.39	59.40	**92.49**	–	65.82	86.59
	Imperative (1)	Without	83.33	18.99	32.25	71.11	–	18.63	79.09
		With	**91.66**	18.99	32.25	71.11	–	18.63	79.09
	Interrogative (54)	Without	**87.22**	**53.41**	**67.04**	**89.53**	–	**69.53**	**91.32**
		With	86.97	44.15	60.41	87.67	–	63.77	90.19
UR	Declarative (785)	Without	**79.72**	**54.93**	**52.56**	–	**59.18**	**50.46**	**88.27**
		With	76.25	48.11	45.11	–	52.64	43.60	86.17
	Exclamatory (27)	Without	**71.14**	**30.05**	**27.17**	–	**32.76**	**32.37**	**80.88**
		With	71.77	25.76	24.75	–	28.87	28.23	79.11
	Imperative (8)	Without	**72.06**	**31.84**	**33.72**	–	**34.63**	**37.64**	**79.16**
		With	62.25	22.63	24.82	–	25.83	29.68	77.35
	Interrogative (80)	Without	**83.90**	**69.90**	**68.72**	–	**73.04**	**69.45**	**92.28**
		With	81.02	64.96	63.80	–	68.13	64.48	90.55

Italian demonstrates a more stable distribution of declarative sentences between training (87.39%) and test (87.57%) sets, yet the pipeline still shows consistent performance degradation. The baseline SMATCH score for declarative sentences (90.91%) drops to 89.45% with the pipeline approach. Interrogative sentences, representing

10.14% of training and 9.73% of test data, show a significant performance decline across all metrics when processed through the pipeline, with SMATCH dropping from 87.22% to 86.97% and more dramatic drops in BLEU (53.41% to 44.15%) and METEOR (67.04% to 60.41%). Exclamatory sentences, despite limited representation, show notable baseline performance (91.92% SMATCH) but experience substantial degradation through the pipeline (88.66%).

Urdu exhibits the most pronounced training-test distribution stability for declarative sentences (93.82% training, 87.22% test) but shows the most severe pipeline performance degradation. Declarative sentences suffer a significant SMATCH drop from 79.72% to 76.25%. Interrogative sentences, despite having lower representation in both training (4.71%) and test (8.89%) sets, achieve the highest baseline performance among all Urdu sentence types (83.90% SMATCH) but still deteriorate with pipeline processing (81.02%). Imperative sentences, with minimal representation in both sets, show the most dramatic performance decline, with SMATCH dropping from 72.06% to 62.25% and substantial degradation across all other metrics.

The analysis reveals a consistent pattern of pipeline performance degradation across all three languages, though with varying severity. English shows the most resilient performance with relatively modest degradation across sentence types. Italian demonstrates moderate performance drops, particularly pronounced in semantic metrics. Urdu exhibits the most severe degradation, suggesting that language-specific structural characteristics may amplify the challenges posed by distributional imbalances. This cross-linguistic comparison indicates that the pipeline's error amplification tendency is influenced both by training-test distribution mismatches and by inherent linguistic complexities specific to each language. Table 6.6 lists multi-lingual results with the utilization of the impact of sentence types on the performance of pipeline approaches.

6.4.3 Analysis based on Structural Complexity

The distribution analysis based on structural complexity reveals significant imbalances across different sentence types in both training and test sets. In the training data, simple sentences dominate across all three languages, with English showing the most balanced distribution (70.18% simple, 14.30% complex, 9.40% compound, and 6.12% compound-complex). Italian and Urdu display an even stronger bias toward simple sentences (88.05% and 93.31% respectively), with minimal representation of other structures. This imbalance becomes even more pronounced in the test sets, where simple sentences constitute approximately over 94% of the data across all languages, and compound-complex sentences are entirely absent. We have used spaCy to classify sentences based on structural complexity from the dataset. Table 6.7 shows the percentage-wise structural distribution of sentences in the training and test sets for EN, IT, and UR.

For **English** language performance, the results present interesting variations across different sentence types. In simple sentences, which comprise the majority of the test set (1079 examples), the non-pipeline approach generally outperforms, achieving higher scores across most metrics (SMATCH: 93.79%, BLEU: 71.18%, METEOR: 87.63%). However, the pipeline approach shows promising results in complex sentences, marginally outperforming in SMATCH (85.65% vs 85.45%), though falling behind in other metrics. For compound sentences, the performance between the two approaches

Table 6.7: Training and test set structure type percentages.

Structure Type	EN		IT		UR	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Simple	70.18	95.32	88.05	98.20	93.31	94.89
Complex	14.30	1.94	5.77	0.54	2.49	2.22
Compound	9.40	2.74	4.64	1.26	4.09	2.89
Compound-complex	6.12	0.00	1.54	0.00	0.10	0.00

remains remarkably close, with the pipeline approach achieving slight advantages in BLEU (67.80% vs 67.58%) and BERT Score (98.12% vs 98.11%).

Italian language results demonstrate distinct patterns across different sentence structures. For simple sentences, which form the vast majority of the test set (545 examples), the non-pipeline approach consistently outperforms across all metrics. However, the most interesting results appear in compound sentences, where despite the small sample size (7 examples), the pipeline approach demonstrates superior performance across multiple metrics, including BLEU (65.39% vs 64.71%), METEOR (82.15% vs 79.54%), COMET (91.78% vs 89.36%), and others. This suggests that the pipeline approach might be particularly effective for handling compound structures in Italian, though the limited sample size warrants cautious interpretation.

Table 6.8: Impact of structural complexity on evaluation results with and without pipeline for EN, IT, and UR. Bold indicates the better results.

Lang.	Imp. Type (ex.)	Pipeline	SMATCH (F1)	BLEU	METEOR	COMET	ROUGE	chrF	BERTScore
EN	Simple (1079)	Without	**93.79**	**71.18**	**87.63**	**95.89**	–	**85.03**	**98.55**
		With	93.26	69.44	86.76	95.48	–	83.92	98.38
	Complex (22)	Without	85.45	**67.32**	**89.98**	**93.75**	–	**84.01**	**98.11**
		With	**85.65**	59.96	83.99	89.50	–	77.50	97.19
	Compound (31)	Without	**91.15**	67.58	87.45	**94.41**	–	83.27	98.11
		With	91.11	**67.80**	**87.45**	94.39	–	**83.22**	**98.12**
IT	Simple (545)	Without	**90.55**	**56.58**	**72.49**	**89.96**	–	**70.38**	**92.80**
		With	89.20	52.91	69.52	88.51	–	67.26	91.83
	Complex (3)	Without	**90.93**	**68.73**	**88.03**	**91.90**	–	**86.57**	**98.16**
		With	89.98	50.53	68.59	83.68	–	67.06	92.21
	Compound (7)	Without	**91.60**	64.71	79.54	89.36	–	80.22	94.08
		With	88.39	**65.39**	**82.15**	**91.78**	–	**81.22**	**95.63**
UR	Simple (854)	Without	**79.95**	**55.81**	**53.48**	–	**59.81**	**51.82**	**88.49**
		With	76.71	49.23	46.42	–	53.54	45.24	86.47
	Complex (20)	Without	**73.23**	**42.42**	**39.91**	–	**49.14**	**41.37**	**83.58**
		With	67.08	41.65	39.26	–	46.58	40.33	82.86
	Compound (26)	Without	**78.87**	**48.83**	**49.62**	–	**54.06**	**48.55**	**86.34**
		With	73.95	37.16	36.70	–	42.53	36.26	82.39

Urdu language results present a clear pattern favoring the non-pipeline approach across all sentence types and metrics. In simple sentences (854 examples), the non-pipeline approach maintains a significant lead across all metrics, with particularly notable gaps in BLEU (55.81% vs 49.23%) and METEOR (53.48% vs 46.42%). This pattern continues and even amplifies in complex and compound sentences, where the performance gaps become more pronounced. The compound sentences show the most dramatic differences, with the non-pipeline approach outperforming by substantial margins (e.g.,

BLEU: 48.83% vs 37.16%). All results are listed in Table 6.8.

The overall analysis reveals several key insights about structural complexity's impact on performance. Generally, performance tends to decrease as structural complexity increases across all languages. The gap between pipeline and non-pipeline approaches often widens with increased structural complexity, though this pattern varies by language. The results also highlight the challenge of evaluating performance on complex and compound structures due to limited sample sizes, particularly in Italian and Urdu. While the non-pipeline approach generally shows superior performance, the pipeline approach demonstrates specific strengths in certain contexts, particularly in Italian compound sentences and some aspects of English complex and compound sentence processing. These findings suggest that while the non-pipeline approach might be preferable as a general solution, there could be value in considering a hybrid approach that leverages the strengths of both methods in specific linguistic contexts.

This comprehensive analysis underscores the importance of considering both structural complexity and language-specific characteristics in developing and evaluating natural language processing systems. The varying performance patterns across different languages and sentence types suggest that a one-size-fits-all approach might not be optimal and that future developments might benefit from language-specific optimizations and structural considerations.

6.4.4 Polarity Impact on Performance

Polarity based distribution analysis reveals interesting patterns across languages in both training and test sets. English and Urdu show similar distributions with a strong bias toward affirmative sentences, while Italian presents a notably different pattern with a majority of negative sentences. Specifically, in the training set, English (84.73%) and Urdu (88.09%) heavily favor affirmative sentences, while Italian shows a reverse trend with 60.80% negative sentences. This pattern persists in the test sets, where English and Urdu maintain high percentages of affirmative sentences (91.34% and 90.00% respectively), while Italian continues its bias toward negative sentences (63.42%). We have used TextBlob (https://pypi.org/project/textblob/0.9.0/) to extract these sentence types from the dataset. Table 6.9 provides statistical numbers for affirmative and negative sentence types for EN, IT, and UR test sets.

Table 6.9: Training and test set polarity type percentages.

Polarity Type	**EN**		**IT**		**UR**	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Affirmative	84.73	91.34	39.20	36.58	88.09	90.00
Negative	15.27	8.66	60.80	63.42	11.91	10.00

For **English** language performance, the results show consistently strong performance across both affirmative and negative sentences, with the non-pipeline approach maintaining a slight edge. With affirmative sentences (1034 examples), the non-pipeline approach achieves better scores across all metrics (SMATCH: 93.56%, BLEU: 71.14%, METEOR: 87.89%). The performance on negative sentences (98 examples) is remark-

ably similar, with the non-pipeline approach again outperforming (SMATCH: 93.53%, BLEU: 69.64%, METEOR: 85.32%). The minimal performance difference between affirmative and negative sentences suggests that English processing is robust across polarity types.

Italian language results present an interesting case given its unique distribution favoring negative sentences. For affirmative sentences (203 examples), the non-pipeline approach shows strong performance (SMATCH: 90.18%, BLEU: 60.85%, METEOR: 76.15%). The performance on negative sentences (352 examples), which constitute the majority, remains strong with the non-pipeline approach (SMATCH: 90.78%, BLEU: 54.39%, METEOR: 70.66%). Notably, while the pipeline approach consistently trails behind, the performance gap remains relatively stable across both polarities, suggesting consistent handling of both sentence types.

Urdu language results reveal an interesting pattern where negative sentences, despite being the minority (90 examples), actually show slightly better performance than affirmative ones. The non-pipeline approach achieves higher SMATCH scores on negative sentences (82.45% vs 79.47% for affirmative), though other metrics remain comparable. This suggests that the processing of negative sentences in Urdu might be more straightforward than initially expected. The pipeline approach maintains the same pattern but with lower overall scores, showing larger performance gaps compared to the non-pipeline approach. Table 6.10 provides results for affirmative and negative sentence types with and without pipeline for EN, IT, and UR test sets.

Table 6.10: Impact of sentence polarity (affirmative and negative) on evaluation results with and without pipeline for EN, IT, and UR. Bold indicates the better results.

Lang.	Imp. Type (ex.)	Pipeline	SMATCH (F1)	BLEU	METEOR	COMET	ROUGE	chrF	BERTScore
EN	Affirmative (1034)	Without	**93.56**	**71.14**	**87.89**	**95.78**	–	**85.14**	**98.53**
		With	93.07	69.37	86.88	95.32	–	83.98	98.36
	Negative (98)	Without	**93.53**	**69.64**	**85.32**	**96.09**	–	**83.14**	**98.61**
		With	92.86	67.58	85.16	95.52	–	81.58	98.30
IT	Affirmative (203)	Without	**90.18**	**60.85**	**76.15**	**92.15**	–	**74.94**	**93.77**
		With	89.14	57.98	73.59	90.94	–	72.08	92.77
	Negative (352)	Without	**90.78**	**54.39**	**70.66**	**88.69**	–	**68.09**	**92.32**
		With	89.22	50.22	67.42	87.13	–	64.77	91.37
UR	Affirmative (810)	Without	**79.47**	**55.36**	**53.23**	–	**59.51**	**51.59**	**88.28**
		With	76.25	48.47	45.92	–	52.89	44.86	86.20
	Negative (90)	Without	**82.45**	**54.85**	**51.65**	–	**58.46**	**50.59**	**88.65**
		With	77.92	50.85	46.48	–	54.66	44.93	86.89

The analysis reveals several key insights about polarity's impact on performance. First, the systems generally handle both polarities well, with relatively small performance variations between affirmative and negative sentences within each language. Second, the non-pipeline approach consistently outperforms across all languages and polarities, suggesting its robustness in handling different sentence types. Third, the unique distribution in Italian, with its preference for negative sentences, doesn't seem to negatively impact performance, indicating that the systems have adequately adapted to this linguistic characteristic.

These findings carry implications for system development and optimization. The consistent performance across polarities suggests that current approaches are well-

balanced in handling both affirmative and negative constructions. However, the persistent advantage of the non-pipeline approach indicates that maintaining semantic coherence through unified processing might be particularly important for preserving meaning across different polarity types. The results also highlight the importance of considering language-specific characteristics in system development, as demonstrated by the successful handling of Italian's negative-heavy distribution and Urdu's superior performance on negative sentences despite their minority status in the training data.

6.4.5 Analyzing the Impact of Sentence Voices

The distribution analysis based on sentence voices shows a strong bias toward active voice across all three languages in both training and test sets. In the training data, the distribution is remarkably similar across languages, with active voice dominating at 90.58% for English, 92.06% for Italian, and 92.01% for Urdu. This pattern becomes even more pronounced in the test sets, where active voice sentences increase to 93.37%, 94.05%, and 93.78% respectively. The consistency of this distribution across languages suggests a universal preference for active voice constructions in natural language. We have used spaCy to classify these sentences based on the voice types from the dataset. Table 6.11 presents active and passive voice examples in training and test sets of EN, IT, and UR datasets.

Table 6.11: Training and test set voice type percentages.

Voice Type	**EN**		**IT**		**UR**	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
Active	90.58	93.37	92.06	94.05	92.01	93.78
Passive	9.42	6.63	7.94	5.95	7.99	6.22

English language results reveal some fascinating patterns in the handling of voice types. For active voice sentences (1057 examples), the non-pipeline approach demonstrates superior performance across all metrics (SMATCH: 93.57%, BLEU: 70.33%, METEOR: 87.36%). However, the most interesting findings emerge in passive voice sentences (75 examples), where we see a mixed pattern of success. The pipeline approach achieves a higher SMATCH score (94.88% vs 93.32%), marking one of the few instances where it outperforms the non-pipeline approach. Despite this, the non-pipeline approach maintains higher scores in other metrics for passive constructions, with notably higher BLEU (80.44% vs 78.21%) and METEOR (92.00% vs 90.36%) scores. Interestingly, both approaches achieve better scores on several metrics for passive sentences compared to active ones, suggesting that passive constructions, though less frequent, might be more straightforward to process.

Italian language performance shows a clear preference for the non-pipeline approach across both voice types. With active voice sentences (522 examples), the non-pipeline approach consistently outperforms (SMATCH: 90.46%, BLEU: 57.34%, METEOR: 73.07%). For passive voice sentences (33 examples), despite the small sample size, the non-pipeline approach maintains its advantage with higher scores across all metrics (SMATCH: 92.19%, BLEU: 47.55%, METEOR: 66.23%). Notable is the fact that while

SMATCH scores are actually higher for passive sentences, other metrics show lower performance compared to active voice, suggesting that while semantic preservation might be easier in passive constructions, generating natural language output becomes more challenging.

Urdu language results demonstrate a consistent pattern favoring the non-pipeline approach, but with some interesting nuances between active and passive voice handling. For active voice sentences (844 examples), the non-pipeline approach shows strong performance (SMATCH: 79.85%, BLEU: 55.51%, METEOR: 53.40%). In passive voice sentences (56 examples), while the non-pipeline approach still outperforms, there's a slight decline in performance across most metrics (SMATCH: 78.54%, BLEU: 52.31%, METEOR: 48.04%). This suggests that Urdu might find passive constructions more challenging to process compared to active ones, unlike the pattern seen in English and Italian. All evaluation results are presented in Table 6.12.

Several key insights emerge from this analysis about the impact of voice on processing performance. First, the high proportion of active voice sentences in training data doesn't necessarily translate to better performance on active constructions — in fact, both English and Italian show higher SMATCH scores for passive voice sentences. Second, the pipeline approach shows particular promise in handling English passive constructions, achieving its most notable success in this category. Third, the impact of voice on performance varies significantly by language, with Urdu showing a different pattern from English and Italian.

Table 6.12: Impact of sentence voice (active/passive) on evaluation results with and without pipeline for EN, IT, and UR. Bold indicates the better results.

Lang.	Imp. Type (ex.)	Pipeline	SMATCH (F1)	BLEU	METEOR	COMET	ROUGE	chrF	BERTScore
EN	Active (1057)	Without	**93.57**	**70.33**	**87.36**	**95.81**	–	**84.65**	**98.51**
		With	92.93	68.57	86.47	95.31	–	83.48	98.32
	Passive (75)	Without	93.32	**80.44**	**92.00**	**95.75**	–	**89.40**	**98.93**
		With	**94.88**	78.21	90.36	95.64	–	87.84	98.84
IT	Active (522)	Without	**90.46**	**57.34**	**73.07**	**90.15**	–	**70.91**	**92.89**
		With	89.11	53.72	70.11	88.66	–	67.75	91.93
	Passive (33)	Without	**92.19**	**47.55**	**66.23**	**86.94**	–	**65.66**	**92.16**
		With	90.60	42.63	62.83	86.32	–	62.45	91.04
UR	Active (844)	Without	**79.85**	**55.51**	**53.40**	–	**59.64**	**51.64**	**88.31**
		With	76.44	48.97	46.33	–	53.38	45.08	86.31
	Passive (56)	Without	**78.54**	**52.31**	**48.04**	–	**55.83**	**49.31**	**88.56**
		With	76.06	44.77	40.67	–	48.31	41.66	85.77

These findings have important implications for system development and optimization. The successful handling of passive voice despite its lower representation in training data suggests that current approaches are robust in managing syntactic variations. However, the varying patterns across languages indicate that voice handling might benefit from language-specific optimizations. The superior performance of the pipeline approach on English passive constructions also suggests that decomposing complex syntactic transformations might be beneficial in specific linguistic contexts. Future developments might consider leveraging these insights to create more nuanced, language-aware approaches to handling voice variations.

6.5 Analysis of Error Patterns in Pipeline Processing

The pipeline approaches (PGP and GPG) exhibit complex error dynamics that warrant detailed analysis. Our investigation focuses on understanding why these pipeline approaches often fail to improve performance, by examining specific types of errors that emerge and propagate through the pipeline stages. This analysis reveals systematic patterns in how errors evolve and compound, providing insights into the limitations of pipeline processing for semantic parsing and generation tasks.

6.5.1 Semantic Parsing Errors

For semantic parsing, we have focused on the following aspects of logical representations that have contributed—though minorly—to the performance degradation.

Erroneous WordNet Sense Assignment: Our analysis begins with WordNet sense-related errors, which form a significant category of pipeline failures. When processing sentences like "Let's fly a kite." we observe how initial sense assignment errors in the parser stage can transform through the pipeline. While the standalone parser might assign an incorrect sense (e.g., fly.v.01), the pipeline process often amplifies this error rather than correcting it (refer to Table 6.13, example 1). The generation stage, working with this incorrect sense, produces text that further deviates from the original meaning. When this generated text is parsed again, it frequently results in even more divergent sense assignments, creating a pattern of progressive semantic drift.

Omission of Logical Concepts: The handling of logical concepts through pipeline stages reveals another critical area of failure. For instance, in processing questions like "Is your father Spanish?" we observe how temporal and logical concepts are progressively distorted through pipeline iterations (refer to Table 6.13, Example 2). The initial parsing might omit certain logical concepts (like "time.n.08 EQU now"), but rather than recovering these concepts, subsequent pipeline stages often introduce new, incorrect concepts or transform existing ones in ways that fundamentally alter the semantic representation. This pattern demonstrates how pipeline processing can progressively degrade the logical structure of the original meaning.

Generation of Incorrect Thematic Roles: Thematic role assignments show particularly problematic behavior in pipeline processing. Taking the example "I caught a fish!" we observe how initial role assignments can deteriorate through pipeline stages (refer to Table 6.13, Example 3). What might begin as a simple Agent/Recipient confusion in the parser can evolve into more complex role misassignments through generation and subsequent parsing. The pipeline stages tend to amplify these role confusions, often resulting in semantically incoherent representations that bear little resemblance to the original meaning structure.

Erroneous Index Assignment: Index coherence presents another significant challenge in pipeline processing. Examining cases like "Mayuko designed a dress for herself." we find that referential relationships often break down through pipeline iterations. An initial

index error (such as confusing "Beneficiary +1" with "Beneficiary +3") typically leads to cascading failures in maintaining coreference relationships (refer to Table 6.13, Example 4). Each pipeline stage tends to introduce new indexing inconsistencies, ultimately resulting in DRS representations that fail to capture the intended referential structure of the original text.

Table 6.13: Analyzing error patterns through the lens of semantic parsing.

Gold Text	**Pars (DRS)**	**Pars-Gen (Text)**	**Pars-Gen-Pars (DRS)**	**Gold DRS**
Let's fly a kite.	time.n.08 TSU now person.n.01 EQU speaker fly.v.01 Time -2 Agent -1 Theme +1 kite.n.03	Let's fly kites.	time.n.08 TSU now person.n.01 EQU speaker fly.v.01 Quantity + Time -2 Agent -1 Theme +1 kite.n.03	time.n.08 TSU now person.n.01 EQU speaker fly.v.05 Time -2 Agent -1 Theme +1 kite.n.03
Is your father Spanish?	person.n.01 EQU hearer person.n.01 Role +1 father.n.01 Of -2 be.v.03 Theme -2 Source +1 country.n.02 Name "spain"	Your father is Spanish.	person.n.01 EQU hearer person.n.01 Role +1 father.n.01 Of -2 time.n.08 EQU now be.v.03 Theme -3 Time -1 Source +1 country.n.02 Name "spain"	time.n.08 EQU now person.n.01 EQU hearer person.n.01 Role +1 father.n.01 Of -2 be.v.03 Time -4 Theme -2 Source +1 country.n.02 Name "spain"
I caught a fish!	person.n.01 EQU speaker catch.v.08 Recipient -1 Time +1 Theme +2 time.n.08 TPR now fish.n.01	I myself caught a fish.	person.n.01 EQU speaker catch.v.08 Recipient Experiencer Of -1 Time +1 Theme +2 time.n.08 TPR now fish.n.01	person.n.01 EQU speaker catch.v.08 Agent -1 Time +1 Theme +2 time.n.08 TPR now fish.n.01
Mayuko designed a dress for herself.	female.n.02 Name "Mayuko" design.v.03 Agent -1 Time +1 Result +2 dress.n.01 Beneficiary +1 time.n.08 TPR now female.n.02 ANA -4	Mayuko designed this dress on time for herself.	female.n.02 Name "Mayuko" design.v.03 Agent -1 Time +1 Result +2 Time +2 Beneficiary +1 time.n.08 TPR now dress.n.01 female.n.02 ANA -4	female.n.02 Name "Mayuko" design.v.03 Agent -1 Time +1 Result +2 Beneficiary +3 time.n.08 TPR now dress.n.01 female.n.02 ANA -4

6.5.2 Generation Errors

The analysis of generation-related errors in pipeline processing reveals complex patterns of error emergence and propagation. Rather than focusing on correction mechanisms, our investigation examines why and how generation errors persist and often amplify through pipeline stages, particularly in the GPG configuration. This analysis provides insights into the fundamental challenges of maintaining semantic fidelity through multiple processing stages.

Grammatical Inaccuracies: Grammatical coherence represents a significant challenge in pipeline processing. When processing DRS representations like "high.a.02 Value ? AttributeOf +1 mountain.n.01 Name 'Mount Kinabalu'", we observe how initial grammatical errors can trigger more complex semantic misinterpretations through pipeline stages. What begins as a simple grammatical error (e.g., "How high of Mount Kinabalu?") often leads to more severe semantic distortions in subsequent pipeline stages, Table 6.14, Example 1. This pattern suggests that grammatical errors are not merely

surface-level issues but can fundamentally affect how meaning is preserved through pipeline iterations.

Table 6.14: Analyzing error patterns through the lens of text generation.

Gold DRS	Gen (Text)	Gen-Pars (DRS)	Gen-Pars-Gen (Text)	Gold Text
high.a.02 Value ? AttributeOf +1 mountain.n.01 Name "Mount Kinabalu"	How high of Mount Kinabalu?	high.a.02 Time +1 AttributeOf +2 time.n.08 EQU now mountain.n.01 Name "Mount Kinabalu"	High is Mount Kinabalu.	How high is Mount Kinabalu?
person.n.01 Name ? found.v.02 Agent -1 Time +1 Theme +3 time.n.08 TPR now striptease.n.02 club.n.07 Name "Chippendale" Theme -1	Who founded the striptease club Chippendale?	person.n.01 Name ? found.v.01 Agent -1 Time +1 Theme +3 time.n.08 TPR now striptease.n.01 club.n.06 Name "Chippendale" Theme -1 club.n.06 EQU -1	Who found the striptease club Chippendale club?	Who founded the Chippendale striptease club?
male.n.02 Name "Jack" book.n.01 Creator -1 time.n.08 EQU now interesting.a.01 AttributeOf -2 Time -1	Jack's books are interesting.	male.n.02 Name "Jack" book.n.01 User -1 time.n.08 EQU now interesting.a.01 AttributeOf -2 Time -1	Jack his book is interesting.	Jack's book is interesting.
entity.n.01 EQU ? be.v.06 Theme -1 Co-Theme +1 square_root.n.01 Of +1 number.n.02 EQU 100	What is the square root of a hundred?	entity.n.01 EQU ? be.v.02 Co-Theme -1 Time +1 Theme +2 time.n.08 EQU now square_root.n.01 PartOf +1 entity.n.01 Quantity +1 quantity.n.01 EQU 100	What is the square root value of the number 100?	What's the square root of 100?

Word Position Misalignment: Word order handling reveals another critical failure mode in pipeline processing. Analysis of cases like "person.n.01 Name ? found.v.02 Agent -1 Time +1 Theme +3 time.n.08 TPR now striptease.n.02 club.n.07 Name 'Chippendale' Theme -1" demonstrates how word position errors can amplify through pipeline stages. When the generator produces text with incorrect word order (e.g., "Who founded the striptease club Chippendale?" see Table 6.14, Example 2), subsequent parsing often introduces additional semantic distortions, as the parser struggles to reconstruct the intended logical relationships from the scrambled input. This cascading effect shows how seemingly minor word order issues can lead to significant semantic drift.

Singular-Plural Discrepancies: Number agreement presents a particularly interesting pattern of error propagation. Examining cases like the DRS for "Jack's book is interesting", we observe how singular-plural inconsistencies introduced in generation can lead to more fundamental semantic misrepresentations through pipeline stages. While the initial generation error (producing "Jack's books are interesting." see Table 6.14, Example 3) might seem minor, it often triggers more complex errors in subsequent parsing, as the change in number can affect logical relationships and quantification in the semantic representation. This demonstrates how morphological errors can evolve into deeper semantic inconsistencies.

Textual Representation Variations: The problem of textual variation manifests uniquely in pipeline processing. When analyzing cases like the DRS for questions about square roots, we find that seemingly innocuous textual variations (such as representing "100" as "a hundred") can lead to unexpected semantic divergences through pipeline iterations—see Table 6.14, Example 4. While these variations might preserve meaning in isolation, they often trigger parsing errors in subsequent stages, as the parser may interpret the paraphrased expressions differently from their original forms. This reveals a fundamental tension between semantic equivalence and textual consistency in pipeline processing.

6.5.3 Cross-Lingual Analysis

Cross-linguistic analysis reveals distinct error patterns in semantic processing that vary significantly across languages, shaped by their underlying structural characteristics. In English, with its relatively simple morphology, error patterns emerge in more predictable ways, primarily manifesting as issues with sense assignments and logical concept handling during parsing, while generation tasks show particular sensitivity to grammatical and word order errors. Italian, possessing a richer morphological structure, presents more complex challenges. During parsing, these manifest primarily as difficulties in thematic role assignments, while generation tasks frequently stumble over number agreement and morphological consistency. Urdu, with its distinctive combination of word order and complex morphology, demonstrates the most severe degradation patterns of all three languages. Its pipeline processing shows pronounced deterioration across all categories, with particular challenges in maintaining word order flexibility and agreement systems through both parsing and generation stages.

Our analysis also reveals fundamental issues with pipeline approaches that transcend specific languages. Each stage of the pipeline introduces its own potential for error, and these errors demonstrate a strong tendency to amplify rather than mitigate through subsequent processing stages. When parsing errors occur, they lead to the creation of imperfect DRS. These imperfect representations then serve as input for the generation stage, which produces text that increasingly deviates from the original meaning. This creates a cyclical pattern of semantic drift, where each pass through the pipeline amplifies the deviation from the intended semantic content, resulting in progressively degrading output quality.

The implications of these findings suggest that the challenges in semantic processing are more fundamental than previously understood. Rather than viewing pipeline approaches as potential error correction mechanisms, the evidence points toward the need for more robust standalone models that can handle complex semantic representations directly. The focus should shift toward developing architectures that maintain semantic consistency without requiring multiple transformation stages. This is particularly crucial for maintaining fidelity across the full spectrum of language-specific features, from morphological complexity to word order flexibility.

Understanding these failure modes has significant implications for future research in both semantic parsing and generation. The field needs to address how to better handle language-specific morphological features while maintaining semantic fidelity throughout the processing chain. This requires not just improvements in individual components but a re-conceptualization of how we approach semantic processing tasks altogether. The

patterns of error propagation identified in this analysis highlight the need for integrated approaches that can maintain semantic consistency more effectively than current pipeline-based methods, particularly when dealing with the complex interplay between semantic representation and surface realization across diverse linguistic structures.

6.6 Revealing the Pipeline Approach

The PGP and GPG pipeline configurations, designed to leverage the reversible nature of DRS, were intended to improve the semantic parsing and generation tasks by mitigating errors. However, the results clearly demonstrate that, rather than achieving this goal, the pipelines frequently failed, often amplifying errors across multiple stages. This section critically examines why the pipeline approaches were unsuccessful, focusing on experimental results and concrete examples from the findings.

Error Amplification Instead of Mitigation

A critical observation from the experimental results is the consistent error amplification across the pipeline, particularly in languages with complex morphology and syntax. For instance, in the PGP pipeline, parsing accuracy measured through SMATCH showed minimal degradation in English, dropping from 93.56% to 93.06%—a modest 0.5% decrease. However, in Italian, this decline was more pronounced, from 90.56% to 89.19% (1.37% drop), while Urdu saw a substantial 3.35% decline, from 79.77% to 76.42% (see Table 6.1).

These results highlight that while English, with its relatively simple morphology, demonstrated resilience to error propagation, both Italian and Urdu experienced error amplification. The GPG pipeline followed a similar trend in the generation task (listed in Table 6.1), where English BLEU scores dropped slightly from 71.01 to 69.25, but Italian and Urdu showed significant declines, with Italian's BLEU score falling from 56.76 to 53.06 and Urdu from 53.31 to 48.72. This suggests that the reversible nature of the pipelines, while theoretically promising, resulted in greater error accumulation in languages with more complex structures.

Linguistic Complexity and Cross-Linguistic Variation

The linguistic complexity of Italian and Urdu proved to be a major challenge for the pipeline approach. English, being morphologically simpler, maintained a relatively stable performance across both parsing and generation tasks. However, the experimental results for Italian and Urdu clearly indicated higher variability and error propagation.

For example, in the PGP pipeline, while 86.13% of English examples remained unchanged in terms of parsing performance, only 80.36% of Italian and 49.88% of Urdu examples showed similar stability—see Table 6.2. The error propagation was particularly severe in Urdu, where 37.44% of examples exhibited decreased performance (compared to only 9.54% in English). These results suggest that the pipeline approach struggled to handle morphological richness and syntactic flexibility, particularly in Urdu, where both parsing and generation tasks were heavily impacted.

Semantic Drift and Structural Mismatches

One of the most significant issues in the pipeline approach was semantic drift, where the output progressively diverged from the intended meaning. This was particularly evident in the generation tasks in the GPG pipeline, where minor initial errors in word order or thematic role assignments led to cascading errors.

For instance, in the sentence "Mayuko designed a dress for herself," initial parsing errors in thematic role assignment (incorrectly handling the Beneficiary role) were amplified through subsequent pipeline stages (see Table 6.13). The final output showed a complete mismatch in logical structure, with the Beneficiary role shifted, leading to incorrect sentence generation. Similar errors were observed in the sentence "I caught a fish," where an initial Recipient/Experiencer confusion resulted in the parser misinterpreting the roles, causing further degradation in meaning—example taken from Table 6.13.

The data also showed that longer and more complex sentences were particularly prone to semantic drift. For example, English's SMATCH score for short sentences (0-4 tokens) dropped from 90.89 to 89.69, while long sentences (9+ tokens) showed a decline from 90.32 to 89.89. In Urdu, where medium-length sentences were most common, the decline was steep, with SMATCH dropping from 81.32% to 77.40%, reflecting the pipeline's inability to handle complex morphological and syntactic structures effectively (see Table 6.4).

Mismatch Between Surface Form and Semantic Content

Another critical issue was the pipeline's difficulty in preserving the deep semantic content encoded in DRS while transforming it into surface text. This problem was particularly evident in the GPG pipeline, where the generated text often deviated from the semantic intent of the original DRS. For instance, the BLEU and METEOR scores for generated text across all languages decreased consistently, indicating that the text generation phase introduced disfluent or semantically inconsistent output.

In Italian, BLEU scores dropped by 3.7 points and METEOR by 3 percentage points in the GPG pipeline, reflecting substantial surface form inconsistencies. The problem was even more severe in Urdu, where the METEOR score fell from 53.07 to 45.98, a 7% decrease, highlighting significant issues with maintaining semantic coherence—see Table 6.1. The inability to correct surface-level errors through pipeline processing, especially in languages with complex grammar, underscores a fundamental limitation of the pipeline approach in balancing surface fluency with semantic accuracy.

Handling of Linguistic Ambiguity

The pipelines struggled with linguistic ambiguity, particularly in sentences with polysemy or multiple possible interpretations. This was most evident in Italian and Urdu, where ambiguous structures resulted in divergent parsing and generation outputs. For instance, when parsing sentences with multiple potential WordNet sense assignments, the pipelines failed to correct initial errors, amplifying the confusion through subsequent stages. The parsing error rates were particularly high for ambiguous sentences in Urdu, where more than 42.77% (see Table 6.2) of the generated examples exhibited a

decline in performance. This pattern suggests that the models employed in the pipeline lacked the robustness needed to handle ambiguous language phenomena, particularly in low-resource languages like Urdu, where training data scarcity exacerbated the problem.

Inability to Correct Logical and Thematic Role Errors

A key weakness of the pipeline approach was its inability to correct logical errors and thematic role misassignments. Errors introduced in the initial parsing or generation stage often persisted throughout the pipeline, leading to compounding issues. For example, in the DRS-to-text generation task, errors in thematic role assignment for sentences like "Your father is Spanish" or "I caught a fish" were not corrected in the subsequent parsing stage, resulting in further semantic inconsistencies (see 3rd and 4th example in Table 6.13).

The breakdown of results showed that thematic role errors contributed to a significant portion of the performance decline, particularly in Urdu, where parsing errors frequently resulted in incorrect logical structures. The SMATCH score for Urdu parsing dropped significantly, from 79.77% to 76.42% (exemplified in Table 6.1), largely due to the pipeline's inability to correct such errors.

When and Why does the pipeline work?

Considering the question "When and Why does the pipeline work?", we provide here some speculations related to example 3 of Table 6.14. We note that the singular/plural feature is not explicitly denoted in the DRS, but it is only implicitly represented by the name "Jack". Moreover, we note that the only difference between the original input and the Gen-Pars output is the presence of the thematic role `USER` in contrast to `CREATOR`. Searching in the training set we found that the `USER` role has 729 instances while `CREATOR` has 220 instances. We can speculate that the standalone generator is not able to account for the standard singular form related to "Jack" since its original role, that is `CREATOR`, is not frequent in the training set. In contrast, the Gen-Pars-Gen system is able to realize the singular form of the verb since it has a more frequent semantic role, that is `USER`. In other words, we speculate that the role of the pipeline is to "correct" the input toward a more standard form, that is to transform the original input into a form closer to the instances that are in the training set. As part of this ongoing research, we are conducting a deeper investigation into the underlying factors that could logically account for the success observed in pipeline approaches.

6.7 Chapter Conclusion

This chapter systematically investigated the reversible nature of semantic parsing and text generation through DRS, leveraging pipeline approaches across three languages: English, Italian, and Urdu. The primary objective was to assess the impact of two distinct pipeline configurations—PGP and GPG—on error propagation or mitigation without additional model training. By employing pre-trained language models, the study explored how these reversible processes influence the performance of both parsing and generation tasks, providing valuable insights into cross-linguistic error dynamics.

The key findings of this research demonstrate that, while the reversible pipeline approach offers the potential for correcting errors, it more frequently leads to error amplification, particularly in languages with complex morphology and syntactic structures, such as Urdu and Italian. The study revealed that English exhibited the greatest stability, with minimal performance degradation in both parsing and generation tasks. In contrast, Italian and Urdu experienced greater volatility, as errors introduced during one stage of the pipeline tended to amplify through subsequent stages.

Through a detailed analysis of error patterns across linguistic dimensions—sentence length, structural complexity, sentence type, polarity, and voice—this chapter provides an in-depth understanding of how specific language characteristics influence error propagation. The study shows that the reversible nature of DRS-based pipelines, while theoretically promising, is limited in practical effectiveness due to the compounding of errors in complex sentence structures and morphologically rich languages.

This research contributes to the field by proposing a novel framework for evaluating error propagation in multilingual DRS-based systems. It emphasizes the necessity of more robust models that maintain semantic fidelity across diverse linguistic contexts, rather than relying on reversible pipelines for error correction. By identifying both the potential and limitations of these approaches, the findings lay the groundwork for future advancements in semantic parsing and generation, advocating for the development of more integrated and language-specific solutions that address the inherent challenges of multilingual semantic processing.

Limitations: The potential of our PGP and GPG pipelines to exploit the task reversibility of DRS offers opportunities for effective error dynamics, whether through propagation or mitigation. However, the predominance of error propagation over error mitigation is attributed to the dependency of these pipeline approaches on pre-trained language models. In our experimental implementation, we utilized the best-performing models with state-of-the-art results for the languages involved. Yet, the data examples used to train the English DRS processing models vastly outnumbered those for Italian and Urdu, posing a challenge in terms of model generalization and robustness capabilities. Furthermore, the limitations of traditional evaluation metrics, such as SMATCH (which only considers structural overlap) and BLEU and METEOR (which are based on n-gram overlap), further complicate the assessment of these results. In our analysis, we resorted to human evaluation, which is computationally expensive and time-consuming. Additionally, our analysis has highlighted the linguistic imbalance across the various DRS variants, which also poses a limitation to the fair evaluation of the models. These findings suggest the need for a more balanced dataset to train models that can overcome these limitations and deliver the best possible results.

Chapter 7

Conclusions

DRS provides a formal, language-neutral framework for capturing semantic information, enabling texts with equivalent meanings across diverse languages to share a common representation. This characteristic makes DRS an effective tool for cross-linguistic semantic tasks, offering a consistent method to model meaning. The reversible nature of DRS-based semantic parsing (text-to-DRS) and text generation (DRS-to-text) further enhances its applicability, allowing for flexible and bidirectional processing of semantic information. This dissertation contributes to advancing these processes through innovative methodologies and multilingual applications, bridging gaps in semantic representation, processing, and evaluation.

Central to this dissertation is the enhancement of semantic parsing and text generation models for multilingual and low-resource contexts. Through novel data augmentation and delexicalization techniques, we addressed critical challenges, such as data scarcity and linguistic diversity, to strengthen model generalization and robustness. Our data augmentation strategies introduced Supersense-based enrichment, lexical transformations using WordNet relations (e.g., hypernyms, hyponyms, synonyms, antonyms, troponyms), and named entity substitutions in both in-context and out-of-context scenarios. Additionally, grammatical augmentations, such as tense transformations, added richness to the training data. These strategies were designed to improve semantic coherence while increasing the diversity and complexity of the datasets.

In parallel, we explored data delexicalization techniques aimed at reducing the dependency of models on specific lexical items. By leveraging Supersense and WordNet-based lexical substitutions, we ensured that the core semantic meaning was preserved while encouraging the models to generalize better across unseen examples. This focus on abstraction and generalization has practical implications for improving the adaptability of semantic parsing and generation models in varied linguistic settings.

To evaluate these contributions, through our preliminary investigation, we identified limitations in traditional evaluation methods that primarily rely on structural or surface-level criteria. To address this, we proposed bidirectional evaluation measures—Parse-Generate (PARS-GEN) and Generate-Parse (GEN-PARS)—which integrate structural accuracy with linguistic quality. These methods offer a comprehensive framework to assess the performance of DRS-based systems, capturing both semantic and

structural nuances often overlooked by traditional metrics such as SMATCH, BLEU, or BERTScore. By implementing these methods on datasets spanning English, Italian, and Urdu, we demonstrated their effectiveness in providing a holistic view of system performance.

Furthermore, we exploited the reversible nature of semantic parsing and text generation through pipeline approaches, specifically Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG). These pipelines were designed to investigate error propagation and mitigation in DRS-based tasks. While the experiments revealed that errors often propagate through the pipeline, we also identified instances of error correction, which provided critical insights into the dynamics of linguistic structure, sentence complexity, and polarity. These findings contribute to our understanding of error trends and support the development of more balanced datasets and models.

A significant contribution of this dissertation lies in its application of these methodologies to multilingual tasks, extending the predominantly English-focused research in this domain. By integrating Urdu and Italian, two linguistically distinct and underrepresented languages, we demonstrated the generalizability and adaptability of our approaches. For Urdu, we developed the first semantically annotated corpus, the Urdu Meaning Bank (UMB), which served as the foundation for semantic parsing and text generation tasks. For Italian, we introduced cross-lingual data augmentation strategies to address the challenges posed by limited data. These multilingual efforts underscore the importance of broadening semantic resources and methodologies to include diverse languages.

While much of the previous research in semantic parsing and generation has been focused on English, this dissertation is the first to apply these methodologies to multilingual tasks, specifically using English, Italian, and Urdu—three linguistically distinct languages. Our work introduces novel multilingual data augmentation strategies, data delexicalization techniques, robust evaluation measures, and the exploration of reversible pipeline approaches, marking a significant advancement in the field.

Below we revisit the research questions posed in the introduction section of this dissertation (see Chapter 1, Section 1.1), summarize the findings derived from our experiments, and suggest future research directions that could further build upon the contributions made in this work.

RQ1.1: Is it possible to augment a logical data representation such as DRS while maintaining semantic coherence?

This thesis investigates the feasibility of augmenting logical data representations, such as DRS, while preserving their semantic coherence. As discussed in Chapter3, Section 3.1, and Chapter 4, Section 4.3, augmenting DRS requires careful handling to avoid disrupting its inherent logical structure. By employing strategies such as named entity augmentation, WordNet-based lexical substitutions, and the adaptation of WordNet Supersenses, along with grammatical transformations like tense variations, this research demonstrates that new data can be generated while maintaining both semantic and contextual coherence. Specifically, our findings show that in-context and out-of-context lexical substitutions can be effectively utilized to preserve the semantic integrity of the

DRS, provided that these substitutions align with the original logical structure.

RQ1.2: How can we generate new data that is contextually similar to the original DRS representations?

In Chapters 3 and 4, Sections 3.2 and 4.3, we explore methods for generating new data that remains contextually similar to the original DRS representations. The thesis outlines a range of data augmentation techniques, such as named entity replacement, lexical substitutions (e.g., using WordNet), and SuperSense tagging, aimed at replacing key lexical categories like proper nouns, common nouns, and verbs, and other lexical categories like adverb and adjectives. These controlled variations introduce lexical diversity while ensuring that the augmented data stays aligned with the original DRS-text pairs. This process improves the generalization and robustness of models, enhancing their ability to handle new, contextually relevant data.

RQ1.3: What roles do in-context and out-of-context vocabulary play for character-level and word-level decoder models in DRS parsing and generation?

As detailed in Chapter 3, Section 3.6.1, this research examines the roles of in-context and out-of-context vocabulary for character-level and word-level decoder models in DRS parsing and generation. Our findings indicate that in-context vocabulary augmentation benefits word-level decoders, such as WB-bi-LSTM, by reinforcing known vocabulary and ensuring grammatical consistency. In contrast, character-level models, such as CB-bi-LSTM, demonstrate greater flexibility with out-of-context vocabulary, enabling broader generalization in sequence generation. These results highlight the importance of tailoring vocabulary strategies to the specific strengths of the model type, thereby optimizing performance for DRS tasks.

RQ1.4: How does grammatical, semantic, and pragmatic world knowledge influence the learning process in DRS-based tasks?

Chapter 3 and Chapter 4 address the influence of grammatical, semantic, and pragmatic world knowledge on the learning process for DRS-based tasks. Our research underscores the necessity of integrating multiple layers of linguistic knowledge—grammatical, semantic, and pragmatic—into neural models to effectively parse and generate DRS representations. Grammatical knowledge supports syntactic structure, while semantic knowledge aids in interpreting meaning within the DRS framework. Pragmatic and world knowledge, which encompasses contextual understanding and real-world facts, allows models to generate more appropriate and contextually relevant responses. We show that this multi-layered approach improves performance, particularly in handling complex phenomena like anaphora, negation, and quantification, while ensuring the semantic coherence of the generated outputs.

RQ1.5: Does augmentation lead to improved performance when training sequence-to-sequence models like Long Short-Term Memory (LSTM) or fine-tuning Transformer models for DRS tasks?

Chapter 3, Section 3.6, explores the impact of data augmentation on sequence-to-sequence models such as Long Short-Term Memory (LSTM) and fine-tuned Transformer models for DRS tasks. Our findings reveal that data augmentation significantly improves the performance of both model types across various evaluation metrics. Augmented data enhances generalization and robustness, as evidenced by improved BLEU, METEOR, and BERTScores for LSTM models. Transformer models, particularly byT5 (as presented in Chapter 4, Section 4.4.2), benefit even more from augmentation, showing substantial gains in capturing complex semantic structures during DRS-to-text generation tasks. This demonstrates that both LSTM and Transformer models leverage augmented data to improve performance, with Transformer models exhibiting the most pronounced improvements due to their advanced architecture and handling of intricate augmentation schemes.

RQ1.6: How do pre-trained large language models (LLMs) like ChatGPT and Claude interpret and process DRS structures when given as prompts?

In Chapters 3 and 4, Sections 3.7.2 and 4.7, we analyze how LLMs such as ChatGPT and Claude interpret and process DRS structures when presented as prompts. Our results indicate that although these models are capable of handling general-purpose language tasks, they struggle to accurately parse and generate text based on structured, domain-specific data like DRS. In both zero-shot and few-shot settings, these models often produce text that reflects logical explanations rather than direct translations of DRS examples. This limitation is primarily due to the lack of specialized training on formal semantic representations, which impacts the accuracy of LLMs in handling DRS-based tasks.

RQ1.7: How does the quality of augmented data, characterized by semantic and contextual accuracy, influence the effectiveness of data augmentation in enhancing performance in semantic parsing and natural language generation tasks?

Chapter 4, particularly Section 4.5, addresses how the quality of augmented data impacts the performance of models in semantic parsing and natural language generation tasks. The research demonstrates that high-quality, semantically accurate augmented data significantly enhances model effectiveness. This improvement was particularly evident when models were trained on datasets enriched with well-curated augmentations, such as named entity modifications, lexical substitutions, and grammatical transformations. These enhancements preserved contextual and semantic fidelity, allowing models to better capture complex language constructs, such as tense and pragmatic context, across English, Italian, and Urdu. The study highlights that augmentations which maintain both semantic coherence and contextual relevance are instrumental in improving the model's generalization capabilities and robustness. This effect is especially pronounced in low-resource settings, where data quality can compensate for limited dataset sizes, leading to more reliable performance in parsing and generation tasks.

RQ1.8: What is the relative contribution of manually corrected (gold) data com-

pared to larger volumes of potentially less accurate silver data in improving model performance?

Chapter 4, Section 4.5.1, examines the comparative impact of gold (manually corrected) and silver (partially verified) datasets on model performance. The research demonstrates that while large volumes of silver data contribute to enhancing model performance, the inclusion of high-quality, manually corrected gold data yields the most substantial performance gains, especially in nuanced tasks like DRS parsing and text generation. The findings indicate that a balanced combination of gold and augmented silver data provides optimal results. Models trained on both data types achieved higher accuracy and generalizability compared to models relying predominantly on silver data. This result suggests that high-quality annotations in the gold data play a critical role in improving the accuracy of semantic parsing and generation tasks, compensating for limitations in data volume.

RQ2.1: How can we effectively delexicalize DRS representations while maintaining their connections to external lexical databases like WordNet and VerbNet?

Chapter 3, Section 3.4 of this thesis addresses the effective delexicalization of DRS representations while maintaining links to external lexical databases such as WordNet and VerbNet. Our study developed a delexicalization method that carefully replaces proper nouns and common nouns with placeholders that map to semantic categories in WordNet and VerbNet. By employing supersenses for nouns, our approach allows models to generalize across different lexical entities while preserving essential semantic and syntactic relationships within the DRS. This ensures that the delexicalized representations retain the necessary connections to external resources, enhancing the models' ability to handle logical consistency and contextual information during parsing and generation tasks.

RQ2.2: Can the use of supersenses for nouns contribute to enhancing the generalization power of neural models in DRS tasks?

The use of supersenses for nouns is shown to enhance the generalization capability of neural models in DRS tasks by abstracting semantic information, thus allowing models to better recognize patterns across varying lexical contexts. By using supersenses, particularly in data delexicalization, the model is equipped with broader semantic categories (e.g., "noun_artifact" or "noun_substance") rather than specific lexical items. This abstraction mitigates dependency on specific terms and improves model adaptability to new or less frequent lexical entities as evident in Chapter 3, Section 3.4.2.

RQ2.3: What is the impact of combining logically delexicalized data with fully lexicalized data on model performance?

Integrating logically delexicalized data with fully lexicalized data has been found to yield substantial improvements in model performance. This combination provides a balanced dataset where the model learns both abstracted structures and specific lexical

details, enhancing its capacity to generate accurate and contextually relevant text. Results indicate that the inclusion of both data types benefits sequence-to-sequence tasks by increasing model robustness and accuracy, as demonstrated in the experiments detailed in Chapter 3 Section 3.7.

RQ2.4: How do delexicalization and augmentation techniques interact to affect model performance?

Delexicalization and data augmentation interact synergistically to boost model performance in DRS tasks. Specifically, delexicalization reduces model dependency on specific lexical items, while augmentation with supersense-tagged nouns provides varied syntactic and semantic structures. This approach enables the model to better capture diverse linguistic contexts, leading to improved generalization across novel inputs. The combination of these techniques is extensively evaluated in Chapter 3, showcasing notable enhancements in model accuracy and consistency.

RQ2.5: What are the differences in behavior between pre-training and fine-tuning approaches when applied to delexicalized DRS data?

Pre-training and fine-tuning on delexicalized DRS data show distinct behaviors. Pre-training allows models to grasp general semantic structures from abstracted data, which aids in understanding DRS representations broadly. Fine-tuning, on the other hand, adapts the model to specific syntactic and lexical nuances relevant to the DRS-to-text generation task, especially with added supersenses in delexicalized data. Experiments in Chapter 3 highlight how bi-LSTM pre-training combined with fine-tuning using byT5 achieves better alignment between abstracted structures and task-specific language nuances.

RQ3.1: How can we create a high-quality, semantically annotated corpus for Urdu that is freely available for research purposes?

The Urdu Meaning Bank (UMB) was developed to address the lack of high-quality, semantically annotated resources for Urdu. This corpus, generated by translating English logic-text pairs from the PMB-5.0.0 into Urdu, underwent extensive manual correction to align DRS concepts with Urdu's unique syntactic order. With 1,200 gold and 6,857 silver training examples, and a ninefold increase in examples through multi-faceted data augmentation, this resource offers a foundational tool for advancing Urdu NLP tasks in parsing and generation (Chapter 4, Section 4.2).

RQ3.2: What are the challenges and solutions in developing effective semantic parsing and generation models for Urdu?

Developing effective semantic parsing and generation models for Urdu faces challenges including Urdu's rich morphology, unique script, and limited lexical resources. Solutions include data augmentation to address resource scarcity and cross-lingual adaptation, which leverages existing English semantic resources to create an Urdu-specific

corpus. Further, specialized neural models are fine-tuned to capture Urdu's syntactic and semantic nuances, enhancing model adaptability in low-resource settings (Chapter 4, Section 4.2).

RQ3.3: How can we adapt and apply sound semantic data augmentation approaches to Urdu, ensuring the generation of contextually similar and semantically correct data?

To ensure contextually similar and semantically correct data in Urdu, data augmentation was carefully adapted to respect Urdu's linguistic structure. Techniques such as lexical substitutions, named-entity augmentation, and grammatical augmentation through tense variations were employed to enrich the dataset without compromising grammatical or semantic accuracy, thereby improving model robustness. This approach facilitated a significant expansion in training data and strengthened the performance of Urdu parsing and generation models (Chapter 4, Section 4.3.

RQ3.4: To what extent does data augmentation enhance the generalization power of parsing and generation models for Urdu?

Data augmentation has been shown to enhance the generalization capabilities of Urdu semantic parsing and generation models by expanding data diversity. Using individual and compound augmentations, experiments indicated a notable improvement in performance metrics, such as BLEU, METEOR, and ROUGE, over non-augmented data. Compound augmentation, integrating multiple augmentation strategies, yielded the highest gains, indicating its efficacy in boosting model generalization (Chapter 4, Section 4.5.3).

RQ3.5: How does the performance of Urdu semantic processing (parsing and generation) compare to that of other languages, and what insights can be gained from this comparison?

Urdu's semantic processing performance, when compared to languages like English and Italian, reveals valuable cross-linguistic insights. Although English, as a high-resource language, maintains superior performance across metrics, Urdu's augmented models achieved BLEU and METEOR scores competitive with low-resource European languages, highlighting the effectiveness of Urdu-specific augmentations and the potential of tailored data techniques to bridge resource disparities (Chapter 4, Section 4.5.3).

RQ4.1: How can we develop and implement a novel cross-lingual augmentation methodology that leverages English WordNet to enhance Italian semantic datasets?

To enhance Italian semantic datasets, a novel cross-lingual augmentation methodology was implemented using English WordNet. This approach involved leveraging lexical similarities and shared semantic structures between English and Italian to create contextually and semantically appropriate augmentations. By employing English-based resources, Italian data was enriched with semantically meaningful lexical items, maintain-

ing coherence with Italian linguistic structures. This process, discussed in Chapter 4, Section 4.3, represents a significant advancement in using high-resource languages to augment mid-resource languages like Italian.

RQ4.2: What is the effectiveness of this augmentation technique in improving performance scores for both DRS parsing and generation tasks in Italian?

The application of cross-lingual augmentation led to substantial improvements in performance scores for DRS parsing and generation tasks in Italian. Experiments demonstrated that models trained with augmented datasets achieved higher scores across SMATCH, BLEU, METEOR, and COMET metrics compared to non-augmented baselines. These enhancements underscore the capability of cross-lingual data transfer to improve semantic model performance in Italian, as detailed in Chapter 4, Section 4.5.2.

RQ4.3: How does cross-lingual augmentation affect the handling of Italian-specific linguistic features in semantic processing?

The cross-lingual augmentation strategy preserved essential Italian-specific linguistic features, such as flexible word order, rich verbal morphology, and grammatical gender, by carefully adapting English resources to align with Italian grammar and syntax. This methodology ensured that the augmented data did not compromise the linguistic uniqueness of Italian while benefiting from enhanced semantic content. The integration of these language-specific adjustments is outlined in Chapter 4, Section 4.3.

RQ4.4: To what extent is this approach scalable and applicable to other low-resource languages in the domain of semantic NLP?

The cross-lingual augmentation method, while tailored for Italian, was designed to be adaptable to other low-resource languages. By focusing on semantic consistency and linguistic compatibility, the methodology demonstrates potential for broader application in languages that share typological characteristics or Indo-European roots. This scalability suggests that similar cross-lingual strategies could address data scarcity in other low-resource NLP contexts, as presented in Chapter 4.

RQ5.1: How can we effectively evaluate semantic parsing and text generation beyond current structural and surface-level metrics?

The research introduces alternative evaluation frameworks i.e., PARS/PARS-GEN and GEN/GEN-PARS, that incorporate both structural and linguistic quality assessments. Traditional metrics like SMATCH, BLEU, and METEOR often fail to capture the full depth of semantic nuances, leading to a limited view of model performance. By converting parsed DRS representations back to natural language (PARS-GEN) and then parsing generated text into DRS (GEN-PARS), these methods offer a dual perspective that ensures both structural and linguistic integrity, capturing semantic fidelity more comprehensively. Chapter 5, Section 5.3 provides detailed insights into this dual approach.

RQ5.2: How does structural accuracy in semantic parsing influence linguistic quality in text generation?

The research emphasizes that structural accuracy in semantic parsing plays a critical role in determining the linguistic quality of generated text. High SMATCH F1-scores in PARS evaluations correlate strongly with superior generation metrics, as evidenced by BLEU, METEOR, and COMET scores. This relationship underscores the importance of preserving DRS structures to achieve linguistically coherent output, as discussed in Chapter 5, Section 5.4.

RQ5.3: How do evaluation challenges and error patterns vary across different languages?

The study reveals that evaluation challenges and error patterns vary significantly across languages, influenced by structural complexity and morphological features. English, with relatively simple morphology, exhibits stable parsing and generation metrics, while Italian and Urdu encounter issues due to complex syntax and morphology. For example, Italian faces difficulties in role assignments, whereas Urdu struggles with maintaining syntactic consistency and word order. These language-specific insights, discussed in Chapter 5, Section 5.4, highlight the need for customized evaluation approaches.

RQ5.4: Can the reversible nature of semantic parsing and text generation be exploited for improved evaluations?

The reversible nature of semantic parsing and text generation can indeed be exploited for improved evaluations. By utilizing both PARS-GEN and GEN-PARS methodologies, researchers can leverage the inherent reversibility to cross-validate results between parsing and generation processes. This approach not only enhances the robustness of evaluations but also allows for a more nuanced understanding of how well the systems perform across different tasks. The correlation analysis between these two evaluation paradigms further underscores their effectiveness in capturing both structural and linguistic fidelity, thereby improving overall assessment strategies in semantic processing (see Chapter 5, Section 5.4.1).

RQ6.1: How does the reversible nature of semantic parsing and text generation with DRS affect error propagation and correction across different languages?

The reversible nature of semantic parsing and text generation through DRS affects error propagation significantly, with a clear trend toward error amplification rather than correction, particularly in languages with complex morphology and syntax such as Urdu and Italian. While English maintains relative stability with minor error propagation, Urdu and Italian exhibit substantial error magnification. This disparity is attributed to each language's structural complexity, which causes initial parsing or generation errors to compound through subsequent pipeline stages, as observed in the PGP and GPG pipelines (Chapter 6, Section 6.3).

RQ6.2: Can language models be effectively utilized in a pipeline approach to investigate error dynamics without additional model training?

The study effectively utilized pre-trained byT5 models within the PGP and GPG pipelines to analyze error dynamics without further training. These pipelines demonstrate that pre-trained language models can reveal error patterns and propagation in multilingual semantic tasks. Results indicate that while pre-trained models are beneficial for maintaining performance in high-resource languages, their performance deteriorates in low-resource languages like Urdu, which emphasizes the importance of model adaptability across languages (Chapter 6, Section 6.2)

RQ6.3: What are the performance changes achieved by the proposed reversible pipelines compared to baseline models across different languages?

The reversible pipelines (PGP and GPG) introduce varying performance changes across languages, often resulting in a decline compared to baseline models. English shows minimal performance reduction (e.g., a 0.5% drop in SMATCH F1), while Italian and Urdu experience more pronounced declines in both parsing and generation tasks. Specifically, Urdu's BLEU score drops from 55.31 to 48.72 in the GPG pipeline, reflecting a significant challenge for morphologically rich languages (Chapter 6, Table 6.1).

RQ6.4: Which types of errors are more effectively addressed or amplified by the PGP and GPG pipelines in each language?

The PGP and GPG pipelines have limited success in addressing errors and tend to amplify them, especially in low-resource languages. Common error types, such as thematic role misassignments and semantic drift, persist through the pipeline stages. For instance, thematic role assignment errors, particularly in Urdu, often remain uncorrected, leading to further inaccuracies in logical structures. This tendency underscores the challenge of handling complex syntactic features in low-resource settings (Chapter 6, Section 6.5).

RQ6.5: What are the capabilities and limitations of the reversible pipeline approaches in different linguistic contexts?

The reversible pipeline approaches exhibit both strengths and limitations in handling linguistic diversity. While effective for high-resource languages with simpler morphological structures, they struggle with languages that have rich morphological and syntactic complexity. The findings suggest that while the pipelines provide insights into error dynamics, their utility is constrained in practical applications due to error amplification, particularly in Urdu and Italian. This research emphasizes the need for more robust, language-specific models to maintain semantic fidelity without depending on multiple pipeline stages (Chapter 6, Section 6.7).

Challenges and Future Works

This dissertation contributes valuable insights into DRS semantic parsing and text generation through advancements in data transformations i.e., augmentation and delexicalization, task reversibility, and alternate evaluation methodologies. Nevertheless, some limitations identified across different sections suggest fruitful directions for future research. Here are key areas for continued exploration and development:

1. Expansion of Semantic Resources for Non-English Languages: A core limitation in data augmentation stemmed from the reliance on English WordNet-based supersenses, which constrained lexical transformations in low-resource languages such as Italian and Urdu. Future work should focus on expanding semantic resources, including supersense taxonomies, for these and other non-English languages. Building comprehensive lexical resources for under-resourced languages will enhance cross-linguistic consistency and allow for more effective DRS-based transformation techniques.

2. Development of DRS-Specific Pretrained Models: The reliance on general pretrained models highlighted certain performance gaps in semantic parsing and generation tasks, especially for DRS-specific applications. Future studies could prioritize developing and training models directly on DRS-text pairs, which may better capture the nuanced relationships required for accurate semantic parsing and text generation. Such models, tailored to DRS-based tasks, could significantly improve performance, particularly in multilingual and low-resource settings.

3. Enhancement of Error Mitigation in Pipeline Approaches: The reversible pipeline approaches explored in this thesis—Parse-Generate-Parse (PGP) and Generate-Parse-Generate (GPG)—showed more instances of error propagation than mitigation, particularly for morphologically rich languages. Further research could refine these pipeline configurations, perhaps by incorporating adaptive error-detection mechanisms or intermediate correction stages, to better control error dynamics across languages. Investigating language-specific pipeline adjustments could also enhance stability and reliability in low-resource languages like Urdu and Italian.

4. Balanced Multilingual DRS Datasets: Our findings indicate a need for more balanced datasets across different languages to ensure fair and accurate evaluations in DRS processing. Developing extensive multilingual datasets, especially for underrepresented languages, would improve the robustness and applicability of DRS-based language models in broad contexts. These datasets should encompass diverse linguistic phenomena and be rigorously evaluated to support the cross-linguistic adaptability of semantic processing systems.

5. Advancement of Evaluation Metrics Beyond Structural Overlap: Traditional metrics like BLEU, METEOR, and SMATCH fall short in fully capturing semantic quality, often focusing narrowly on structural or lexical overlaps. Through our preliminary investigation, we explored one way to evaluate both structural and linguistic quality. But future research can aim to create advanced evaluation metrics through neural-based measures or graph-based analysis. Such metrics would enable a more comprehensive assessment of DRS transformations, reflecting nuanced meaning preservation across

various contexts.

Bibliography

O. Abend and A. Rappoport. Universal conceptual cognitive annotation (ucca). In *Proceedings of the* 51*st Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 228–238, 2013.

L. Abzianidze, J. Bjerva, K. Evang, H. Haagsma, R. van Noord, P. Ludmann, D.-D. Nguyen, and J. Bos. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In M. Lapata, P. Blunsom, and A. Koller, editors, *Proceedings of the* 15*th Conference of the European Chapter of the Association for Computational Linguistics*: *Volume* 2, *Short Papers*, pages 242–247, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL `https://aclanthology.org/E17-2039`.

L. Abzianidze, R. van Noord, C. Wang, and J. Bos. The parallel meaning bank: A framework for semantically annotating multiple languages. *Applied mathematics and informatics*, 25(2):45–60, 2020. ISSN 1512-0074.

T. Ahmed and A. Hautli. Developing a basic lexical resource for urdu using hindi wordnet. *Proceedings of CLT*10, *Islamabad*, *Pakistan*, 2010.

S. Amari. A theory of adaptive pattern classifiers. *IEEE Transactions on Electronic Computers*, 3:299–307, 1967.

M. S. Amin, L. Anselma, and A. Mazzei. The role of activation function in neural ner for a large semantically annotated corpus. In 2022 *International Conference on Emerging Trends in Electrical*, *Control*, *and Telecommunication Engineering* (*ETECTE*), pages 1–6, 2022a. doi: 10.1109/ETECTE55893.2022.10007317.

M. S. Amin, A. Mazzei, and L. Anselma. Towards data augmentation for drs-to-text generation. In D. Nozza, L. C. Passaro, and M. Polignano, editors, *Proceedings of the Sixth Workshop on Natural Language for Artificial Intelligence* (*NL4AI* 2022) *co-located with* 21*th International Conference of the Italian Association for Artificial Intelligence* (*AI*IA* 2022), *Udine*, *November* 30*th*, 2022, volume 3287 of *CEUR Workshop Proceedings*, pages 141–152. CEUR-WS.org, 2022b. URL `https://ceur-ws.org/Vol-3287/paper14.pdf`.

M. S. Amin, S. T. H. Rizvi, A. Mazzei, and L. Anselma. Assistive data glove for isolated static postures recognition in american sign language using neural network. *Electronics*,

12(8), 2023. ISSN 2079-9292. doi: 10.3390/electronics12081904. URL https://www.mdpi.com/2079-9292/12/8/1904.

M. S. Amin, L. Anselma, and A. Mazzei. Exploring data augmentation in neural DRS-to-text generation. In Y. Graham and M. Purver, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2164–2178, St. Julian's, Malta, Mar. 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.eacl-long.132.

D. Anderson and G. McNeill. Artificial neural networks technology. *Kaman Sciences Corporation*, 258(6):1–83, 1992.

N. Asher. *Reference to abstract objects in discourse*, volume 50. Springer Science & Business Media, 2012.

N. Asher and A. Lascarides. *Logics of conversation*. Cambridge University Press, 2003.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. URL https://api.semanticscholar.org/CorpusID:11212020.

X. Bai, Y. Chen, and Y. Zhang. Graph pre-training for AMR parsing and generation. In S. Muresan, P. Nakov, and A. Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6001–6015, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.415. URL https://aclanthology.org/2022.acl-long.415.

L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, and N. Schneider. Abstract meaning representation for sembanking. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186, 2013.

S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

V. Basile. From logic to language: Natural language generation from logical forms. *PhD thesis, University of Groningen*, 2015.

V. Basile and J. Bos. Towards generating text from discourse representation structures. In *ENLG'11 Proceedings of the 13th European Workshop on Natural Language Generation*, pages 145–150, 2011.

S. Basodi, C. Ji, H. Zhang, and Y. Pan. Gradient amplification: An efficient way to train deep neural networks. *Big Data Mining and Analytics*, 3(3):196–207, 2020.

G. Bebis and M. Georgiopoulos. Feed-forward neural networks. *Ieee Potentials*, 13(4): 27–31, 1994.

D. Beck, G. Haffari, and T. Cohn. Graph-to-sequence learning using gated graph neural networks. In I. Gurevych and Y. Miyao, editors, *Proceedings of the* 56*th Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 273–283, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1026. URL https://aclanthology.org/P18-1026.

J. Belouadi and S. Eger. ByGPT5: End-to-end style-conditioned poetry generation with token-free language models. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the* 61*st Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 7364–7381, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.406. URL https://aclanthology.org/2023.acl-long.406.

M. Bevilacqua, R. Blloshmi, and R. Navigli. One spring to rule them both: Symmetric amr semantic parsing and generation without a complex pipeline. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12564–12573, 2021.

R. Blloshmi, R. Tripodi, and R. Navigli. XL-AMR: Enabling cross-lingual AMR parsing with transfer learning techniques. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the* 2020 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 2487–2500, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.195. URL https://aclanthology.org/2020.emnlp-main.195.

T. Bögel, M. Butt, A. Hautli, and S. Sulger. *Developing a finite-state morphological analyzer for Urdu and Hindi*. Universität Potsdam, 2008.

J. Bos. Wide-coverage semantic analysis with Boxer. In *Semantics in Text Processing. STEP* 2008 *Conference Proceedings*, pages 277–286. College Publications, 2008. URL https://aclanthology.org/W08-2222.

J. Bos. Quantification annotation in discourse representation theory. In *ISA* 2021-17*th Workshop on Interoperable Semantic Annotation, Groningen/Virtuel, Netherlands, June*, pages 1–29, 2021.

J. Bos. The sequence notation: Catching complex meanings in simple graphs. In *Proceedings of the* 15*th International Conference on Computational Semantics* (*IWCS* 2023), pages 1–14, Nancy, France, 2023.

J. Bos, S. Clark, M. Steedman, J. R. Curran, and J. Hockenmaier. Wide-coverage semantic representations from a CCG parser. In *COLING* 2004: *Proceedings of the* 20*th International Conference on Computational Linguistics*, pages 1240–1246, Geneva, Switzerland, aug 23–aug 27 2004. COLING. URL https://aclanthology.org/C04-1180.

J. Bos, V. Basile, K. Evang, N. Venhuizen, and J. Bjerva. The groningen meaning bank. In N. Ide and J. Pustejovsky, editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer, 2017.

L. Bottou. Online algorithms and stochastic approximations. *Online learning in neural networks*, 1998.

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. *Advances in neural information processing systems*, 20, 2007.

L. Bottou et al. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nımes*, 91(8):12, 1991.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020. URL http://arxiv.org/abs/2005.14165.

M. Butt and T. H. King. Urdu and the parallel grammar project. In *COLING-02: The 3rd Workshop on Asian Language Resources and International Standardization*, 2002.

D. Cai and W. Lam. AMR parsing via graph-sequence iterative inference. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1290–1301, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.119. URL https://aclanthology.org/2020.acl-main.119.

S. Cai and K. Knight. Smatch: an evaluation metric for semantic feature structures. In H. Schuetze, P. Fung, and M. Poesio, editors, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (*Volume 2: Short Papers*), pages 748–752, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL https://aclanthology.org/P13-2131.

D. Caldo, S. Bologna, L. Conte, M. S. Amin, L. Anselma, V. Basile, M. M. Hossain, A. Mazzei, P. Heritier, R. Ferracini, E. Kon, and G. De Nunzio. Machine learning algorithms distinguish discrete digital emotional fingerprints for web pages related to back pain. *Scientific Reports*, 13(1):4654, Mar 2023. doi: 10.1038/s41598-023-31741-2.

W. Cao, X. Wang, Z. Ming, and J. Gao. A review on neural networks with random weights. *Neurocomputing*, 275:278–287, 2018.

T. Castro Ferreira, C. Gardent, N. Ilinykh, C. van der Lee, S. Mille, D. Moussallem, and A. Shimorina. The 2020 bilingual, bi-directional WebNLG+ shared task: Overview and evaluation results (WebNLG+ 2020). In T. Castro Ferreira, C. Gardent, N. Ilinykh, C. van der Lee, S. Mille, D. Moussallem, and A. Shimorina, editors, *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web* (*WebNLG+*), pages 55–76, Dublin, Ireland (Virtual), 12 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.webnlg-1.7.

K. Cho. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv*:1406.1078, 2014.

M. Ciaramita and M. Johnson. Supersense tagging of unknown nouns in wordnet. In *Proc*, pages 168–175. 2003 Conference on Empirical Methods in Natural Language Processing, 2003.

A. Copestake, D. Flickinger, C. Pollard, and I. A. Sag. Minimal recursion semantics: An introduction. *Research on language and computation*, 3:281–332, 2005.

S. Cucerzan. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of the* 2007 *joint conference on empirical methods in natural language processing and computational natural language learning* (*EMNLP-CoNLL*), pages 708–716, 2007.

L. Cui, Y. Wu, J. Liu, S. Yang, and Y. Zhang. Template-based named entity recognition using BART. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Findings of the Association for Computational Linguistics*: *ACL-IJCNLP* 2021, pages 1835–1845, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.finding s-acl.161. URL https://aclanthology.org/2021.findings-acl.161.

A. M. Dai and Q. V. Le. Semi-supervised sequence learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/7137debd45ae4d0ab9aa953017286b20-Paper.pdf.

M. Damonte and S. B. Cohen. Structural neural encoders for AMR-to-text generation. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the* 2019 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, *Volume* 1 (*Long and Short Papers*), pages 3649–3658, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1366. URL https://aclanthology.org/N19-1366.

N. G. De Bruijn. Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the church-rosser theorem. In *Indagationes mathematicae* (*proceedings*), volume 75, pages 381–392. Elsevier, 1972.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the* 2019 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, *Volume* 1 (*Long and Short Papers*), pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

A. Di Fabio, S. Conia, and R. Navigli. Verbatlas: a novel large-scale verbal semantic resource and its application to semantic role labeling. In *Proceedings of the* 2019 *Conference on Empirical Methods in Natural Language Processing and the* 9*th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), pages 627–637, 2019.

M. S. Divate. Sentiment analysis of marathi news using lstm. *International journal of Information technology*, 13(5):2069–2074, 2021.

H. Dong, J. Zhang, D. McIlwraith, and Y. Guo. I2t2i: Learning text to image synthesis with textual data augmentation. In 2017 *IEEE international conference on image processing* (*ICIP*), pages 2015–2019. IEEE, 2017.

R. Dror, G. Baumer, S. Shlomov, and R. Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. *in Proc.*, 56:1383–1392, 2018. doi: 10.18653/v1/P18-1128.

O. Dušek, J. Novikova, and V. Rieser. Findings of the E2E NLG challenge. In E. Krahmer, A. Gatt, and M. Goudbeek, editors, *Proceedings of the* 11*th International Conference on Natural Language Generation*, pages 322–328, Tilburg University, The Netherlands, Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6539. URL https://aclanthology.org/W18-6539.

O. Dušek, D. M. Howcroft, and V. Rieser. Semantic noise matters for neural natural language generation. In K. van Deemter, C. Lin, and H. Takamura, editors, *Proceedings of the* 12*th International Conference on Natural Language Generation*, pages 421–426, Tokyo, Japan, Oct.–Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-8652. URL https://aclanthology.org/W19-8652.

J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

K. Evang, V. Basile, G. Chrupała, and J. Bos. Elephant: Sequence labeling for word and sentence segmentation. In *Proceedings of the EMNLP* 2013, 2013.

A. Fan and C. Gardent. Multilingual AMR-to-text generation. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the* 2020 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 2889–2901, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.231. URL https://aclanthology.org/2020.emnlp-main.231.

F. Fancellu, S. Gilroy, A. Lopez, and M. Lapata. Semantic graph parsing with recurrent neural network DAG grammars. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the* 2019 *Conference on Empirical Methods in Natural Language Processing and the* 9*th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), pages 2769–2778, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1278. URL https://aclanthology.org/D19-1278.

C. Fellbaum. *WordNet*: *An electronic lexical database*. MIT press, 1998.

S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics*: *ACL-IJCNLP* 2021, pages 968–988, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.84. URL https://aclanthology.org/2021.findings-acl.84.

R. Fernandez Astudillo, M. Ballesteros, T. Naseem, A. Blodgett, and R. Florian. Transition-based parsing with stack-transformers. In T. Cohn, Y. He, and Y. Liu,

editors, *Findings of the Association for Computational Linguistics*: *EMNLP* 2020, pages 1001–1007, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.89. URL https://aclanthology.org/2020.findings-emnlp.89.

J. Flanigan, C. Dyer, N. A. Smith, and J. G. Carbonell. Generation from abstract meaning representation using tree transducers. In *Proceedings of the* 2016 *conference of the north american chapter of the association for computational linguistics*: *Human language technologies*, pages 731–739, 2016.

Q. Fu, Y. Zhang, J. Liu, and M. Zhang. DRTS parsing with structure-aware encoding and decoding. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the* 58*th Annual Meeting of the Association for Computational Linguistics*, pages 6818–6828, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.609. URL https://aclanthology.org/2020.acl-main.609.

M. Hanna and O. Bojar. A fine-grained analysis of BERTScore. In L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussa, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. J. Yepes, P. Koehn, T. Kocmi, A. Martins, M. Morishita, and C. Monz, editors, *Proceedings of the Sixth Conference on Machine Translation*, pages 507–517, Online, Nov. 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.wmt-1.59.

S. Hao, D.-H. Lee, and D. Zhao. Sequence to sequence learning with attention mechanism for short-term passenger flow prediction in large-scale metro system. *Transportation Research Part C*: *Emerging Technologies*, 107:287–300, 2019.

A. Hautli and M. Butt. Towards a computational semantic analyzer for urdu. In *Proceedings of the* 9*th Workshop on Asian Language Resources*, pages 71–78, 2011.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

J. Hoffman, M. Mohri, and N. Zhang. Algorithms and theory for multiple-source adaptation. *Advances in neural information processing systems*, 31, 2018.

J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

M. Horvat, A. Copestake, and B. Byrne. Hierarchical statistical semantic realization for minimal recursion semantics. In *Proceedings of the* 11*th International Conference on Computational Semantics*, pages 107–117, 2015.

Y. Hou, Y. Liu, W. Che, and T. Liu. Sequence-to-sequence data augmentation for dialogue language understanding. arxiv. preprint, 2018.

J. Howard and S. Ruder. Universal language model fine-tuning for text classification. In I. Gurevych and Y. Miyao, editors, *Proceedings of the* 56*th Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1031. URL https://aclanthology.org/P18-1031.

K. Imamura and E. Sumita. Nict self-training approach to neural machine translation at nmt-2018. *in Proc.*, 2:110–115, 2018. doi: 10.18653/v1/W18-2713].

S. Jadon. A survey of loss functions for semantic segmentation. In 2020 *IEEE conference on computational intelligence in bioinformatics and computational biology* (*CIBCB*), pages 1–7. IEEE, 2020.

M. R. Jafar and M. R. Jafar. The challenges toward the implications of official urdu: Challenges toward the implication of official urdu language. *Pacific International Journal*, 5(4):33–37, Dec. 2022. doi: 10.55014/pij.v5i4.234. URL https://rclss.com/pij/article/view/234.

A. K. Jain, J. Mao, and K. M. Mohiuddin. Artificial neural networks: A tutorial. *Computer*, 29(3):31–44, 1996.

K. M. Jaszczolt and K. Jaszczolt. *Semantics, pragmatics, philosophy: A journey through meaning*. Cambridge University Press, 2023.

K. Jones, S. Strassel, K. Walker, D. Graff, and J. Wright. Multi-language speech collection for NIST LRE. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (*LREC*'16), pages 4253–4258, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL https://aclanthology.org/L16-1674.

M. Junczys-Dowmunt, K. Heafield, H. Hoang, R. Grundkiewicz, and A. Aue. Marian: Cost-effective high-quality neural machine translation in c++. *Arxiv. /abs/*, 1805: 12096, 2018.

H. Kamp. A theory of truth and semantic representation, 277-322, jag groenendijk, tmv janssen and mbj stokhof, eds. In J. A. G. Groenendijk, editor, *Formal methods in the study of language*. U of Amsterdam, 1981.

H. Kamp and U. Reyle. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Publishers, Dordrecht, 1993.

H. Kamp, J. Van Genabith, and U. Reyle. Discourse representation theory. In *Handbook of Philosophical Logic: Volume* 15, pages 125–394. Springer, 2010.

L. Kang, P. Riba, M. Rusiñol, A. Fornés, and M. Villegas. Pay attention to what you read: non-recurrent handwritten text-line recognition. *Pattern Recognition*, 129:108766, 2022.

P. Kapanipathi, I. Abdelaziz, S. Ravishankar, S. Roukos, A. Gray, R. Astudillo, M. Chang, C. Cornelio, S. Dana, A. Fokoue, et al. Leveraging abstract meaning representation for knowledge base question answering. *arXiv preprint arXiv*:2012.01707, 2020.

K. Kipper, A. Korhonen, N. Ryant, and M. Palmer. Extending verbnet with novel verb classes. In *LREC*, pages 1027–1032. Genoa, 2006.

I. Konstas, S. Iyer, M. Yatskar, Y. Choi, and L. Zettlemoyer. Neural AMR: Sequence-to-sequence models for parsing and generation. In R. Barzilay and M.-Y. Kan, editors, *Proceedings of the* 55*th Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 146–157, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1014. URL https://aclanthology.org/P17-1014.

H. Lai, A. Toral, and M. Nissim. Thank you BART! rewarding pre-trained models improves formality style transfer. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the* 59*th Annual Meeting of the Association for Computational Linguistics and the* 11*th International Joint Conference on Natural Language Processing* (*Volume* 2: *Short Papers*), pages 484–494, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-short.62. URL https://aclanthology.org/2021.acl-short.62.

M. Lewis and M. Steedman. A* CCG parsing with a supertag-factored model. In A. Moschitti, B. Pang, and W. Daelemans, editors, *Proceedings of the* 2014 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 990–1000, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1107. URL https://aclanthology.org/D14-1107.

M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the* 58*th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

B. Li, Y. Wen, W. Qu, L. Bu, and N. Xue. Annotating the little prince with chinese amrs. In *Proceedings of the* 10*th Linguistic Annotation Workshop held in Conjunction with ACL* 2016 (*LAW-X* 2016), pages 7–15, 2016.

C. Li and J. Flanigan. Improving neural machine translation with the Abstract Meaning Representation by combining graph and sequence transformers. In L. Wu, B. Liu, R. Mihalcea, J. Pei, Y. Zhang, and Y. Li, editors, *Proceedings of the* 2*nd Workshop on Deep Learning on Graphs for Natural Language Processing* (*DLG4NLP* 2022), pages 12–21, Seattle, Washington, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.dlg4nlp-1.2. URL https://aclanthology.org/2022.dlg4nlp-1.2.

C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

T. Lin, Y. Wang, X. Liu, and X. Qiu. A survey of transformers. *AI open*, 3:111–132, 2022.

F. Liu, J. Flanigan, S. Thomson, N. Sadeh, and N. A. Smith. Toward abstractive summarization using semantic representations. In R. Mihalcea, J. Chai, and A. Sarkar, editors, *Proceedings of the* 2015 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, pages 1077–1086, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1114. URL https://aclanthology.org/N15-1114.

F. Liu, Q. Liu, S. Bannur, F. Pérez-García, N. Usuyama, S. Zhang, T. Naumann, A. Nori, H. Poon, J. Alvarez-Valle, et al. Compositional zero-shot domain transfer with text-to-text models. *Transactions of the Association for Computational Linguistics*, 11:1097–1113, 2023.

G. Liu and J. Guo. Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337:325–338, 2019.

J. Liu, S. B. Cohen, and M. Lapata. Discourse representation structure parsing. In *Proceedings of the* 56*th Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 429–439, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1040. URL https://aclanthology.org/P18-1040.

J. Liu, S. B. Cohen, and M. Lapata. Discourse representation parsing for sentences and documents. In *Proceedings of the* 57*th Annual Meeting of the Association for Computational Linguistics*, pages 6248–6262, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1629. URL https://aclanthology.org/P19-1629.

J. Liu, S. B. Cohen, and M. Lapata. Text generation from discourse representation structures. In *Proceedings of the* 2021 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, pages 397–415, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.35. URL https://aclanthology.org/2021.naacl-main.35.

Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020.

A. C. M. Lorenzo, M. Maru, and R. Navigli. Fully-semantic parsing and generation: The babelnet meaning representation. In *Proceedings of the* 60*th Annual Meeting of the Association for Computational Linguistics* (*Volume* 1: *Long Papers*), pages 1727–1741, 2022.

T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In L. Màrquez, C. Callison-Burch, and J. Su, editors, *Proceedings of the* 2015 *Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1166. URL https://aclanthology.org/D15-1166.

A. Marasović and A. Frank. SRL4ORL: Improving opinion role labeling using multi-task learning with semantic role labeling. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the* 2018 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, *Volume* 1 (*Long Papers*), pages 583–594, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1054. URL https://aclanthology.org/N18-1054.

B. McCann, J. Bradbury, C. Xiong, and R. Socher. Learned in translation: Contextualized word vectors. *Advances in neural information processing systems*, 30, 2017.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Neural Information Processing Systems*, 2013. URL https://api.semanticscholar.org/CorpusID:16447573.

S. Narayan and C. Gardent. Hybrid simplification using deep semantics and machine translation. In *Annual Meeting of the Association for Computational Linguistics*, 2014. URL https://api.semanticscholar.org/CorpusID:15489071.

R. Navigli and S. P. Ponzetto. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the* 48*th annual meeting of the association for computational linguistics*, pages 216–225, 2010.

J. Niehren and S. Thater. Bridging the gap between underspecification formalisms: Minimal Recursion Semantics as dominance constraints. In *Proceedings of the* 41*st Annual Meeting of the Association for Computational Linguistics*, pages 367–374, Sapporo, Japan, July 2003. Association for Computational Linguistics. doi: 10.3115/1075096.1075143. URL https://aclanthology.org/P03-1047.

R. v. Noord. *Neural boxer at the IWCS shared task on DRS parsing*. in Proc. IWCS Shared Task on Semantic Parsing, Gothenburg, Sweden. Association for Computational Linguistics[, 2019. doi: 10.18653/v1/W19-1204].

G. Oliveira dos Santos, E. L. Colombini, and S. Avila. CIDEr-R: Robust consensus-based image description evaluation. In W. Xu, A. Ritter, T. Baldwin, and A. Rahimi, editors, *Proceedings of the Seventh Workshop on Noisy User-generated Text* (*W-NUT* 2021), pages 351–360, Online, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.wnut-1.39. URL https://aclanthology.org/2021.wnut-1.39.

OpenAI. Gpt-4 technical report, 2023.

J. Opitz. SMATCH++: Standardized and extended evaluation of semantic graphs. In A. Vlachos and I. Augenstein, editors, *Findings of the Association for Computational Linguistics*: *EACL* 2023, pages 1595–1607, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.118. URL https://aclanthology.org/2023.findings-eacl.118.

T. O'Gorman, M. Regan, K. Griffitt, U. Hermjakob, K. Knight, and M. Palmer. Amr beyond the sentence: the multi-sentence amr corpus. In *Proceedings of the 27th international conference on computational linguistics*, pages 3693–3702, 2018.

M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106, 2005.

X. Pan, T. Cassidy, U. Hermjakob, H. Ji, and K. Knight. Unsupervised entity linking with abstract meaning representation. In *Proceedings of the 2015 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 1130–1139, 2015.

J. D. Paola and R. A. Schowengerdt. A review and analysis of backpropagation neural networks for classification of remotely-sensed multi-spectral imagery. *International Journal of remote sensing*, 16(16):3033–3058, 1995.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In A. Moschitti, B. Pang, and W. Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https://aclanthology.org/D14-1162.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202.

W. Poelman, R. van Noord, and J. Bos. Transparent semantic parsing with universal dependencies using graph transformations. In *29th International Conference on Computational Linguistics*, pages 4186–4192. Association for Computational Linguistics (ACL), 2022.

M. Popović. chrF: character n-gram F-score for automatic MT evaluation. In O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, C. Hokamp, M. Huck, V. Logacheva, and P. Pecina, editors, *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-3049. URL https://aclanthology.org/W15-3049.

M. Post and D. Vilar. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational*

Linguistics: *Human Language Technologies*, *Volume* 1 (*Long Papers*), pages 1314–1324, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1119. URL https://aclanthology.org/N18-1119.

L. Procopio, R. Tripodi, and R. Navigli. Sgl: Speaking the graph languages of semantic parsing via multilingual translation. In *Proceedings of the* 2021 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, pages 325–337, 2021.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training, 2018. URL https://www.mikecaptain.com/resources/pdf/GPT-1.pdf.

S. Rao, D. Marcu, K. Knight, and H. Daumé III. Biomedical event extraction using abstract meaning representation. In *BioNLP* 2017, pages 126–135, 2017.

R. Rei, C. Stewart, A. C. Farinha, and A. Lavie. COMET: A neural framework for MT evaluation. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the* 2020 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 2685–2702, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.213. URL https://aclanthology.org/2020.emnlp-main.213.

L. F. R. Ribeiro, Y. Zhang, C. Gardent, and I. Gurevych. Modeling global and local node contexts for text generation from knowledge graphs. *Transactions of the Association for Computational Linguistics*, 8:589–604, 2020. doi: 10.1162/tacl_a_00332. URL https://aclanthology.org/2020.tacl-1.38.

L. F. R. Ribeiro, J. Pfeiffer, Y. Zhang, and I. Gurevych. Smelting gold and silver for improved multilingual AMR-to-Text generation. In M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, editors, *Proceedings of the* 2021 *Conference on Empirical Methods in Natural Language Processing*, pages 742–750, Online and Punta Cana, Dominican Republic, Nov. 2021a. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.57. URL https://aclanthology.org/2021.emnlp-main.57.

L. F. R. Ribeiro, M. Schmitt, H. Schütze, and I. Gurevych. Investigating pretrained language models for graph-to-text generation. In A. Papangelis, P. Budzianowski, B. Liu, E. Nouri, A. Rastogi, and Y.-N. Chen, editors, *Proceedings of the* 3*rd Workshop on Natural Language Processing for Conversational AI*, pages 211–227, Online, Nov. 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.nlp4convai-1.20. URL https://aclanthology.org/2021.nlp4convai-1.20.

H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

G. Sarti and M. Nissim. IT5: Text-to-text pretraining for Italian language understanding and generation. In N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, editors, *Proceedings of the* 2024 *Joint International Conference on Computational Linguistics, Language Resources and Evaluation* (*LREC-COLING* 2024), pages 9422–9433, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.823.

R. Sennrich. How grammatical is character-level neural machine translation? assessing MT quality with contrastive translation pairs. In M. Lapata, P. Blunsom, and A. Koller, editors, *Proceedings of the* 15*th Conference of the European Chapter of the Association for Computational Linguistics*: *Volume* 2, *Short Papers*, pages 376–382, Valencia, Spain, Apr. 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-2060.

S. Sharma, J. He, K. Suleman, H. Schulz, and P. Bachman. Natural language generation in dialogue using lexicalized and delexicalized data. *ArXiv*, abs/1606.03632, 2016. URL https://api.semanticscholar.org/CorpusID:14949670.

K. C. Sheang and H. Saggion. Controllable sentence simplification with a unified text-to-text transfer transformer. In A. Belz, A. Fan, E. Reiter, and Y. Sripada, editors, *Proceedings of the* 14*th International Conference on Natural Language Generation*, pages 341–352, Aberdeen, Scotland, UK, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.inlg-1.38. URL https://aclanthology.org/2021.inlg-1.38.

A. Shimorina and C. Gardent. Handling rare items in data-to-text generation. In *Proceedings of the* 11*th international conference on natural language generation*, pages 360–370, 2018.

C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Z. Shou, Y. Jiang, and F. Lin. AMR-DA: Data augmentation by Abstract Meaning Representation. In S. Muresan, P. Nakov, and A. Villavicencio, editors, *Findings of the Association for Computational Linguistics*: *ACL* 2022, pages 3082–3098, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.244. URL https://aclanthology.org/2022.findings-acl.244.

M. A. Sobrevilla Cabezudo and T. Pardo. Towards a general Abstract Meaning Representation corpus for Brazilian Portuguese. In A. Friedrich, D. Zeyrek, and J. Hoek, editors, *Proceedings of the* 13*th Linguistic Annotation Workshop*, pages 236–244, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4028. URL https://aclanthology.org/W19-4028.

J. Solawetz and S. Larson. Lsoie: A large-scale dataset for supervised open information extraction. *arXiv preprint arXiv*:2101.11177, 2021.

L. Song and D. Gildea. SemBleu: A robust metric for AMR parsing evaluation. In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the* 57*th Annual Meeting*

of the Association for Computational Linguistics, pages 4547–4552, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1446. URL https://aclanthology.org/P19-1446.

L. Song, D. Gildea, Y. Zhang, Z. Wang, and J. Su. Semantic neural machine translation using amr. *Transactions of the Association for Computational Linguistics*, 7:19–31, 2019.

L. Song, A. Wang, J. Su, Y. Zhang, K. Xu, Y. Ge, and D. Yu. Structural information preserving for graph-to-text generation. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7987–7998, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.712. URL https://aclanthology.org/2020.acl-main.712.

L. Stankevičius, M. Lukoševičius, J. Kapočiūtė-Dzikienė, M. Briedienė, and T. Krilavičius. Correcting diacritics and typos with a byt5 transformer model. *Applied Sciences*, 12 (5):2636, 2022.

I. Sutskever. Sequence to sequence learning with neural networks. *arXiv preprint arXiv*:1409.3215, 2014.

M. Turpin, J. Michael, E. Perez, and S. R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting, 2023.

I. J. Unanue, G. Haffari, and M. Piccardi. T3l: Translate-and-test transfer learning for cross-lingual text classification. *Transactions of the Association for Computational Linguistics*, 11:1147–1161, 2023.

M. Vainio, A. S. Suni, T. Raitio, J. Nurminen, J. Järvikivi, and P. Alku. New method for delexicalization and its application to prosodic tagging for text-to-speech synthesis. In *Interspeech* 2010: *Annual Conference of the International Speech Communication Association*, 2009.

C. van der Lee, A. Gatt, E. van Miltenburg, S. Wubben, and E. Krahmer. Best practices for the human evaluation of automatically generated text. In K. van Deemter, C. Lin, and H. Takamura, editors, *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan, Oct.–Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-8643. URL https://aclanthology.org/W19-8643.

R. A. Van der Sandt. Presupposition projection as anaphora resolution. *Journal of semantics*, 9(4):333–377, 1992.

J. E. Van Gysel, M. Vigus, J. Chun, K. Lai, S. Moeller, J. Yao, T. O'Gorman, A. Cowell, W. Croft, C.-R. Huang, et al. Designing a uniform meaning representation for natural language processing. *KI-Künstliche Intelligenz*, 35(3):343–360, 2021.

R. van Noord. Character-based neural semantic parsing. *PhD thesis, University of Groningen*, 2021.

R. van Noord, L. Abzianidze, H. Haagsma, and J. Bos. Evaluating scoped meaning representations. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, and T. Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation* (*LREC* 2018), pages 1–9, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1267.

R. van Noord, A. Toral, and J. Bos. Linguistic information in neural semantic parsing with multiple encoders. In *Proc. 13th International Conference on Computational Semantics-Short Papers*, pages 24–31. Association for Computational Linguistics (ACL), 2019.

R. van Noord, A. Toral, and J. Bos. Character-level representations improve DRS-based semantic parsing even in the age of BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 4587–4603, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.371. URL https://aclanthology.org/2020.emnlp-main.371.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

N. J. Venhuizen. Projection in discourse: A data-driven formal semantic analysis. *PhD thesis, University of Groningen*, 2015.

N. J. Venhuizen, J. Bos, P. Hendriks, and H. Brouwer. Discourse semantics with information structure. *Journal of Semantics*, 35(1):127–169, 2018.

S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

C. Wang, R. van Noord, A. Bisazza, and J. Bos. Input representations for parsing discourse representation structures: Comparing English with Chinese. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 2: Short Papers*), pages 767–775, Online, Aug. 2021a. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-short.97. URL https://aclanthology.org/2021.acl-short.97.

C. Wang, R. van Noord, A. Bisazza, and J. Bos. Evaluating text generation from discourse representation structures. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics* (*GEM* 2021), pages 73–83, 2021b.

C. Wang, H. Lai, M. Nissim, and J. Bos. Pre-trained language-meaning models for multilingual parsing and generation. In *Findings of the Association for Computational*

Linguistics: *ACL* 2023, pages 5586–5600, Toronto, Canada, July 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.345. URL https://aclanthology.org/2023.findings-acl.345.

C. Wang, X. Zhang, and J. Bos. Discourse representation structure parsing for Chinese. In S. Chatzikyriakidis and V. de Paiva, editors, *Proceedings of the* 4*th Natural Logic Meets Machine Learning Workshop*, pages 62–74, Nancy, France, June 2023b. Association for Computational Linguistics. URL https://aclanthology.org/2023.naloma-1.7.

T. Wang, X. Wan, and H. Jin. AMR-to-text generation with graph transformer. *Transactions of the Association for Computational Linguistics*, 8:19–33, 2020. doi: 10.1162/tacl_a_00297. URL https://aclanthology.org/2020.tacl-1.2.

B. Warner and M. Misra. Understanding neural networks as statistical tools. *The american statistician*, 50(4):284–293, 1996.

D. Xu, J. Li, M. Zhu, M. Zhang, and G. Zhou. Improving AMR parsing with sequence-to-sequence pre-training. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the* 2020 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 2501–2511, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.196. URL https://aclanthology.org/2020.emnlp-main.196.

K. Xu, L. Wu, Z. Wang, Y. Feng, and V. Sheinin. SQL-to-text generation with graph-to-sequence model. In E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, editors, *Proceedings of the* 2018 *Conference on Empirical Methods in Natural Language Processing*, pages 931–936, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1112. URL https://aclanthology.org/D18-1112.

L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the* 2021 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, pages 483–498, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.41. URL https://aclanthology.org/2021.naacl-main.41.

L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, and C. Raffel. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022.

N. Xue, O. Bojar, J. Hajič, M. Palmer, Z. Urešová, and X. Zhang. Not an interlingua, but close: Comparison of English AMRs to Chinese and Czech. In N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation* (*LREC*'14), pages 1765–1772, Reykjavik, Iceland, May 2014.

European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/384_Paper.pdf.

S. Yang, W.-T. Xiao, M. Zhang, S. Guo, J. Zhao, and S. Furao. Image data augmentation for deep learning: A survey. *ArXiv*, abs/2204.08610, 2022. URL https://api.semanticscholar.org/CorpusID:248240105.

W.-t. Yih, X. He, and C. Meek. Semantic parsing for single-relation question answering. In *Proceedings of the* 52*nd Annual Meeting of the Association for Computational Linguistics* (*Volume* 2: *Short Papers*), pages 643–648, 2014.

F. Yu and X. Xu. A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved bp neural network. *Applied Energy*, 134:102–113, 2014.

S. Zhang, X. Ma, K. Duh, and B. Van Durme. AMR parsing as sequence-to-graph transduction. In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the* 57*th Annual Meeting of the Association for Computational Linguistics*, pages 80–94, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1009. URL https://aclanthology.org/P19-1009.

T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with BERT. In 8*th International Conference on Learning Representations*, *ICLR* 2020, *Addis Ababa*, *Ethiopia*, *April* 26-30, 2020. OpenReview.net, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.

X. Zhang, C. Wang, R. van Noord, and J. Bos. Gaining more insight into neural semantic parsing with challenging benchmarks. In C. Bonial, J. Bonn, and J. D. Hwang, editors, *Proceedings of the Fifth International Workshop on Designing Meaning Representations @ LREC-COLING* 2024, pages 162–175, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.dmr-1.17.

W. Zhong, J. Xu, D. Tang, Z. Xu, N. Duan, M. Zhou, J. Wang, and J. Yin. Reasoning over semantic-level graph for fact checking. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the* 58*th Annual Meeting of the Association for Computational Linguistics*, pages 6170–6180, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.549. URL https://aclanthology.org/2020.acl-main.549.

G. Zhou and G. Lampouras. Webnlg challenge 2020: Language agnostic delexicalisation for multilingual rdf-to-text generation. In *Proceedings of the* 3*rd International Workshop on Natural Language Generation from the Semantic Web* (*WebNLG+*), pages 186–191, 2020.

J. Zhou, T. Naseem, R. Fernandez Astudillo, and R. Florian. AMR parsing with action-pointer transformer. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the* 2021 *Conference of the North American Chapter of the Association for Computational*

Linguistics: *Human Language Technologies*, pages 5585–5598, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.443. URL https://aclanthology.org/2021.naacl-main.443.

P. Zhou, Z. Qi, S. Zheng, J. Xu, H. Bao, and B. Xu. Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. *arXiv preprint arXiv*:1611.06639, 2016.

List of Acronyms

NLP	Natural Language Processing
DRS	Discourse Representation Structure
DRSs	Discourse Representation Structures
LSTM	Long Short-Term Memory
LLM	Large Language Model
PMB	Parallel Meaning Bank
DRT	Discourse Representation Theory
AMR	Abstract Meaning Representation
MRS	Minimal Recursion Semantics
CCG	Combinatory Categorial Grammar
UMB	Urdu Meaning Bank
SBN	Simplified Box Notation
BMR	BabelNet Meaning Representation
UCCA	Universal Conceptual Cognitive Annotation
ANN	Artificial Neural Networks
UMR	Universal Meaning Representation
FFNN	Feed-Forward Neural Network
seq2seq	sequence-to-sequence
RNNs	Recurrent Neural Networks
BOS	Beginning of Sequence
EOS	End of Sequence
PLMs	Pre-trained Language Models
NLG	Natural Language Generation
RDF	Resource Description Framework
ROSE	Robust Overall Semantic Evaluation
T5	Text-to-Text Transfer Transformer

Appendix – Additional Research Contributions

This section presents summaries of additional research papers published alongside the core work of this thesis. These publications, while not directly part of the main thesis, represent related research efforts that complement and expand upon the central themes explored in the main body of work. The papers summarized here demonstrate the breadth of research interests and collaborations undertaken during the course of this doctoral study.

This section is divided into three further sections, each dedicated to a distinct publication:

1. **Neural Named Entity Recognition in Semantically Annotated Corpora:** This section summarizes the paper titled "The Role of Activation Function in Neural NER for a Large Semantically Annotated Corpus" by (Amin et al., 2022). This study explores the impact of different activation functions on the performance of neural networks in Named Entity Recognition tasks, particularly when working with large semantically annotated corpora.

2. **Assistive Technology for American Sign Language Recognition:** The second section presents a summary of "Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network" by (Amin et al., 2023). This paper describes the development and implementation of a sensor-based glove designed to recognize static gestures in American Sign Language, with potential applications in assistive technology and communication.

3. **Machine Learning in Health Information Analysis:** The final section summarizes "Machine Learning Algorithms Distinguish Discrete Digital Emotional Fingerprints for Web Pages Related to Back Pain" by (Caldo et al., 2023). This is collaborative work with other researchers that focuses on applying machine learning and sentiment analysis techniques to identify emotional patterns in online health-related content, focusing on musculoskeletal conditions such as back pain.

Each of the summaries listed in the sections below provides an overview of the research objectives, methodologies, key findings, and potential implications of the respective studies. While these papers diverge from the main focus of the thesis, they

demonstrate the application of similar computational and analytical techniques to diverse domains, including natural language processing, assistive technologies, and health informatics.

A. The Role of Activation Function in Neural NER for a Large Semantically Annotated Corpus

This paper presents a comprehensive study on the impact of activation functions in neural Named Entity Recognition (NER) systems, a crucial task in natural language processing and computational semantics. The authors implement a Bidirectional Long Short-Term Memory (Bi-LSTM) model with a Conditional Random Field (CRF) layer, a state-of-the-art architecture for sequence labeling tasks, to perform NER on the Groningen Meaning Bank (GMB) corpus. This large semantically annotated dataset, containing 62,010 sentences with multiple layers of linguistic annotation, provides a rich testbed for evaluating NER performance.

The primary focus of this study is on the role of activation functions in neural NER systems. We have systematically evaluated 13 different activation functions, including both classical and modern variants. Our findings reveal that only four activation functions — Sigmoid, Exponential, SoftMax, and SoftPlus — yielded satisfactory results for this task. Notably, the Sigmoid function achieved the highest accuracy at 95.17%, closely followed by Exponential (95.14%), SoftMax (94.76%), and SoftPlus (94.38%). These results underscore the critical role that activation functions play in the performance of neural NER systems and, by extension, in computational semantic tasks.

This implementation utilizes a sophisticated neural architecture, comprising a 4-layer embedding with 64 dimensions, 100 RNN units, and a CRF layer for final prediction. The setup, combined with careful hyperparameter tuning, allows for a nuanced exploration of how different activation functions affect the model's ability to learn and represent semantic information in text. The use of the Stochastic Gradient Descent (SGD) optimizer across all experiments ensures a fair comparison between activation functions.

In the broader context of computational semantics, these experiments offer valuable insights for neural semantic parsing and generation tasks. The superior performance of certain activation functions suggests that they may be more adept at capturing the subtle semantic nuances present in natural language. These findings could potentially generalize to other sequence labeling tasks in semantics, such as semantic role labeling or fine-grained entity typing.

Moreover, the study highlights the importance of hyperparameter op-

timization in neural approaches to semantic tasks. The significant performance variations observed across different activation functions demonstrate that low-level architectural choices can have a substantial impact on the ability of the model to process and generate semantic information. This underscores the need for careful consideration of these elements when designing neural models for semantic parsing and generation. Our experimental findings not only provide practical guidance for improving NER performance but also offer insights that could be leveraged to enhance a wide range of semantic processing tasks. As the field continues to advance, such detailed explorations of neural network components will be crucial in developing more sophisticated and accurate models for understanding and generating natural language semantics.

Reference: M. S. Amin, L. Anselma and A. Mazzei, "The Role of Activation Function in Neural NER for a Large Semantically Annotated Corpus", 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 2022, pp. 1-6, doi: 10.1109/ETECTE55893.2022.10007317. https://ieeexplore.ieee.org/document/10007317

B. Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network

In this paper, we introduce an innovative sensor-equipped glove designed to recognize static gestures in American Sign Language (ASL), with a focus on alphabetic and numeric signs. This assistive device utilizes flex sensors and a gyroscope to capture data on finger bending and hand orientations. The collected information is then analyzed using a fully-connected neural network, which has been trained on a custom dataset. The primary application of the glove is in fingerspelling, a crucial aspect of ASL used for expressing proper nouns and technical terminology.

In the development of the glove, we prioritized simplicity and efficiency, strategically limiting the number of sensors to balance complexity reduction with recognition accuracy. The training process of the neural network employs a scaled conjugate gradient backpropagation algorithm, resulting in high accuracy rates for both alphabetic and numeric gestures. To optimize the performance of the neural model, we experimented with various activation functions, including ReLU, Tanh, and Sigmoid. The findings indicate that the glove demonstrates strong performance, with promising accuracy levels in both training and testing phases for alphabetic and numeric datasets.

The conceptual formalities of data glove share significant commonalities with the field of computational semantics, particularly in relation to neural semantic parsing and generation. Just as neural semantic parsing aims to convert structured data (such as Discourse Representation Structures or logical forms) into natural language text or interpret textual information into structured representations, the assistive glove system transforms hand gestures into structured numeric data for neural network interpretation and sign recognition.

The fundamental concept of translating one form of meaning (in this case, gestures) into another (recognized alphabetic/numeric symbols) mirrors the core principles of semantic parsing and generation. In these processes, structured representations like DRSs are converted into human language or vice versa. The application of neural networks for classification in the glove system is analogous to the use of neural models in semantic tasks, where input signals are interpreted (whether text or gestures) to produce meaningful structured data.

Furthermore, the process of refining neural networks for gesture recog-

nition — through testing different activation functions, analyzing overfitting, and enhancing classification accuracy — parallels the optimization strategies employed in neural semantic parsing to improve models for accurate interpretation of language and meaning structures. This research thus exemplifies how neural approaches can bridge the gap between physical and computational semantics by interpreting sensor-based gesture data and connecting it to linguistic and symbolic meanings.

Reference: Amin, Muhammad Saad, Syed Tahir Hussain Rizvi, Alessandro Mazzei, and Luca Anselma. 2023. "Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network" Electronics 12, no. 8: 1904.
https://doi.org/10.3390/electronics12081904
https://www.mdpi.com/2079-9292/12/8/1904

C. Machine learning algorithms distinguish discrete digital emotional fingerprints for web pages related to back pain

In this paper, we explore the application of machine learning and sentiment analysis to identify emotional patterns in online health content, specifically focusing on web pages discussing musculoskeletal conditions such as back pain and joint disorders. We theorized that successful health-related web pages exhibit consistent emotional patterns that could potentially predict the specific pathology they address. By extracting emotional content (including joy, disgust, surprise, and sadness) from these pages, the study employed supervised machine learning algorithms, primarily Support Vector Machines (SVM) and decision trees, to classify and differentiate between pathologies based on their emotional content. Notably, disgust emerged as a key emotion in distinguishing between different pathologies, suggesting that these emotional fingerprints might influence patients' perceptions and behaviors regarding chronic pain.

This study is grounded in the biopsychosocial (BPS) model of health, which recognizes the impact of emotional and psychological factors on physical health outcomes. We suggest that the emotional content of medical web pages could contribute to a collective consciousness among patients regarding specific conditions, potentially influencing the development and persistence of chronic pain. This interplay between digital information, emotional responses, and health outcomes underscores the importance of understanding how the emotional content in digital health resources affects users.

The fundamental concept of identifying and categorizing emotional patterns in web content aligns closely with the objectives of computational semantics, particularly in the realm of neural semantic parsing and generation. Neural semantic parsing involves transforming unstructured text into structured representations, such as Discourse Representation Structures (DRS), while generation focuses on producing coherent text from these structured meanings. In this study, the emotional content extracted from web pages can be viewed as a semantic structure reflecting the biopsychosocial profile of patients. The use of machine learning to classify emotional fingerprints mirrors the neural models employed in parsing and generating meaning representations from text. The approach to sentiment analysis, which involves identifying discrete emotional components from textual data, parallels the way semantic parsers

extract logical or emotional information from sentences. The machine learning models used in this study to map emotional fingerprints onto health conditions share similarities with how neural semantic parsers map linguistic structures onto their meanings, highlighting the need for precise feature extraction and classification.

Through this study, we demonstrate how computational techniques can bridge linguistic semantics with real-world applications, much like how neural semantic parsing and generation connect structured meaning representations with natural language processing. This connection emphasizes the potential of combining computational semantic models with affective computing to address complex health information systems.

Reference: Caldo, D., Bologna, S., Conte, L. et al. Machine learning algorithms distinguish discrete digital emotional fingerprints for web pages related to back pain. Sci Rep 13, 4654 (2023).
https://doi.org/10.1038/s41598-023-31741-2
https://www.nature.com/articles/s41598-023-31741-2