IRIS AperTO

UNIVERSITÀ DEGLI STUDI DI TORINO

AperTO - Archivio Istituzionale Open Access dell'Università di Torino

Dependency Syntax in the Automatic Detection of Irony and Stance

This is the author's manuscript

Original Citation:

Availability:

This version is available http://hdl.handle.net/2318/1873580 since 2022-09-08T09:13:54Z

Terms of use:

Open Access

Anyone can freely access the full text of works made available as "Open Access". Works made available under a Creative Commons license can be used according to the terms and conditions of said license. Use of all other works requires consent of the right holder (author or publisher) if not exempted from copyright protection by the applicable law.

(Article begins on next page)

06 September 2026

University of Turin

Doctoral School of Sciences and Innovative Technologies PhD
Program in Computer Science
XXXIII cycle

PhD Dissertation

Alessandra Teresa Cignarella

Dependency Syntax in the Automatic Detection of Irony and Stance

Advisors
Cristina Bosco
Università degli Studi di Torino, Italy
Paolo Rosso
Universitat Politècnica de València, Spain

PhD Coordinator
Marco Grangetto

September 2021

Universitat Politècnica de València

Departamento de Sistemas Informáticos y Computación

Tesis de Doctorado en Informática

Alessandra Teresa Cignarella

Dependency Syntax in the Automatic Detection of Irony and Stance

Directores de Tesis
Cristina Bosco
Università degli Studi di Torino, Italy
Paolo Rosso
Universitat Politècnica de València, Spain

Septiembre 2021

To all the people who inspired me and supported me during this amazing and life-changing journey.

To myself.[1]

[1] I started writing the first version of this dissertation during the first health emergency lockdown due to the outbreak of the Covid-19 pandemic in Italy (starting from March 2020). I was able to submit the first draft in April 2021, and I later received the comments of reviewers around July 2021. The final (and current) version of the thesis has been registered in September 2021.

Abstract

The present thesis is part of the broad panorama of studies of Natural Language Processing (NLP). In particular, it is a work of Computational Linguistics (CL) designed to study in depth the contribution of syntax in the field of sentiment analysis and, therefore, to study texts extracted from social media or, more generally, online content.

Furthermore, given the recent interest of the scientific community in the Universal Dependencies (UD) project, which proposes a morphosyntactic annotation format aimed at creating a "universal" representation of the phenomena of morphology and syntax in a manifold of languages, in this work we made use of this format, thinking of a study in a multilingual perspective (Italian, English, French and Spanish). Although the UD format was originally conceived to be applied to prose texts and, therefore, to texts considered more "standard" from the point of view of morphosyntactic norms and punctuation, in more recent years the same scheme has begun to be applied also to user-generated content (UGC), i.e. texts extracted from social media, blogs, forums and microblogging platforms, such as Reddit, Twitter or Wikipedia pages. Inevitably, the application of this annotation framework to such a peculiar textual genre, in which the texts are accompanied by multimedia elements such as links, photos and videos, emojis and non-standardized punctuation, has opened up several problems in the Universal Dependencies community, many of which are still the subject of open and heated debate today.

In this work we will therefore try to provide an exhaustive presentation of the morphosyntactic annotation format of UD, in particular underlining the most relevant issues regarding their application to UGC. Two sub-areas of Sentiment Analysis will be presented, and used as case studies, in order to test the research hypotheses: the first case study will be in the field of automatic Irony Detection and the second in the area of Stance Detection. In both cases, historical notes will be provided that can serve as a context for the reader, an introduction to the problems faced will be outlined and the activities proposed in the computational linguistics community will be described. Furthermore, particular attention will be paid to the resources currently available as well as to those developed specifically for the study of the aforementioned phenomena. Finally, through the description of a series of experiments, both within evaluation campaigns and within independent studies, I will try to describe the contribution that syntax can provide to the resolution of such tasks, as subfields of Sentiment Analysis.

This thesis is a revised collection of my three-year PhD career and collocates within the growing trend of studies devoted to make Artificial Intelligence results more explainable, going beyond the achievement of highest scores in performing tasks, but rather making their motivations understandable and comprehensible for experts in the domain.

The novel contribution of this work mainly consists in the exploitation of features that are based on morphology and dependency syntax, which were used in order to create vectorial representations of social media texts in various languages and for two different tasks. Such features have then been paired with a manifold of machine learning classifiers, with some neural networks and also with the language model BERT.

Results suggest that fine-grained dependency-based syntactic information is highly

informative for the detection of irony, and less informative for what concerns stance detection. Nonetheless, dependency syntax might still prove useful in the task of stance detection if firstly irony detection is considered as a preprocessing step. I also believe that the dependency syntax approach that I propose could shed some light on the explainability of a difficult pragmatic phenomenon such as irony. In fact, the several studies presented here allowed to investigate whether syntactic structures, independently from the target language, may provide information useful to understand whether a message is ironic or not.

Abstract

La presente tesi si colloca nell'ampio panorama degli studi sul trattamento automatico del linguaggio (Natural Language Processing, NLP). In particolare, si tratta di un lavoro di Linguistica Computazionale atto a studiare approfonditamente il contributo della sintassi nel campo dell'analisi del sentimento (Sentiment Analysis) e dunque a studiare testi estratti da social media o, più in generale, contenuti creati in rete.

Inoltre, visto il recente interesse della comunita scientifica verso il progetto delle Universal Dependencies (UD), che propone un formato di annotazione morfosintattica mirato a creare una rappresentazione "universale" dei fenomeni di morfologia e sintassi nelle lingue, nel presente lavoro ci si è avvalsi di questo formato, pensando ad uno studio in una prospettiva multilingue, considerando le lingue conosciute dalla candidata (Italiano, Inglese, Francese e Spagnolo). Sebbene il formato UD fosse stato originariamente concepito per essere applicato a testi di prosa, e dunque, a testi considerati più "standard" da un punto di vista di norme morfosintattiche e di punteggiatura, in anni più recenti lo stesso schema ha iniziato ad essere applicato anche a contenuti generati dagli utenti in rete (user-generated content, UGC), e cioè testi estratti da social media, blog, forum e piattaforme di microblog, come ad esempio Twitter o pagine di Wikipedia. Inevitabilmente, l'applicazione di tale cornice di annotazione ad un genere testuale così peculiare, in cui i testi sono corredati da elementi multimediali quali link, foto e video, emoji e una punteggiatura non normativizzata, ha aperto diverse problematiche nella comunità delle Universal Dependencies, molte delle quali sono, ancora oggi, oggetto di dibattito aperto ed acceso.

In questo lavoro si cercherà dunque, di fornire una esaustiva presentazione del formato di annotazione morfosintattica delle Universal Dependencies, in particolare sottolineando le tematiche più rilevanti riguardo alla loro applicazione a contenuti UGC, estratti da reti sociali. Si presenteranno prevalentemente due sottoaree della Sentiment Analysis, usate come casi di studio, per testare le ipotesi di ricerca: il primo caso di studio sarà nel campo del riconoscimento automatico dell'ironia (Irony Detection) e il secondo nell'area del riconoscimento automatico della *stance*[2] (Stance Detection). In entrambe le casistiche si forniranno dei cenni storici che possano servire da contesto per il lettore, si delineerà una introduzione al problema affrontato e si descriveranno le attività proposte nella comunità di linguistica computazionale sino ad oggi. Inoltre si porrà particolare attenzione alle risorse attualmente disponibili oltre che a quelle sviluppate appositamente per lo studio dei suddetti fenomeni. Infine, attraverso la descrizione di una serie di esperimenti, sia all'interno di campagne di valutazione, sia all'interno di studi autonomi, si cercherà di descrivere l'apporto che la sintassi può fornire alla risoluzione di problematiche nel campo della Sentiment Analysis.

Questa tesi è una raccolta rivisitata dei lavori svolti durante gli ultimi tre anni e mezzo del mio percorso di dottorato di ricerca e si colloca all'interno del filone di studi dedicati a rendere i risultati dell'Intelligenza Artificiale più spiegabili[3], andando oltre

[2] Traducibile come 'atteggiamento' o 'posizione' nei confronti di un target.

[3] AI explainability.

l'ottenimento di prestazioni più elevate nella risoluzione di certi task, ma piuttosto rendendo comprensibili le caratteristiche sottostanti per gli esperti del settore.

Il contributo innovativo di questo lavoro consiste principalmente nell'aver sfruttato informazioni morfologiche e sintattiche, che sono state utilizzate per creare rappresentazioni vettoriali di testi estratti dai social media in varie lingue e per due task diversi. Tali features sono state poi usate all'interno di una varietà di classificatori di machine learning, con alcune reti neurali e anche con il modello BERT.

I risultati suggeriscono che le informazioni sintattiche a grana fine sono altamente informative per il roconoscimento automatico dell'ironia e meno informative per ciò che riguarda il rilevamento della stance. Tuttavia, la sintassi a dipendenze potrebbe ancora rivelarsi utile nel task di stance detection se in primo luogo viene riconosciuta l'ironia, venendo considerata come una prima fase di pre-elaborazione. Credo anche che l'approccio qui proposto, basato principalmente su caratteristiche estratte dalla sintassi a dipendenze, possa far luce sulla spiegabilità e comprension di un fenomeno pragmatico complesso come l'ironia. Infatti, i diversi studi qui presentati hanno permesso di indagare se le strutture sintattiche, indipendentemente dalla lingua di destinazione, possano fornire informazioni utili per capire se un messaggio è ironico o meno.

Resumen

La presente tesis se enmarca dentro del amplio panorama de estudios relacionados con el Procesamiento del Lenguaje Natural (NLP). En concreto, se trata de un trabajo de Lingüística Computacional (CL) cuyo objetivo principal es estudiar en profundidad la contribución de la sintaxis en el campo del análisis de sentimientos y, en concreto, aplicado a estudiar textos extraídos de las redes sociales o, más en general, de contenidos online.

Además, dado el reciente interés de la comunidad científica por el proyecto Universal Dependencies (UD), en el que se propone un formato de anotación morfosintáctica destinado a crear una representación "universal" de la morfología y sintaxis aplicable a diferentes idiomas, en este trabajo se utiliza este formato con el propósito de realizar un estudio desde una perspectiva multilingüe (italiano, inglés, francés y español). Aunque el formato UD se concibió originalmente para ser aplicado a textos escritos en la lengua "estándar" desde el punto de vista de las normas morfosintácticas y la puntuación, recientemente se ha comenzado a aplicar el mismo esquema también a contenidos generados por usuarios en línea (User-Generated Content, UGC), es decir, a textos extraídos de redes sociales, blogs, foros y plataformas de microblogging, como las páginas de Reddit, Twitter o Wikipedia, en el que se utiliza un registro más informal. Inevitablemente, la aplicación de este formato de anotación a un registro textual tan peculiar, en el que los textos van acompañados de elementos multimedia como enlaces, fotos y videos, emojis y puntuación no estandarizada, ha abierto varios problemas en la comunidad de Universal Dependencies, muchos de los cuales siguen siendo objeto de un debate abierto y acalorado en la actualidad.

En este trabajo, por lo tanto, se presenta una descripción exhaustiva del formato de anotación morfosintáctica de UD, en particular, subrayando las cuestiones más relevantes en cuanto a su aplicación a los UGC generados en las redes sociales. El objetivo final es analizar y comprobar si estas anotaciones morfosintácticas sirven para obtener información útil para los sistemas/modelos de detección de la ironía y del stance o posicionamiento. Se presentarán dos subáreas de análisis de sentimientos y se utilizarán como ejemplos de estudio para probar las hipótesis de la investigación: el primer caso se centra en el área de la detección automática de la ironía y el segundo en el área de la detección del stance o posicionamiento. En ambos casos, se proporcionan los antecendentes y trabajos relacionados notas históricas que pueden servir de contexto para el lector, se introducen/plantean los problemas encontrados y se describen las distintas actividades propuestas para resolver estos problemas en la comunidad de la lingüística computacional. Se presta especial atención a los recursos actualmente disponibles, así como a los desarrollados específicamente para el estudio de los fenómenos antes mencionados. Finalmente, a través de la descripción de una serie de experimentos, llevados a cabo tanto en campañas de evaluación como en estudios independientes, se describe la contribución que la sintaxis puede brindar a la resolución de esas tareas, como subcampos del análisis de sentimientos.

Esta tesis es el resultado de toda la investigación que he llevado a cabo durante mi doctorado en una colección revisada de mi carrera de doctorado de los últimos tres años y medio, y se ubica dentro de la tendencia creciente de estudios dedicados a hacer que los

resultados de la Inteligencia Artificial sean más explicables, yendo más allá del logro de puntajes más altos en la realización de tareas, sino más bien haciendo comprensibles sus motivaciones y qué los procesos sean más comprensibles para los expertos en el dominio.

La contribución principal y más novedosa de este trabajo consiste en la explotación de características (o rasgos) basadas en la morfología y la sintaxis de dependencias, que se utilizaron para crear las representaciones vectoriales de textos procedentes de redes sociales en varios idiomas y para dos tareas diferentes. A continuación, estas características se han emparejado/combinado con una variedad de clasificadores de aprendizaje automático, con algunas redes neuronales y también con el modelo de lenguaje BERT.

Los resultados sugieren que la información sintáctica basada en dependencias utilizada es muy informativa para la detección de la ironía y menos informativa en lo que respecta a la detección del posicionamiento. No obstante, la sintaxis basada en dependencias podría resultar útil en la tarea de detección del posicionamiento si, en primer lugar, la detección de ironía se considera un paso previo al procesamiento en la detección del posicionamiento. También creo que el enfoque basado casi completamente en sintaxis de dependencias que propongo en esta tesis podría ayudar a explicar mejor un fenómeno pragmático tan difícil de detectar e interpretar como la ironía. De hecho, los diversos estudios que se presentan permitieron analizar si las estructuras sintácticas, independientemente del idioma , pueden aportar información útil para comprender y clasificar si un mensaje es irónico o no.

Resum

La present tesi s'emmarca dins de l'ampli panorama d'estudis relacionats amb el Processament del Llenguatge Natural (NLP). En concret, es tracta d'un treball de Lingüística Computacional (CL), l'objectiu principal del qual és estudiar en profunditat la contribució de la sintaxi en el camp de l'anàlisi de sentiments i, en concret, aplicat a l'estudi de textos extrets de les xarxes socials o, més en general, de continguts online.

A més, el recent interès de la comunitat científica pel projecte Universal Dependències (UD), en el qual es proposa un format d'anotació morfosintàctica destinat a crear una representació " universal " de la morfologia i sintaxi aplicable a diferents idiomes, en aquest treball s'utilitza aquest format amb el propòsit de realitzar un estudi des d'una perspectiva multilingüe (italià, anglès, francès i espanyol). Tot i que el format UD es va concebre originalment per ser aplicat a textos escrits en la llengua " estàndard " des del punt de vista de les normes morfosintàctiques i de la puntuació, recentment s'ha començat a aplicar el mateix esquema també a continguts generats per usuaris(User-Generated Content, UGC), és a dir, a textos extrets de xarxes socials, blocs, fòrums i plataformes de microblogging, com les pàgines de Reddit, Twitter o Wikipedia, en què s'utilitza un registre més informal. Inevitablement, l'aplicació d'aquest format d'anotació a un registre textual tan peculiar, en el qual els textos van acompanyats d'elements multimèdia com enllaços, fotos i vídeos, emojis i puntuació no estandarditzada, ha plantejat diversos problemes a la comunitat d'Universal Dependencies, molts dels quals segueixen sent objecte d'un debat obert i acalorat en l'actualitat.

En aquest treball, per tant, es presenta una descripció exhaustiva del format d'anotació morfosintàctica d'UD, en particular, posant més èmfasi en les qüestions més rellevants pel que fa a la seva aplicació als UGC generats a les xarxes socials. L'objectiu final és analitzar i comprovar si aquestes anotacions morfosintàctiques serveixen per obtenir informació útil per als sistemes/models de detecció de la ironia i del stance o posicionament. Es presentaran dues subàrees de l'anàlisi de sentiments i s'utilitzaran com a exemples d'estudi per provar les hipòtesis de la investigació: el primer cas se centra en l'àrea de la detecció automàtica de la ironia i el segon en l'àrea de la detecció del stance o posicionament. En tots dos casos es proporcionen els antecedents i treballs relacionats que poden servir de context per al lector, es plantegen els problemes trobats i es descriuen les diferents activitats proposades per resoldre aquests problemes en la comunitat de la lingüística computacional. Es fa especialment referència als recursos actualment disponibles, així com als desenvolupats específicament per a l'estudi dels fenòmens abans esmentats. Finalment, a través de la descripció d'una sèrie d'experiments, duts a terme tant en campanyes d'avaluació com en estudis independents, es descriu la contribució que la sintaxi pot oferir a la resolució d'aquestes tasques, com subcamps de l'anàlisi de sentiments.

Aquesta tesi és el resultat de tota la investigació que he dut a terme durant el meu doctorat els últims tres anys i mig, i se situa dins de la tendència creixent d'estudis dedicats a fer que els resultats de la Intel·ligència Artificial siguin més explicables, que vagin més enllà de l'assoliment de puntuacions més altes en la realització de tasques, sinó més aviat fent comprensibles les seves motivacions i què els processos siguin més

comprensibles per als experts en el domini.

La contribució principal i més nova d'aquest treball consisteix en l'explotació de característiques (o trets) basades en la morfologia i la sintaxi de dependències, que s'utilitzen per crear les representacions vectorials de textos procedents de xarxes socials en diversos idiomes i per a dues tasques diferents. A continuació, aquestes característiques s'han combinat amb una varietat de classificadors d'aprenentatge automàtic, amb algunes xarxes neuronals i també amb el model de llenguatge BERT.

Els resultats suggereixen que la informació sintàctica utilitzada basada en dependències és molt informativa per a la detecció de la ironia i menys informativa pel que fa a la detecció del posicionament. Malgrat això, la sintaxi basada en dependències podria ser útil en la tasca de detecció del posicionament si, en primer lloc, la detecció d'ironia es considera un pas previ al processament en la detecció del posicionament. També crec que l'enfocament basat gairebé completament en sintaxi de dependències que proposo en aquesta tesi podria ajudar a explicar millor un fenomen pragmàtic tan difícil de detectar i d'interpretar com la ironia. De fet, els diversos estudis que es presenten han permès d'analitzar si les estructures sintàctiques, independentment de l'idioma, poden aportar informació útil per comprendre i classificar si un missatge és irònic.

This thesis has been revised and positively evaluated, considering it admissible for the final defense, by Dr. **Els, Lefever** (Department of Translation, Interpreting and Communication, LT^3 – Language and Translation Technology Team, Ghent University, Belgium), Prof. Dr. **Joakim, Nivre** (Department of Linguistics and Philology, Uppsala University, Sweden), and Dr. **Kareem, Darwish** (Arabic Language Technologies, Quatar Computing Research Institute, Hamad Bin Khalifa University, Quatar).

List of abbreviations

- **AI:** Artificial Intelligence
- **BERT:** Bidirectional Encoder Representations from Transformers
- **BiLSTM:** Bidirectional Long Short-Term Memory
- **BOW:** Bag of Words
- **CL:** Computational Linguistics
- **CMC:** Computer-Mediated Communication
- **CNN:** Convolutional Neural Network
- **DL:** Deep Learning
- **DT:** Decision Tree
- **GRU:** Gated Recurrent Unit
- **HSC:** Hate Speech Corpus
- **IAA:** Inter-Annotator Agreement
- **LAS:** Labeled Attachment Score
- **LR:** Logistic Regression
- **LSTM:** Long Short-Term Memory
- **MDS:** Multidimensional scaling
- **MFC:** Most Frequent Class
- **ML:** Machine Learning
- **MLP:** Multilayer perceptron
- **MV:** Majority Voting
- **NB (and MNB):** Naïve Bayes (and Multinomial Naïve Bayes)

- **NLP:** Natural Language Processing
- **P:** Precision
- **PoS:** Part of Speech
- **R:** Recall
- **RF:** Random Forest
- **RNN:** Recurrent Neural Network
- **SD:** Stance Detection
- **SDQC:** Support, Deny, Query, Comment
- **SVC (or SVM):** Support Vector Classifier (or Support Vector Machine)
- **UAS:** Unlabeled Attachment Score
- **UD:** Universal Dependencies
- **UGC:** User-Generated Content

Contents

1 Introduction **1**
- 1.1 Natural Language Processing . . . 1
  - 1.1.1 Automatic text classification . . . 3
  - 1.1.2 Sentiment analysis, irony detection and stance detection . . . 6
- 1.2 Morphology and syntax . . . 7
  - 1.2.1 *Universal Dependencies* . . . 8
  - 1.2.2 The UD framework and social media data . . . 9
- 1.3 Problem statement and research questions . . . 12
- 1.4 Structure of the thesis . . . 15
- 1.5 Contributions . . . 16

2 Irony detection **20**
- 2.1 Shared tasks and corpora for irony detection . . . 22
  - 2.1.1 Organization of *IronITA 2018* . . . 27
  - 2.1.2 The creation of an ironic corpus: TWITTIRÒ . . . 33
- 2.2 From feature-based approaches to deep learning . . . 50
- 2.3 Irony detection using dependency syntax . . . 52
  - 2.3.1 Participation in the *IroSVA 2019* shared task . . . 53
  - 2.3.2 Multilingual irony detection with neural models . . . 57
- 2.4 Concluding remarks on irony detection . . . 66

3 Stance detection **69**
- 3.1 Shared tasks and corpora for stance detection . . . 70
  - 3.1.1 Organization of *SardiStance 2020* . . . 74
- 3.2 Machine learning approaches for stance detection . . . 82
  - 3.2.1 Participation in the *StanceCat 2017* shared task . . . 86
- 3.3 Stance detection using dependency syntax . . . 90
  - 3.3.1 Participation in the *RumorEval 2019* shared task . . . 91
  - 3.3.2 Multilingual stance detection with neural models . . . 97
- 3.4 Concluding remarks on stance detection . . . 102

4 The interaction of irony and stance **105**
- 4.1 Annotating irony on the *SardiStance* dataset . . . 106
- 4.2 Analyzing morphology and syntax in ironic tweets . . . 108

4.3 A tentative error analysis comparing irony and stance 112

5 Conclusions and future work **114**
5.1 Conclusions 116
5.2 Future work 119

Chapter 1

Introduction

The present PhD thesis is developed within the framework of *Computational Linguistics*, an area that sees the connection of Linguistics and Computer Science. In particular, the dissertation broadens within the field of *Natural Language Processing*, which deals with the computational treatment of texts.

Being interested in opinion, sentiment, and subjectivity in texts, my principal target is that of analyzing the impact of syntactic information in sentiment analysis related tasks. In particular I chose *irony detection* and *stance detection* as two main case studies. Previous work that inspired this PhD thesis is to be found in the research activities in relation to sentiment analysis, conducted inside the *CCC group* in the *Computer Science Department* of the *University of Turin* as well as inside the *PRHLT research group* in the *Departamento de Sistemas Informáticos y Computación* of the *Universitat Politècnica de València*. Many other researches contributed in inspiring the work I have done in this thesis and they will be quoted in the following chapters accordingly, and they will be added step by step to the bibliography. Some of them are related to linguistic theories, others are connected to natural language processing and sentiment analysis, and others to machine learning.

1.1 Natural Language Processing

Natural Languages Processing (NLP) is a highly interdisciplinary field. Linguistics, Computer Science, Social and Cognitive Science are involved with the aim of using computers for understanding and manipulating natural languages. In general, NLP can be defined as the scientific discipline that investigates the interactions between computers and natural language.

Several NLP practical applications are used by multitudes of people on a daily basis, on the Internet and on mobile devices such as: spam filtering, recommendation in search engines, assisting chat bots, speech recognition, text dictation, machine translation, and so on. Among other tasks we can also list, there are parsing, summarization, duplicate detection, part of speech tagging, name entity recognition, text classification, sentiment analysis, and many others [Jurafsky and Martin, 2008].

Furthermore, as we can see from the diagram in Figure 1.1, it would be impossible to truly understand the basic principles of NLP without an adequate mention also to other disciplines involved in Artificial Intelligence (AI), Machine Learning (ML) and Deep Learning (DL). These different areas contribute in various ways to the developments and advancements of NLP technology, by joining forces and interacting one with the other, and providing methods for dealing with the same phenomena by means of different approaches, they all together strengthen the richness of the great interdisciplinary field that NLP is.

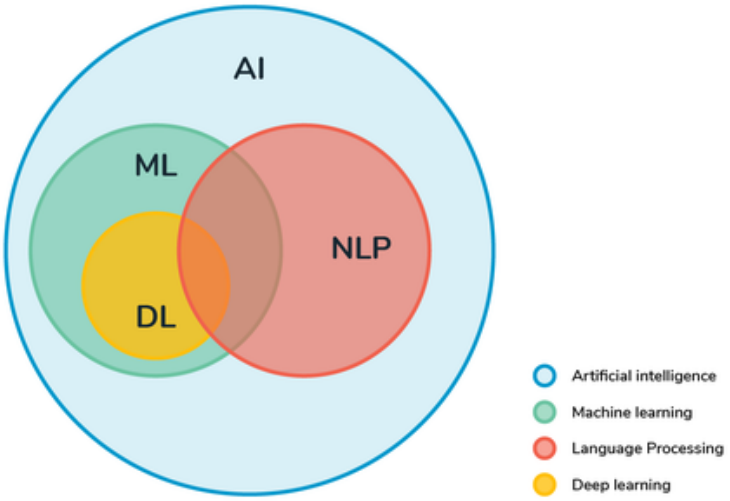


Figure 1.1: Artificial Intelligence, Machine Learning, and Natural Language Processing.[2]

As natural languages can be conveyed either through orality or by writing, the first big differentiation that needs to be made, is that NLP applications can regard either the study of speech or that of texts. Having said that, in the present thesis I will concentrate solely on the latter, with a particular focus on a type of text that is considered to be 'non-standard', i.e., that which is extracted from social media and different media for online communication among users (*user-generated content*, UGC). Regarding this topic, I will be more exhaustive in Section 1.2.

Certainly, "What do other people think?" has always been such an interesting question that it has always been affecting our everyday life and activities. As I mentioned earlier, this thesis collocates within the framework of sentiment analysis (sometimes mentioned as *opinion mining*), a discipline whose main intent is precisely that of proposing computational models that are able to reply to these kinds of questions automatically. Sentiment analysis, among the areas belonging to the field of NLP, is one of those with the longest research tradition and development in a large span of time. In the last decades,

[2]Taken from: https://athenatech.tech/f/ai-machine-learning-ml-and-natural-language-processing-nlp.

thanks to the huge blossoming of interest towards Internet, the enlarging quantity of contents written on the web, but also thanks to the quick development of machine learning and deep learning techniques, many small, extremely specific areas derived from sentiment analysis have been born and that step by step it was no longer just a single field, but a manifold of deeply specific and independent tasks such as: hate speech detection, irony detection, stance detection, and so on.

The early projects in the area of sentiment analysis were primarily focused on interpretation of metaphors, narratives, expressions of points of view and emotions. Later on, also bridging with other kindred subjects such as psychology, anthropology and sociology, sentiment analysis slowly began a new phase that Pang and Lee [2008] define "*social media monitoring and analysis*". Social media, is indeed, the major source of data that is exploited within these fields, as a huge amount of data which is created online instantly, is immediately available at hand and easily exploitable in terms of data collection for research purposes.

Even though, as I said, sentiment analysis may nowadays consist of many different tasks, which all have their unique characteristics and peculiarities, in general, almost all of them can be declined, interpreted and solved as classification tasks. Some of them are interpreted as having a binary meaning (e.g., presence/absence of irony in a text) while others are decoded as multiclass classifications (e.g., stance detection with 'agree', 'disagree', and 'neutral'). Inside the same task, different aspects could be taken into observation, as well as different dimensions measured with diverse metrics. For instance in some works regarding hate speech detection, the dimensions of offensiveness, aggressiveness and stereotype are measured [Poletto et al., 2017].

Even though approaching these tasks as text classification problems is dominant, it is important to stress that some of these tasks –such as stance detection– have been solved as a community detection task or as clustering tasks [Darwish et al., 2020]. All in all, independently from their declination each task is based on principles deriving from *automatic text classification*, so, it is important to highlight the principal characteristics of it, in order to understand the methodological ground on which the present thesis is built.

1.1.1 Automatic text classification

Automatic text classification is a machine learning task that automatically assigns a given document to a set of pre-defined categories based on its textual content and extracted features. There are many practical applications that profit from text classification advancements such as spam detection, language identification or, as in our case, the many facets of sentiment analysis tasks. In this section, I will illustrate briefly all the necessary requirements that are useful in order to understand the basics of automatic text classifications techniques.

The first indispensable step to propose and evaluate a classification method, is the creation or collection of a *corpus*. A corpus is typically a collection of documents, utterances, sentences, blog posts or, as in our case, tweets. Those instances, are considered to be representative of and usable for lexical, grammatical, or linguistic analyses [Ungeheuer

and Wiegand, 2008].

When the goal of the corpus development is to apply supervised machine learning methods, the procedure of corpus collection and annotation usually entails that group of human annotators labels each document assigning one of the possible classes (or labels, tags) to it. The annotators normally rely on some guidelines and on annotation schemes, that are supplied to them in order to have a clear orientation on how to annotate the new instances, thus guiding the annotation process.

After this procedure is over, a certain measure, functioning as index for Inter-Annotator Agreement (IAA) is calculated, in order to assess the quality of the annotated dataset. In many tasks related to sentiment analysis, the preferred coefficients used are Cohen's kappa or Krippendorff's alpha [Artstein and Poesio, 2008]. These steps generate the so called *gold standard* corpus: a collection of labeled documents (or tweets) where each label is accepted as the most valid one.

It is extremely important to create a gold standard corpus. It is indispensable for training supervised systems to have a large correctly annotated corpus, but also for testing unsupervised text classification methods, the availability of smaller annotated datasets is necessary to be used as evaluation benchmark. Generally, assuming that we are within a supervised context, which is currently the most common setting, a large gold standard is created which is first divided into two subsets: *training set* and *test set*.[3] The training set, as the name itself clarifies, is exploited for training an automatic model, while the test set is used for comparing the predicted labels against the gold labels. One of the simplest and most common splitting methods consists in randomly dividing the corpus in 80% of training documents and in 20% of test ones. Another common method is to perform a k-fold validation (usually 5-fold or 10-fold) dividing the corpus in k folds and using each fold once as test and the remaining $k-1$ folds as training.

As κ and α are usually used to compute IAA, other specific units of measure are needed in order to test the performances of such automatic systems. Depending on the type of classification (e.g., binary, multi-class, etc...) performance can be evaluated using a variety of different metrics. The most exploited ones in the community of researchers dealing with sentiment analysis, and related tasks are: *precision*, *recall*, *F1-score* and *accuracy*.

In Figure 1.2, I reported an intuitive scheme[4] of the concepts underlying precision and recall. The color green represents the correct predictions, while the color red the wrong ones. The documents within the circle were predicted as true, the other ones were predicted as negative. Following, the corresponding equations:

$$precision = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{retrieved\ documents\}|} \tag{1.1}$$

$$recall = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{relevant\ documents\}|} \tag{1.2}$$

[3] Sometimes also a third division is provided: the *development set*.

[4] The photo on precision and recall was taken from: https://en.wikipedia.org/wiki/Precision_and_recall.

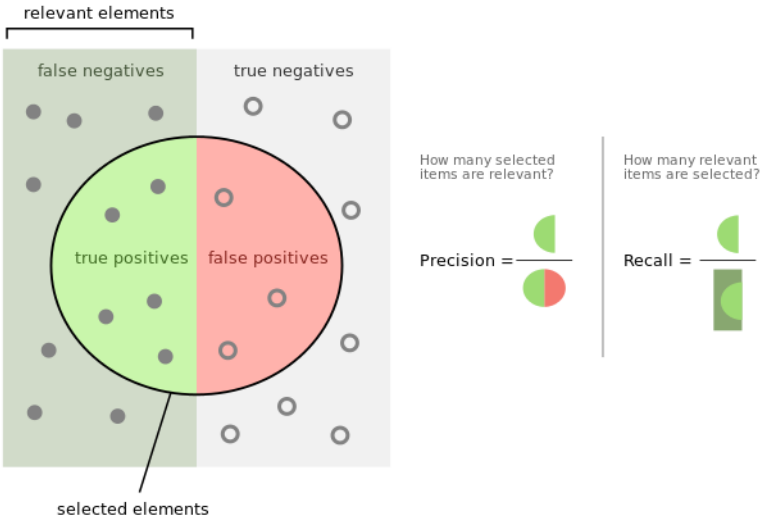


Figure 1.2: Graphical representation of precision and recall.

For classification tasks, the terms *true positives*, *true negatives*, *false positives*, and *false negatives* are used. With them we can compare the results of the classifier against the labels provided by the gold standard corpus. The terms positive and negative refer to the classifier's prediction (sometimes known as the expectation), and the terms true and false refer to whether that prediction corresponds to the gold standard. The four outcomes can be formulated in a 2×2 contingency table or confusion matrix, as follows:

$$precision = \frac{tp}{tp + fp} \tag{1.3}$$

$$recall = \frac{tp}{tp + fn} \tag{1.4}$$

Two measures that can be derived by the calculus of precision and recall, and the count of true/false positives/negatives are also commonly used for evaluating a classification method: *accuracy* and F1-score. *Accuracy* evaluates the number of correct predictions (*true positives*) divided by the total number of predictions, multiplied by 100 to turn it into a percentage value.

$$accuracy = \frac{tp}{tp + tn + fp + fn} * 100 \tag{1.5}$$

Accuracy could be the most accurate metric, depending on the task definition and particularly in those cases in which an almost perfectly balanced label distribution between classes is present. In fact, if the distribution of labels inside a corpus is highly unbalanced towards a class C, the automatic model is likely to assign the most common label to all documents. Thus, the value of the accuracy calculated on the dataset would be equal to the most common label frequency. In cases like the one just described, it is

more feasible to use a measure that takes into account both the precision and the recall: the *F1-score*.

$$\text{F1-score} = 2 * \frac{precision * recall}{precision + recall} \tag{1.6}$$

Furthermore, in cases of non-binary classification tasks, we might want to be interested in measuring the average among the F1-score of each distinct class (*macro F1-score*, or simply *F-macro*):

$$\text{F-macro} = \frac{1}{N} * \sum_{i=1}^{N} * \text{F1-score} \tag{1.7}$$

Having a great amount of possibilities, it is therefore, extremely important to choose a suitable metric depending on the number of classes, but especially depending on the balanced distribution of instances among those classes. For instance, relating to one of the two case studies investigated in this thesis, irony detection and stance detection, the first is usually realized as a binary classification task. Therefore, if the number of instances in the ironic class (*positive*) equals the number of instances in the non-ironic class (*negative)*, we might as well evaluate systems' performances through accuracy. On the other hand, if the dataset is highly unbalanced skewing, for instance, towards the non-ironic class (which is often done on purpose, as to recreate a real-life scenario) it is more feasible to use the F1-macro, averaging the F1-scores of the two classes.

1.1.2 Sentiment analysis, irony detection and stance detection

As I mentioned several times by now, the two case studies that I will deal with in the present thesis are, namely: irony detection and stance detection. They both started to develop as independent tasks, detached from sentiment analysis itself, as the community of computational linguistics continuously grew year after year, but also as the interest of economy and politics started to increase widely regarding the monitoring of people's online opinions.

As also anticipated before, the main research covered in the field of sentiment analysis is the computational treatment of *opinion*, *sentiment*, and *subjectivity* expressed through *polarity* in text [Kouloumpis et al., 2011]. Furthermore, the subject in question is strictly connected to the world of commerce and economics where the trendiest interrogations are, for example: "*Is this product review positive or negative?*", "*Is the customer writing this email satisfied or dissatisfied?*", "*Based on a sample of texts, how are people responding to this product release?*" or even "*How did bloggers' attitudes of presidential candidates change since the election?*". Certain matters do not only provide everyday challenges for researchers, but their results may have a huge impact on society altogether [Lai, 2019].

Sentiment analysis, in particular, using knowledge adopted from statistics, or from machine learning methods, tries to extract or identify the *sentiment* content of a text unit [Pang and Lee, 2008]. One of the main challenges in sentiment analysis is certainly given by the automatic recognition of frequent expressions naturally used in human language,

and, for example, decipher if an expression is positive or negative, thus expressing its *polarity*.

As we will see in Chapter 2, entirely dedicated to irony detection, irony might function as a polarity reverser [Bosco et al., 2013, Reyes and Rosso, 2014], thus hiding the real sentiment of an utterance and subsequently hindering the performance of an automatic system. Therefore, in this sense I believe that solving the problem of irony detection could be regarded as an important and necessary step to be dealing with when computing the overall sentiment of a document [Hernández Farías and Rosso, 2016]. This is the main reason for which I decided to select irony detection as a first case study.

On the other hand, as my second focus and second case study, I opted for that of stance detection. Typically, stance detection is defined as the task of automatically determining whether the author of a text is in favour, against, or neutral towards a given target [Mohammad et al., 2016]. In this thesis, I aimed at the possibility to create and evaluate an approach that could be exploited both in binary classification tasks (e.g., irony detection, see Chapter 2) as well as in multiclass classification tasks (e.g., stance detection, see Chapter 3).

Furthermore, most computational research in irony detection focuses primarily on content-based processing of the linguistic information using semantic or pragmatic devices, most of the time neglecting syntax. However, in addition to these pragmatic devices, some hints about the role of syntax can be found in linguistic literature where the violation of syntactic rules has been reported as a possible trigger of the phenomenon [Michaelis and Feng, 2015, Karoui et al., 2017, Chakhachiro, 2019]. For this main reason, I have decided to concentrate my research on the study of morphology and syntax and especially investigating their contribution in the task of irony detection. As a parallel research I was also curious to check whether syntax could be helpful, not only in the detection of irony, but whether it might prove effective also in other NLP related tasks. This is why I chose a second case study and decided to investigate the impact of syntax on the stance detection task. To the best of my knowledge, syntactic features have never been explored with regard to neither irony detection nor stance detection, therefore, in this sense, I believe the syntax-based approach presented in this thesis is novel and original. The investigation of these two different tasks according to a syntactic perspective also provides the opportunity of discovering some difference between them and to better reflect upon the suitability of multi-tasks approaches rather than approaches especially dedicated to each of them.

1.2 Morphology and syntax

In the present thesis, as a main objective, I would like to take advantage of the morphological and syntactic knowledge, in order to address the two tasks chosen as case studies: irony detection and stance detection. In some sense, my investigation collocates within the growing trend of studies devoted to make more explainable Artificial Intelligence[5] re-

[5]See e.g., XAI at https://en.wikipedia.org/wiki/Explainable_artificial_intelligence and the variety of recent workshops on this topic among which *IJCAI-PRICAI 2020 Workshop on Explainable*

sults, going beyond the achievement of high scores in performing tasks but rather making their motivations understandable for experts in the domain.

In order to do so, firstly, I needed to find a format able to encode morphosyntactic knowledge. Considering the possibility of applying my current and future research in multi-domain and multi-lingual perspective, I have directed my attention towards formats that in principle allow the portability across different textual genres and languages. Without any further doubt, I took the decision of exploiting the format of *Universal Dependencies* (UD) [Nivre et al., 2016, 2020], which, since its creation in 2016 has rapidly become the *de facto* standard for encoding morphology and syntax inside the community of computational linguists.

1.2.1 *Universal Dependencies*

Universal Dependencies is an open community effort to create cross-linguistically consistent treebank annotation for many languages with the goal of facilitating multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective [Nivre et al., 2020]. The whole project started in 2016, with the first guidelines, involving treebanks for 33 different languages, and it is very much alive today with currently available treebanks for 104 languages. It draws on a long tradition of typologically oriented grammatical theories – in particular dependency grammar – in which grammatical relations between words are centrally used to explain how predicate–argument structures are encoded morphosyntactically in different languages while morphological features and part-of-speech classes give the properties of words [De Marneffe et al., 2021].

The current annotation scheme and tagset are based on (Universal) Stanford Dependencies [De Marneffe et al., 2006, De Marneffe and Manning, 2008, De Marneffe et al., 2014], Google universal part-of-speech tags [Petrov et al., 2012], and the Interset interlingua for morphosyntactic tagsets [Zeman, 2008]. The general philosophy is to provide a universal inventory of categories and guidelines to facilitate consistent annotation of similar constructions across languages, while allowing language-specific extensions when necessary. This framework is thus a perfectly good basis for crosslinguistically consistent annotation of typologically diverse languages in a way that supports computational natural language understanding as well as broader linguistic studies [De Marneffe et al., 2021]. The main principles underlying UD are those of offering a linguistic representation that could be useful for morphosyntactic research, semantic interpretation, and for practical natural language processing across different human languages. It highlights simple surface representations that allow parallelism between similar constructions across different languages, despite differences of word order, morphology, and the presence or absence of function words.

In the UD framework, a dependency grammar perspective is adopted, meaning that a phrase has a *head* and all the other components that are contained are *dependents* of that head [De Marneffe et al., 2021]. A "*dependency relation*" is represented in diagrams

Artificial Intelligence (XAI) https://sites.google.com/view/xai2020/home.

(called '*syntactic trees*') by an arrow from the head to the dependent. In Figure 1.3 a partial analysis of a clause is shown.

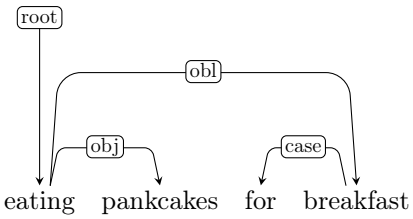


Figure 1.3: Partial UD analysis for a clause.

Through these dependencies, the words of a sentence are organized into a tree structure with the main predicate as the root (*eating*), and dependencies are marked with grammatical relation labels such as *obj* for direct object, *obl* for oblique nominal and *case* for case marking, as in the example shown above.

In UD the primary status is given to words.[6] Words, in fact, are the basic elements connected by dependency relations. They have morphological properties and enter into syntactic relations. In all linguistic theories it is assumed that words can be classified by a word class or part of speech (PoS) according to their behavior within the language system. Because of that, in the framework of Universal Dependencies we distinguish 17 macro-categories of words and other elements of text, and we assign them the labels ("universal part-of-speech tags" → UPOS). The complete list of part of speech tags can be seen in the UD official webpage: `https://universaldependencies.org/u/pos/index.html`.

Then, words are connected to the head via a grammatical relation drawn from a universal typology of 37 grammatical relations that can be read here: `https://universaldependencies.org/u/dep/index.html` – [De Marneffe et al., 2021]. The grammatical relations are organized around whether the head is the head of a clause or nominal, and whether the dependent is a clause, nominal, or modifier. The only exception is made for the `root` relation, which is used for the root of the sentence, with a dummy head that does not need to be explicit.

The application of the UD framework to a manifold of different data genres revealed the existence of several highly frequent constructions that are not discussed in comprehensive grammars. For instance, in NLP it is very frequent to resort to social media data (such as tweets) for analyzing different phenomena. In order to apply the UD framework to non-standard texts a variety of recommendations needs to be made, therefore, I discuss some of them in the following section.

1.2.2 The UD framework and social media data

As mentioned above, the "universality" of the UD framework allows us to think of the present study in a multilingual perspective, because we can aim at encoding datasets in

[6] Despite the challenges in defining words in a crosslinguistically consistent manner.

different languages with the format provided by UD, resulting in datasets that, exploiting the same annotation tagset, are easily comparable. As I briefly mentioned in the abstract, and as I will make clear in both Chapters 2 and 3, I will perform experiments exploiting morphosyntactic elements in a multilingual scenario, taking into consideration four different languages: English, Spanish, French and Italian.

Additionally, the datasets that I collected are all extracted from the microblogging platform Twitter, for a variety of underlying reasons. Microblogging today has become a very popular communication tool among Internet users. Twitter, especially, is used to post emotional states, updates on personal life events or to share beliefs on various different topics. It has been studied, that more and more users post opinions about products they buy, services they use, or express their political and religious beliefs. For these reasons, Twitter has increasingly become a valuable source of people's opinions and thus, the primary source of data for sentiment analysis applications and related tasks [Pak and Paroubek, 2010, 2011].

If on the one hand using Twitter as a corpus for sentiment analysis provides many useful information, on the other hand we must take further precautions when using it for social analysis. Moreover, it can be especially challenging from a linguistic point of view. Many linguistic studies show, in fact, how written language changes enormously in tweets (and social media in general), and resembles a lot more spoken language. Accordingly, even though Twitter should be considered as a written corpus, its data can not be treated as proper written language because *diamesic variation*[7] is reduced to zero in texts that belong to *computer-mediated communication* (CMC) [Bazzanella, 2011], and more in general, this phenomenon applies to all data belonging to the broader category of *user-generated content*. It is, for instance, more colloquial and with characteristics more similar to *speech*. Tweets in general contain a significant amount of deviations from common grammatical norms in every language [MacLeod and Grant, 2012].

Obviously, with all these factors evolving and the times changing, the research community itself has been taking into account more and more the fact that a new kind of data treatment was necessary and that a renovation regarding annotation formats, guidelines and tagsets was also needed. Especially taking into account the scientific community within computational linguistic studies that deals with morphology, syntax and the contributors of Universal Dependencies, it can be appreciated that in order to successfully process the data available from such sources, an increasing number of contributions, especially on Part-of-Speech tagging [Gimpel et al., 2011, Owoputi et al., 2013, Lynn et al., 2015, Bosco et al., 2016a, Çetinoğlu and Çöltekin, 2016, Proisl, 2018, Rehbein et al., 2018, Behzad and Zeldes, 2020] and parsing [Foster, 2010, Petrov and McDonald, 2012, Kong et al., 2014, Liu et al., 2018b, Sanguinetti et al., 2018] has been produced in the last decade. Nevertheless, the automatic processing of user-generated content still represents a challenging task, as it is a continuum of text sub-domains that vary considerably according to the specific conventions and limitations posed by the medium used (blog, discussion forum, online chat, microblog, etc.), the degree of "canonicalness" with

[7]It is the variation of a language depending on the medium of communication. For example, the variety used during a phone call will be different from the variety used to write an e-mail.

respect to a more standard language, as well as the linguistic devices adopted to convey a message. Overall, however, there are some well-recognized phenomena that characterize UGC as a whole, but nevertheless they continue to highlight its treatment as difficult task [Foster, 2010, Seddah et al., 2012, Eisenstein, 2013].

As the availability of *ad hoc* training resources remains an essential factor for the analysis of these texts, the last decade has seen numerous resources of this type being developed. A good proportion of those resources that contain syntactic analyses have been annotated according to the Universal Dependencies scheme [Nivre et al., 2016]. At the time of writing of this thesis, a total of 104 languages are represented within this vast project, with 183 treebanks[8] with contributions from 416 researchers around the world, dealing with extremely varied genres, ranging from news to fiction, medical, legal, religious texts, etc. This linguistic and textual variety demonstrates the adaptability of the annotation framework.

On the one hand, this flexibility opens up the possibility of also adopting the UD scheme for a broad range of user-generated text types (such as tweets). A framework which is proven to be readily adaptable is indeed more likely to fit the needs of diverse UGC data sources, and the wealth of existing materials makes it potentially easier to find precedents for analysis whenever difficult or uncommon constructions are encountered. On the other hand, the current UD guidelines do not fully account for some of the specifics of UGC domains, thus leaving it to the discretion of the individual annotator (or teams of annotators) to interpret the guidelines and identify the most appropriate representation of these phenomena.

In recent work, together with other colleagues of the UD community, we tried to draw attention to the annotation issues of UGC, while attempting to find a cross-linguistically consistent representation, all within a single coherent framework [Sanguinetti et al., 2020, 2021]. In said works, we provided an overview of the existing treebanks of user-generated texts from the Web in six different languages,[9] with a focus on comparing their varying annotation choices with respect to certain phenomena typical of this domain. Next, we presented a systematic analysis of some of these phenomena within the context of the framework of UD, surveying previous solutions, and we proposed, where possible, guidelines aimed at overcoming the inconsistencies found among the existing resources. Given the nature of the phenomena covered and the fact that the existing relevant resources only cover a handful of languages, I am aware that the debate on their annotation is still wide open. Nonetheless, the proposals in such articles, represent the consensus of a fairly large group of UD contributors working on diverse languages and media, with the goal of building a critical mass of resources that are annotated in a consistent way. As such, it can be used as a reference when considering alternative solutions, and it is hoped that the survey of treatments of similar phenomena across resources will help future projects in making choices that are as comparable as possible to common practices in the existing datasets. Furthermore, altogether with the group of UD contributors, we aim at catching the attention and interest of a wider public, in order to shed some light on the importance

[8] Release v2.7, November 15, 2020. For more information, see `https://universaldependencies.org`.
[9] I.e., German, English, French, Irish, Italian, Turkish.

of common guidelines able to treat both morphologically and syntactically texts that are extracted from social media. And that stress how the benefits obtained from this kind of research would apply to a variety of disciplines and NLP applications.

Taking into consideration all these factors, and being myself a contributor of the UD project, with the TWITTIRÒ-UD corpus [Cignarella et al., 2019b] (as I will describe in Chapter 2), I believed the time was ripe for taking UD as a format to encode morphosyntactic information. Especially, the new focus is aimed at treating texts that are extracted from Twitter, and furthermore, exploit knowledge that could be extracted form this kind of representation as useful NLP features to solve a handful of sentiment analysis related tasks.

1.3 Problem statement and research questions

My main purpose is to explore the impact of morphosyntactic information in sentiment analysis related tasks.

Firstly, for both irony (Chapter 2) and stance (Chapter 3), I focus on the importance of the formulation of a clear problem statement, and the subsequent computational modeling of it. Secondly, I highlight my experience in the creation of annotated corpora for those problems, my contribution to the organization of shared tasks and the important lessons learnt, in terms of research understanding. Later on, I propose my first approaches to solve both tasks from a shallow perspective, starting to explore the most feasible way to represent morphosyntactic information, to extract it, and exploit it for classification purposes. I end this process by relying on the Universal Dependencies annotation format. Finally, after having encountered a satisfactory enough combination of features, I combine them – encoded in UD format – and I perform a handful of experiments in a variety of settings.

In the whole thesis a multilingual scenario is kept in mind, exploring four different language settings: English, Spanish, French and Italian, for both irony detection and stance. Furthermore, due to the availability of benchmark datasets, in Chapter 3, regarding stance detection, I also experiment on a fifth language, i.e., Catalan.

Therefore, speculating that we explore an effective way of exploiting syntactic information for the purpose of resolving the above-mentioned tasks, thus, the research questions I aimed at answering could be summarized as follows:

RQ-1 *Could features derived from morphology and syntax help to address the task of irony detection?* – In the work described in Section 2.3.2, I created linguistic resources syntactically annotated according to the well-known dependency-based scheme of *Universal Dependencies*. I took advantage of datasets used in shared tasks for irony detection and I enriched them with morphosyntactic annotation. From these new enriched resources (treebanks), I was able to extract NLP features that encode morphological and syntactic information, and furthermore, the versatility of the UD format allowed me to apply a dependency-based approach for the detection

of irony independently of a target language. Additionally, I experimented with UD-based word embeddings.

RQ-2 *To what extent does using resources such as treebanks for training NLP models improve the performance in irony detection?* – Referring to the datasets encoded in UD format, as in the previous research question, in Section 2.3.2, I proposed three distinct experimental settings. Firstly, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored, with the aim of finding the most informative set of features for detecting the presence of irony. In the second scenario two well-known word-embedding models are tested against gold standard datasets. Finally, in the third setting, dependency-based syntactic features are combined into the Multilingual BERT architecture. Furthermore, I experimented with datasets made available from previous shared tasks on irony detection in four languages: French (DEFT 2017 [Benamara et al., 2017]), English (SemEval-2018 Task 3 [Van Hee et al., 2018a]), Spanish (IroSvA [Ortega et al., 2019]) and Italian (IronITA [Cignarella et al., 2018b]).

In order to have a similar awareness and a well-balanced description for both case studies observed in this thesis, two further research questions about stance detection mirror the two first ones above.

RQ-3 *Could features derived from morphology and syntax help to address the task of stance detection?* – In Section 3.3.2, I describe how I created linguistic resources syntactically annotated according to the well-known dependency-based scheme of *Universal Dependencies*. I exploited six different datasets used in previous shared tasks (or made available by independent researchers) for stance detection and I enriched them with morphosyntactic annotation. From these new enriched resources (treebanks), I was able to extract NLP features that encode morphological and syntactic information, and once again, the versatility of the UD format allowed me to apply a dependency-based approach for the detection of stance independently of a target language.

RQ-4 *To what extent does using resources such as treebanks for training NLP models improve the performance in stance detection?* – Referring to the datasets encoded in UD format, as in the previous research question, in Section 3.3.2, I proposed two distinct experimental settings. Firstly, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored, with the aim of finding the most informative set of features for detecting the presence of stance. In the second scenario those dependency-based syntactic features are combined into the M-BERT architecture. Once again, I experimented with datasets made available by previous shared tasks or by independent researchers on stance detection in five different languages: English (*Hillary Clinton*, SemEval-2016 Task 6 [Mohammad et al., 2016]), Spanish and Catalan (*Catalan Independence*, StanceCat @ IberEval 2017 [Taulé et al., 2017]), French (*Macron*, from [Lai et al., 2020a]),

and Italian (*Constitutional Referendum*, [Lai et al., 2020a] and *Sardines Movement* [Cignarella et al., 2020b]).

However, even though I am curious to assess the applicability of syntax-based features to different task and domains, such as the task of stance detection, which I picked as a second case study, a clarification needs to be made. Syntax is certainly one of the levels of analysis of natural language that, due to its complexity, has been among the most studied within the context of classical NLP. Nevertheless, as I will describe in the following chapters, the attention of researchers in the field of sentiment analysis (especially regarding the goal of solving tasks) has been dedicated mostly towards semantics and pragmatics, rather neglecting syntax and the contribution it can give in task-solving directions. Most of the approaches aiming at solving sentiment analysis related tasks, in fact, use morphology (e.g., PoS tagging for helping the access to lexical resources) or some shallow syntactic analysis combined to other dimensions (e.g., for correctly linking negations or intensifiers to specific portion of a sentence), but without especially focusing on syntax alone. For instance inducing lexico-syntactic patterns based on syntactic dependencies and semantic frames that aim to capture the meaning of a sentence and provide a generalized representation of it.

On this behalf, also connecting with the research questions on SD, I must stress, that my intuition regarding the application of models that are solely based on syntax, to the task of stance detection, might prove not sufficient. In fact, just by observing the following two simple sentences as examples, and their respective dependency-based syntax trees:

Ex.1 I love the Sardines Movement.

Ex.2 I hate the Sardines Movement.

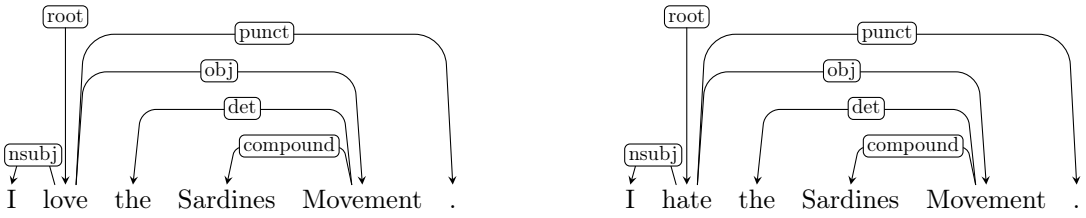


Figure 1.4: Dependency trees of the sentences in Ex.1 (left) and in Ex.2 (right).

without doubt, we can definitely say that the first one (Ex.1) would have a FAVOUR stance towards the target "Sardines Movement", whereas the second one (Ex.2) an AGAINST stance towards it. Nonetheless, the dependency-based syntax trees present no structural difference, they are perfectly equal, as the discriminant between the two examples is purely semantic. Of course, the previous one is a very naïve example, and I certainly believe that there are much more sophisticate manners that humans use in order to

express their stance. It is true though, if on the one hand, the importance of syntax has been widely studied in linguistic literature as a possible trigger for irony, the same can not be said for the task of stance. However, as the field and the possible existing connections between syntax and stance have not been thoroughly studied yet, I would not indulge any further in neglecting the role of syntactic structures in the study of language, and propose its study in the present thesis. Therefore, I decided to apply the same framework to both irony detection and stance detection, hoping to pave some light on the first (as attested in literature) as well on the second case study (not documented in literature), disregarding to appreciate solely the outcomes in terms of the numerical performance of automatic systems, but rather being more focused and interested in the more profound linguistic reasons behind them.

If we manage to understand what is the linguistic knowledge a certain approach leverages when it produces good (or poor) results, among many possible approaches, it could allow us to make more mature choices for following work. Indeed, the future of NLP research needs to go towards approaches that better integrate different types of knowledge (such as syntactic knowledge, for once) and that manage to be more versatile for certain types of data and in different application contexts.

1.4 Structure of the thesis

This thesis consists in a reorganized collection of the most relevant key points of investigations extracted from some research projects in which I was involved during my Ph.D. studies. The material has been inspired, reshaped and rewritten from 12 different papers published in the proceedings of international conferences and workshops such as *CLiC-it*, *SEPLN*, *DepLing*, *IberEval*, *IberLEF*, *SemEval*, *EVALITA*, *LREC* and *COLING* between 2017 and 2020. Some other work has been published or submitted to national and international journals such as *IJCoL* and *Computer, Speech & Language* and *Language Resources and Evaluation* in the last two years.

A brief overview of the contents of the thesis is presented below, with the reference to the publication linked to each chapter or section, summarizing all the work done and resuming the results obtained in the framework of this three-year-long research path. In Chapter 3, I also show some unpublished results, regarding stance detection with dependency syntax and neural networks. Finally, I draw some conclusions and discuss future work in the final chapter.

Chapter 1 – as we have seen, in this chapter I have introduced the reader to the main topics that will be discussed in the present thesis, starting with a broad description of Natural Language Processing and text classification, followed by an introduction on Universal Dependencies, morphology and syntax. I also proposed a brief discussion on the issues that can arise while applying the format of Universal Dependencies to social media data, mainly referring to the following works:

- Sanguinetti et al. [2020],
- Sanguinetti et al. [2021].

Chapter 2 – in the second chapter, which deals with the topic of irony detection, I describe several works regarding such topic:

- In Section 2.1.1, I mainly refer to Cignarella et al. [2018b] in order to describe the *IronITA 2018* shared task.
- In Section 2.1.2, I describe the creation of the multilayered corpus TWITTIRÒ-UD by citing Cignarella et al. [2017], Cignarella et al. [2018a], Cignarella et al. [2019a], Cignarella et al. [2019b] and Cignarella et al. [2019c].
- In Section 2.3, I finally describe some experiments performed in which I leverage morphosyntactic information for irony detection, mainly citing what was done in Cignarella and Bosco [2019] and in Cignarella et al. [2020a].

Chapter 3 – in the third chapter I will deal with the task of stance detection, pointing at the following works:

- In Section 3.1.1, I mainly describe Cignarella et al. [2020b] in order to present the *SardiStance 2020* shared task.
- In Section 3.2.1, I present the work done in Lai et al. [2017a], describing the participation in the *StanceCat 2017* shared task.
- In Section 3.3, I describe my participation in *RumorEval 2019* where I first apply a syntax-based approach to the task of stance detection, referring to Ghanem et al. [2019a].
- In Section 3.3.2, I present a completely new work, specifically done for this thesis where I present a BERT-based approach also leveraging morphosyntactic information.

Chapter 4 – in the fourth chapter I propose a new research in which I explore the interaction between irony and stance, by analyzing the *SardiStance* dataset.

Chapter 5 – in the last chapter, I finally summarize all the important lessons learned and propose new research directions for future work.

1.5 Contributions

Within the last three years, I successfully published several papers in scientific conferences regarding the topics of this dissertation, and also participated and organized shared task challenges on irony detection and stance detection. The list of the works divided by area of research (irony detection, stance detection, Universal Dependencies, and others), and it is also ordered chronologically. The letter "C" refers to papers presented at conferences and "J" refers to journal papers.

[irony detection]

C.1 Cignarella, A. T., Bosco, C., and Patti, V. (2017). **TWITTIRÒ: A social media corpus with a multi-layered annotation for irony**. In *Proceedings of the 4th Italian Conference on Computational Linguistics (CLiC-it 2017)*. CEUR-WS.org.

C.2 Cignarella, A. T., Frenda, S., Basile, V., Bosco, C., Patti, V., and Rosso, P. (2018). **Overview of the EVALITA 2018 task on irony detection in italian tweets (IronITA)**. In *Proceedings of the 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*. CEUR-WS.org.

C.3 Cignarella, A. T., Bosco, C., Patti, V., and Lai, M. (2018). **Application and Analysis of a Multi-layered Scheme for Irony on the Italian Twitter Corpus TWITTIRÒ**. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*. ELRA.

J.4 Cignarella, A. T., Bosco, C., Patti, V., and Lai, M. (2019). **TWITTIRO: an Italian Twitter Corpus with a Multi-layered Annotation for Irony**. *Italian Journal of Computational Linguistics (IJCoL)*.

C.5 Cignarella, A. T., Sanguinetti, M., Bosco, C., and Rosso, P. (2019). **Is This an Effective Way to Annotate Irony Activators?** In *Proceedings of the 6th Italian Conference on Computational Linguistics (CLiC-it 2019)*. CEUR-WS.org.

C.6 Cignarella, A. T., and Bosco, C. (2019). **ATC at IroSva 2019: Shallow syntactic dependency-based features for irony detection in Spanish variants**. In *Proceedings of the 35th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019)*. CEUR-WS.org.

C.7 Cignarella, A. T., Sanguinetti, M., Bosco, C., and Paolo, R. (2020). **Marking Irony Activators in a Universal Dependencies Treebank: The Case of an Italian Twitter Corpus**. In *Proceedings of 12th Conference on Language Resources and Evaluation (LREC 2020)*. ELRA.

C.8 Cignarella, A. T., Basile, V. Sanguinetti, M., Bosco, C., Rosso, P., and Benamara F. (2020). **Multilingual Irony Detection with Dependency Syntax and Neural Models**. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*. ACL.

[stance detection]

C.9 Lai, M., Cignarella, A. T., Hernàndez Farìas, D. H, (2017). **ITACOS at IberEval 2017: Detecting Stance in Catalan and Spanish tweets**. In *Proceedings of the 2nd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2017)*. CEUR-WS.org.

C.10 Ghanem, B., Cignarella, A. T., Bosco, C., Rosso, P., and Pardo Rangel, F. M. (2019,). **UPV-28-UNITO at SemEval-2019 Task 7: Exploiting Post's Nesting and Syntax Information for Rumor Stance Classification**. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval 2019)* ACL.

J.11 Lai, M., Cignarella, A. T., Farías, D. I. H., Bosco, C., Patti, V., and Rosso, P. (2020). **Multilingual stance detection in social media political debates**. *Computer Speech and Language*.

C.12 Cignarella, A. T., Lai, M., Bosco, C., Patti, V., and Rosso, P. (2020). **SardiStance@EVALITA2020: Overview of the Task on Stance Detection in Italian Tweets**. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org.

[*Universal Dependencies*]

C.13 Cignarella, A. T., Bosco, C., and Rosso, P. (2019). Presenting **TWITTIRO-UD: An Italian Twitter Treebank in Universal Dependencies**. In *Proceedings of the 5th International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*. ACL.

C.14 Sanguinetti, M., Bosco, C., Cassidy, L., Çetinoğlu, Ö., Cignarella, A. T., Lynn, T., Rehbein, I., Ruppenhofer, J., Seddah, D., Zeldes, A. (2020). **Treebanking User-Generated Content: A Proposal for a Unified Representation in Universal Dependencies.** In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*. ELRA.

J.15 Sanguinetti, M., Bosco, C., Cassidy, L., Çetinoğlu, Ö., Cignarella, A. T., Lynn, T., Rehbein, I., Ruppenhofer, J., Seddah, D., Zeldes, A. (2021). **Treebanking User-Generated Content: a UD Based Overview of Guidelines, Corpora and Unified Recommendations**. *Accepted with minor revisions at Language Resources and Evaluation Journal, special issue Special Issue "Annotation of non-standard corpora".*

[others]

C.16 Pamungkas, E. W., Cignarella, A. T., Basile, V., and Patti, V. (2018). **Automatic Identification of Misogyny in English and Italian Tweets at EVALITA 2018 with a Multilingual Hate Lexicon**. In *Proceedings of the 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*. CEUR-WS.org.

C.17 Pamungkas, E. W., Cignarella, A. T., Basile, V., and Patti, V. (2018). **14-ExLab@UniTo for AMI at IberEval2018: Exploiting Lexical Knowledge for Detecting Misogyny in English and Spanish Tweets**. In *Proceedings of the 3rd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018)*. CEUR-WS.org.

C.18 Pellegrini, M. and Cignarella, A. T. (2020). **(Stem and Word) Predictability in Italian verb paradigms: An Entropy-Based Study Exploiting the New Resource *LeFFI***. In *Proceedings of the 7th Italian Conference on Computational Linguistics (CLiC-it 2020)*. CEUR-WS.org.

Chapter 2

Irony detection

---

irony

noun [U]

UK /ˈaɪ.rə.ni/ US /ˈaɪ.rə.ni/

THE USE OF WORDS THAT ARE THE OPPOSITE OF WHAT YOU MEAN, AS A WAY OF BEING FUNNY

(CAMBRIDGE DICTIONARY)

In this chapter I will focus on the problem of Irony Detection, that I took into account as first case study, in order to investigate the impact of morphosyntactic information in Sentiment Analysis tasks.

Firstly, I will survey the related work on this topic, devoting particular attention to the description of evaluation campaigns and shared tasks that have been organized in the last few years, stressing the importance of freely available annotated datasets and benchmarks. Later on, in the same section I will provide a broad panorama of approaches and machine learning techniques that are typically used in this field, naming the innovative approaches exploited by participants of the above mentioned shared tasks and outlining the state-of-the-art models.

After having stressed the significant impact of shared tasks and evaluation campaigns to the development of resources and the creation of evaluation benchmarks in different languages, in the second section I will present an overview of the IronITA 2018 shared task (*Irony Detection in Italian tweets*) [Cignarella et al., 2018b], which we organized together with other colleagues within the EVALITA 2018 evaluation campaign for Italian NLP tools and resources.[1]

Thirdly, in a following section, I will focus on the description of the TWITTIRÒ corpus [Cignarella et al., 2017, 2018a], that I developed within my PhD studies, and that in part has also been distributed for training participating systems in the above-

[1] http://www.evalita.it/.

mentioned shared task on *Irony Detection in Italian tweets*. This section should serve as a detailed zoom on the issues that may arise while dealing with a complicated pragmatic phenomenon such as that of irony, especially within the delicate process of corpus creation which entails both annotation and negotiation among human annotators. Starting from a nucleus of some hundreds of tweets collected during the development of my master thesis [Cignarella, 2017], I have later on enhanced the annotations of this corpus within the duration of my PhD up to now, i.e., the TWITTIRÒ-UD treebank, which is my contribution to the Universal Dependencies project [Cignarella et al., 2019b, 2020c].

My direct involvement in the organization of a shared task and in the development of a novel resource allowed a meaningful improvement of my awareness of the importance of formalizing the irony detection problem, turning it into a computational task, also paving the way for a deeper understanding of the major approaches applied to irony detection and for my participation in some shared task.

Lastly, in a more technical section, I will indeed provide the implementation details and evaluation of machine learning systems that exploit morphosyntactic knowledge by taking advantage from the Universal Dependencies annotation format. I will describe my participation in the IroSVA 2019 irony detection shared task (*Irony Detection in Spanish Variants*) [Cignarella and Bosco, 2019] and present the first introductory work that moves in the direction of a syntax-based approach for dealing with irony. Following this, I will comment on further implementations of the above-mentioned system [Cignarella et al., 2020a]. I will provide more accurate descriptions of feature engineering and system development, together with a wider scenario of results and discussion points regarding primarily the impact of syntax on the task of irony detection, which is one of the contributions that I present with this thesis.

I conclude this section with some remarks about irony and the motivation for investigating this phenomenon from a computational perspective. The identification of irony and the description of linguistic devices that contribute to its production are known to be very controversial topics [Grice, 1975, Sperber and Wilson, 1981, Utsumi, 1996]. Understanding an ironic production is certainly difficult for humans, and also in the domain of Sentiment Analysis, the detection of irony is among the tasks currently considered as especially challenging, since the presence of an ironic intention in a text can change the polarity of the opinion expressed to its opposite, i.e., using positive words for intending a negative meaning or – less often – the other way around [Grice, 1978, Attardo, 2000, Barbieri and Saggion, 2014].

This characteristic of irony can significantly undermine automatic systems' accuracy in Sentiment Analysis and makes it crucial to develop irony-aware systems. In fact, the area of Irony Detection has been a great focus of the NLP community in early years [Mihalcea and Pulman, 2007, Reyes et al., 2010, Kouloumpis et al., 2011, Maynard and Funk, 2011, Reyes et al., 2012, Bosco et al., 2013, Reyes et al., 2013, Riloff et al., 2013, Wang, 2013, Barbieri et al., 2014, Hernández Farías et al., 2015, Joshi et al., 2015], and still nowadays it proves to be a particularly challenging and interesting task, which is furthermore confirmed by the many scientific publications that are proposed every year about it [Hernańdez Farías et al., 2016, Joshi et al., 2017, Khodak et al., 2018, Hazarika

et al., 2018, Zhang and Abdul-Mageed, 2019, Zhang et al., 2019, González et al., 2020]. Additionally, the challenge posed to automatic systems may be also further complex when irony is produced in co-occurrence with sarcasm or satire [Hernández Farías and Rosso, 2016, Joshi et al., 2017, Ravi and Ravi, 2017].

Despite a complete consensus on the definition of what the term *irony* might or might not include beneath it, several attempts have been made in the research community, in order to clarify such definitions. For instance, Joshi et al. [2017] in their survey report all the most influential points of view on the subject. However, it is worthwhile to stress that, considering the majority of state-of-the-art studies in computational linguistics, the term *irony* is mostly used as an umbrella term which includes satire, sarcasm and parody due to fuzzy boundaries among them [Giora, 1995, Attardo, 2000, Joshi et al., 2017]. In the present thesis, I will only focus on irony and briefly on sarcasm, from a computational perspective, without entering in a deeper discussion on the boundaries on the definitions of other phenomena such as humor or satire.

As a result, within the last fifteen years, the amount of irony-annotated resources and the organization of shared tasks regarding figurative language processing (among which, irony and sarcasm) for an increasing amount of different languages has considerably grown. This encouraged the subsequent comparison across languages, also considering that cultural differences may be detected beyond irony.

In the next sections I will thus list the shared tasks that have been organized (Section 2.1) and I will provide a broad panorama of approaches and machine learning techniques used in this field outlining the state-of-the-art models (Section 2.2).

2.1 Shared tasks and corpora for irony detection

Shared tasks and evaluation campaigns are generally organized by scientific communities to tackle specific problems which, for several reasons, are considered to be especially challenging to be addressed.

Natural Language Processing is one of the fields with a very long tradition of organizing challenges that helped to foster research in this domain, to the point where plenty of technological advancements were reached precisely thanks to shared tasks' organization.

The major effect of the organization of a shared task is to provide an evaluation benchmark on which the research community can test and comparatively evaluate models and approaches. When the task is thought to be addressed by using machine learning techniques and supervised approaches, like in the case of Irony Detection, annotated datasets must be created and made available not only for testing but also for training tools, and this can be a very expensive activity. It is surely time-consuming to gather texts that are ironic, whether they might be from blogging platforms, social media or online fora. It is even harder to select and annotate them with a common framework and reach a human consensus upon the fact that they actually present signs of irony or generic humor. Also the amount of data matters: in the era of deep learning it is not sufficient to provide hundreds of labelled data, it is rather necessary to work with thousands or millions of instances. This motivates the joint work of different research

groups and institutions in the organization of shared tasks, where each one can contribute at some level: providing experience, gathering data and labelling them. For taking full advantage of this effort, almost always, at the end of the challenge, the data become publicly available for the research community as a whole, from academia to industry.

The increasing interest towards the automatic detection of irony is clearly attested by the broad proposal of shared tasks focusing on this topic within NLP evaluation campaigns in several languages, within the last years. Below, I will outline the most significant events that took place regarding figurative language, irony, humor, satire and sarcasm, following a chronological order. On the other hand, in Table 2.1 I summarize the most relevant information regarding the shared tasks mentioned in this section, grouping them by language.

language	dataset	focus	source
English	SemEval-2015 Task 11 [Ghosh et al., 2015]	figurative language	Twitter
	SemEval-2018 Task 3 [Van Hee et al., 2018a]	irony and different types	Twitter
	Sarcasm detection at PAKDD[2]	sarcasm	Reddit
	Sarcasm Target Identification [Mollá and Joshi, 2019]	target of sarcasm	short texts
	Sarcasm Detection [Ghosh et al., 2020]	sarcasm and verbal irony	Twitter, Reddit
French	DEFT 2017 [Benamara et al., 2017]	figurative language and irony	Twitter
Spanish	IroSvA 2019 [Ortega et al., 2019]	irony and language variants	Twitter, news
Italian	SENTIPOLC 2014 [Basile et al., 2014]	sentiment analysis and irony	Twitter
	SENTIPOLC 2016 [Barbieri et al., 2016]	sentiment analysis and irony	Twitter
	IronITA 2018 [Cignarella et al., 2018b]	irony and sarcasm	Twitter
Arabic	IDAT at FIRE [Ghanem et al., 2019b]	irony	Twitter

Table 2.1: Shared tasks and datasets.

The first pilot task on irony detection was proposed from the Italian NLP community within SENTIPOLC (*SENTIment POLarity Classification*) within the context of EVALITA[3] held in 2014 [Basile et al., 2014]. The task was divided into three subtasks with an increasing level of complexity. The first two tasks were standard SA tasks (subjectivity and polarity classification), whereas the third one was precisely aimed at studying the presence of irony in tweets.

On a related note, it is worth mentioning that in the same year, at *SemEval-2014 Task 9* on Sentiment Analysis in Twitter was organized. In that venue, the organizers started to speculate that the presence of figurative language such as irony and sarcasm might indeed have an effect on sentiment analysis, therefore they introduce sarcastic tweets in the test set to study further on this matter.

The year after, a first task taking into account the presence of irony was organized at *SemEval-2015* on English data, focusing on figurative language in Twitter (*SemEval-2015 Task 11: Sentiment Analysis of Figurative Language in Twitter*) [Ghosh et al., 2015]. Specifically, three broad classes of figurative language were considered: irony, sarcasm

[2]Pacific Asia Knowledge Discovery and Data Mining Conference: https://pakdd16.wordpress.fos.auckland.ac.nz/technical-program/contests/.

[3]Evaluation Campaign of Natural Language Processing and Speech Tools for the Italian language: http://www.evalita.it/.

and metaphor. The task concerned itself with the classification of overall sentiment in tweets. It seemed, then, it was already clear that the presence of irony and sarcasm, which are typically used to criticize or to mock, was not enough for a system to simply determine whether the sentiment of a given tweet is positive or negative. So, given a set of tweets that are rich in metaphor, sarcasm and irony, the goal is to determine whether a user has expressed a positive, negative or neutral sentiment in each, and the degree to which this sentiment has been communicated.

Also in *SemEval-2015 Task 10* on *Sentiment Analysis in Twitter*, being the follow-up of the edition of the previous year, the organizers proposed a task in which the presence of sarcasm was considered in relation to SA, in order to measure its impact.

In 2016, after the success obtained two years before, the organizers of SENTIPOLC proposed the follow up of the 2014 shared task on Italian, enhancing the size of the dataset, and repeating it in the EVALITA 2016 evaluation campaign [Barbieri et al., 2016]. Once again, the main goal was the classification of the polarity of sentiment at message level in Italian tweets, and in one of the three subtasks, participants were also asked to classify posts depending on the presence of irony.

In the same year, 2016, there has also been a data science contest organized as a part of the *Pacific Asia Knowledge Discovery and Data Mining Conference* (PAKDD)[2] on the English language: "*Sarcasm detection on Reddit comments*". In this competition the dataset provided consisted of Reddit comments labeled as either sarcastic or non-sarcastic, and the aim was to propose an automatic system for classifying them.

A year after, in 2017, the interest towards figurative language animated also the French NLP community, in fact, a sentiment analysis task related to the ones mentioned above was proposed also for French: the *Défi Fouille de Texte* (DEFT) at the *2017 Conférence sur le Traitement Automatique des Langues Naturelles* (TALN)[4] [Benamara et al., 2017]. For the first time, the interest was leaning towards the influence of figurative language (particularly irony, sarcasm and humor) in the analysis of opinions from tweets in French. Three tasks of increasing levels of complexity were offered to participants: (T1) to determine the global polarity of non-figurative tweets, (T2) to determine whether a tweet contains figurative language, and (T3) to determine the overall polarity of the figurative and non-figurative tweets.

In 2018, a shared task completely dedicated to irony detection in English tweets, has been proposed at SemEval (*Task 3: Irony Detection in English Tweets*) [Van Hee et al., 2018a]. The organizers proposed two different exercises: in the first one (Task A), given a tweet systems had to determine whether the tweet was ironic or not, and in the second (Task B), if irony was indeed detected, systems had to determine which type of irony was expressed (i.e., verbal irony by means of a polarity clash, situational irony, or another type of verbal irony). The organization of this shared task, started a new interest towards irony, its manifestation and an encouragement of studying it in a deeper and finer-grained manner.

The same year, following the footsteps of our predecessors (SENTIPOLC 2014 [Basile et al., 2014] and 2016 [Barbieri et al., 2016]) and also following the recent trends of the

[4] https://taln2017.cnrs.fr/.

NLP community, I participated in the organization of a new shared task specifically oriented towards fine-grained irony detection in Italian tweets and we proposed it at EVALITA 2018: *(IronITA: Irony Detection in Italian Tweets)* [Cignarella et al., 2018b]. Also in this case, the task was articulated in a first exercise to determine whether a tweet was ironic or not, and a second sub-task in which a tweet previously considered ironic should be deemed either sarcastic or not. Note that, in this case, sarcasm was considered as a specific form of irony, meaning that a tweet could not be sarcastic if it was previously classified as non-ironic. In fact, we defined sarcasm as *a kind of sharp, explicit and sometimes aggressive irony, aimed at hitting a specific target to hurt or criticize without excluding the possibility of having fun* following the prescriptions of Du Marsais et al. [1981] and Gibbs [2000].

Again, following the interest for figurative language in English, in 2019, a shared task regarding *Sarcasm Target Identification* was proposed at the *Australasian Language Technology Association Workshop* (ALTA) [Mollá and Joshi, 2019], underlining the importance of target identification with relation to sarcastic productions.

The most recent shared task about irony in social media has been organized for Spanish language at the *Iberian Languages Evaluation Forum* (IberLEF)[5] in 2019: *Irony Detection in Spanish Variants (IroSvA 2019)*, exploring the differences among varieties of Spanish spoken in Spain, Cuba and Mexico [Ortega et al., 2019]. The datasets included both tweets and online news comments. Moreover, the messages are not considered as isolated texts but together with a given context (e.g., a headline or a topic). The organizers proposed three sub-tasks, one for each variant of Spanish, and after the evaluation window closed, they also encouraged participants to submit system's output obtained by training in a cross-language scenario.

As support to the hypothesis that the research community is more and more interested in figurative language and all its different facets, in the same year and workshop, also the HAHA shared task (*Humor Analysis based on Human Annotation*) was presented [Chiruzzo et al., 2019]. The challenge consisted in detecting whether a Spanish tweet was humorous or not and also providing a '*funniness*' score. The fact that in the same year, at the same evaluation campaign, two tasks on figurative language were presented (one on irony and one on humour), furthermore stresses the importance to distinguish these two phenomena and the attention that the research community is starting to dedicate to them, separately.

Till nowadays, to the best of my knowledge, the only shared task regarding the topic of irony detection in a non-European language is the one organized in 2019 at FIRE (*Forum for Information Retrieval Evaluation*): *Irony Detection in Arabic Tweets* (IDAT 2019) [Ghanem et al., 2019b]. The task consisted of a binary classification of tweets as either ironic or not ironic using a dataset composed of 5,030 tweets written in Arabic, dealing with different political issues and events related to the Middle East and the Maghreb. Data were written in Modern Standard Arabic but also in different Arabic language varieties including Egypt, Gulf, Levantine and Maghrebi dialects.

To conclude, the most recent shared task regarding these topics, was organized within

[5] https://sites.google.com/view/iberlef-2019.

the *Second Workshop on Figurative Language Processing* (FigLang 2020) at ACL 2020: *The Shared Task on Sarcasm Detection* [Ghosh et al., 2020]. In this competition, the term *sarcasm* refers to both sarcasm and verbal irony which are, in this case, considered as synonyms. This last contest, similarly to the one organized at ALTA in 2019, leveraged a particular interest for the context in which sarcasm is uttered (i.e., the entire prior conversation context). Secondly, the dataset was created by texts automatically extracted from two different types of social media platforms: Twitter and Reddit. In fact, both of these platforms allow the writers to mark whether their messages are sarcastic (e.g., #sarcasm hashtag in Twitter and "/s" marker in Reddit).

It is important to stress the importance of the hard work of computational linguists and skilled annotators for the creation of labelled datasets, without whom progress in terms of computational advancements would be very unlikely to be reached. Indeed, the whole process of creating a dataset for a shared task consists of various steps that are rather delicate and highly time-consuming: data collection, data selection, data annotation, data validation.

For creating a dataset to be distributed and used as a ground truth in an experimental setting, researchers could either rely on experts that are internal to their research group or rely on crowdsourcing platforms. Either way, the organizers of the above mentioned shared tasks were able to release datasets whose sizes varied from ca. 3,000 to ca. 12,000 instances. Whether the amount of data is *enough* to successfully serve its purposes is still a matter under an ongoing debate. However a broader consensus is reached among researches dealing with neural architectures, where "the more, the merrier" is a common ground concerning data availability.

From this section we have learned the contribution that the organization of shared tasks gives to the NLP research community, both in terms of possibility of aggregation among researchers interested in the same topics, but also in terms of creation and development of annotated resources.

Still nowadays there is some uncertainty on the definition of humor, irony and sarcasm. However, throughout the last years, more and more researchers have tried to study each phenomenon as unique, trying to highlight their characteristics that make them stand out with respect to others. The trend in research on these topics shows that irony has resulted to be the most naïve kind of making fun of someone/something, while sarcasm also presents traits that are assimilable to a soft form of hate speech, that might be used to poke the recipient. Additionally, irony seems to differentiate from simple humor from the fact that it is often correlated also with a reversal of the meaning and sentiment of an utterance, whereas humor does not have this peculiarity.

In Section 2.1.1, to further argument on this matter, I will describe my personal experience as organizer of a shared task regarding irony detection in Italian tweets (*IronITA 2018*). In the following section (Section 2.2) I will describe the main approaches used in machine learning to tackle the problem of irony detection, many of which were indeed proposed within evaluation campaigns and quickly became state of the art.

2.1.1 Organization of *IronITA 2018*

After having stressed the significant impact of shared tasks to the development of resources and the creation of evaluation benchmarks in different languages, in this section it seems appropriate to present an overview of the IronITA 2018 shared task (*Irony Detection in Italian tweets*) [Cignarella et al., 2018b], which I organized together, with my supervisors and some colleagues from the Department of Computer Science of the *University of Turin* and from the *PRHLT research group* of the *Universitat Politècnica de València*, within the EVALITA 2018 evaluation campaign for Italian NLP tools and resources. By participating in the organization of this task I became more aware of the importance of providing a formal definition of the irony detection problem, turning it into a computational task. Therefore, in the present section I will outline the purpose of the task, describe the datasets used, and briefly survey the main approaches exploited for Italian, also commenting on the participating systems in the shared task.

The task consisted in automatically annotating messages from Twitter for irony and sarcasm. It was organized in a main task (Task A) centered on irony, and a second task (Task B) centered on sarcasm.

Task A was structured as a two-class (or binary) classification exercise where systems had to predict whether a tweet was ironic or not.

Task B consisted instead in a multi-class classification task where systems had to predict one out of the three following labels: **i) sarcasm**, **ii) irony not categorized as sarcasm** (i.e., other kinds of verbal irony or descriptions of situational irony which do not show the characteristics of sarcasm), and **iii) not-irony**.

By organizing these tasks we aimed at encouraging the investigation of irony, and also of sarcasm as a specific type of irony. Sarcasm has been recognized in Bowes and Katz [2011] with a specific target to attack [Attardo, 2000, Dynel, 2014], more offensive and delivered with a cutting tone (rarely ambiguous). According to Lee and Katz [1998] hearers perceive aggressiveness as the feature that mainly distinguishes sarcasm. We defined **sarcasm** as *a kind of sharp, explicit and sometimes aggressive irony, aimed at hitting a specific target to hurt or criticize without excluding the possibility of having fun* [Du Marsais et al., 1981, Gibbs, 2000]. The factors we have taken into account for the annotation are, the presence of:

1. a clear **target**
2. an obvious **intention** to hurt or criticize
3. **negativity** (weak or strong)
4. a **stereotype** or a common place
5. a blatant **intention** to do something against someone
6. **offensiveness** (weak or strong)
7. **aggressiveness** (weak or strong).

We have also tried to differentiate our concept of "sarcasm" from that of "satire", often present in tweets. For us, satire aims to ridicule the target as well as criticize it. Differently from sarcasm, satire is solely focused on a more negative type of criticism and moved by a personal and angry emotional charge.

Finally, by providing a dataset from social media (Twitter), we focus on texts especially hard to be dealt with, because of their shortness and because they will be analyzed out of the context where they were generated.

Training and test data. The data released for the shared task came from different source datasets, namely: Hate Speech Corpus (HSC) [Sanguinetti et al., 2018] and the TWITTIRÒ corpus [Cignarella et al., 2018a], which in turn is composed of tweets from LaBuonaScuola corpus (TW-BS) [Stranisci et al., 2016], Sentipolc corpus (TW-STPC) [Barbieri et al., 2016], Spinoza corpus (TW-SPINO) [Barbieri et al., 2016].

Test data were drawn from the same sources with the addition of some tweets from the TWITA collection, that were annotated by the organizers of the SENTIPOLC 2016 shared task, but were not exploited during the 2016 campaign [Barbieri et al., 2016]. In Table 2.2 are shown the different distributions of tweets in the training set and in the test set, together with the the labels for irony and sarcasm.

	TRAINING SET				TEST SET				
	IRO	NOT	SARC	NOT	IRO	NOT	SARC	NOT	**TOTAL**
TW-BS	467	646	173	294	111	161	51	60	
TW-SPINO	342	0	126	216	73	0	32	41	2,886
TW-STPC	461	625	143	318	0	0	0	0	
HSC	753	683	471	282	185	119	106	79	1,740
TWITA	0	0	0	0	67	156	28	39	223
TOTAL	**3,977**				**872**				**4,849**

Table 2.2: Distribution of tweets according to the topic.

Annotation of the datasets. The annotation process involved four Italian native speakers and focused only on the finer-grained annotation of sarcasm in the ironic tweets, since the presence of irony was already annotated in the source datasets [Basile et al., 2014, Barbieri et al., 2016]. It began by splitting in two halves the dataset and assigning the annotation task for each portion to a different couple of annotators. The first half is constituted by TWITTIRÒ, while the second half is constituted by HSC and TWITA.
In the following step, the final inter-annotator agreement (IAA) had been calculated on all the dataset through Fleiss' kappa coefficient. Then, in order to achieve an agreement on a larger portion of data, the effort of the annotators has been focused only on the cases of disagreement. In particular, the couple previously involved in the annotation of the first half of the corpus produced a new annotation for the tweets in disagreement of the second portion of the dataset, while the couple involved in the annotation of the second half of the corpus did the same on the first portion of the dataset. After that, the cases where the disagreement persisted have been discarded as too ambiguous to be classified (131 tweets). The final IAA calculated with Fleiss' kappa was $\kappa = 0.56$ for

[6]It's a remark on the usage of the split form of the complex preposition "della", which was used as "de la". The example is hard to translate in English.

irony	sarcasm	text
0	0	@SteGiannini @sdisponibile Semmai l'anno DELLA buona scuola. De la, in italiano, non esiste *@SteGiannini @sdisponibile At most it's the year OF good school. Of the, in Italian, doesn't exist*[6]
1	0	Di fronte a queste forme di terrorismo siamo tutti sulla stessa barca. A parte Briatore. Briatore ha la sua. *Facing these forms of terrorism we're all on the same boat. But Briatore. Briatore has its own.*
1	1	Anche oggi sono in arrivo 2000migranti dalla Libia avanti in italia ce posto per tutti vero @laura boldrini? Li puoi accogliere a casa tua *Also today 2000migrants arriving from Libya come on Italy theres space for everyone right @laura boldrini? You can host them in your house*

Table 2.3: Examples for each possible combination of labels.

the tweets belonging to the TWITTIRÒ corpus and $\kappa = 0.52$ for the data from the HSC corpus and it is considered *moderate*[7] and satisfying for the purpose of the shared task.

In this process the annotators relied on the specific definition of sarcasm provided above the following detailed guidelines.[8] Some examples of the annotation are provided in Table 2.3.

Participants and results. A total of 7 teams, both from academia and industry sector participated in at least one of the two tasks of IronITA. Table 2.4 provides an overview of the teams, their affiliation, and the tasks they took part in.

team name and report	institution	tasks
ItaliaNLP, [De Mattei et al., 2018]	ItaliaNLP group ILC-CNR	A,B
UNIBA, [Basile and Semeraro, 2018]	University of Bari	A
X2Check, [Di Rosa and Durante, 2018]	App2Check srl	A
UNITOR, [Santilli et al., 2018]	University of Roma "Tor Vergata"	A,B
Aspie96, [Giudice, 2018]	University of Torino	A,B
UO_IRO, [Ortega-Bueno and Medina Pagola, 2018]	CERPAMID and Havana University	A
venses-itgetarun, n.a.	Ca' Foscari University of Venice	A,B

Table 2.4: Participants

For evaluation and ranking, we implemented two straightforward baseline systems for the task:

- *baseline-MFC* (MostFrequentClass) assigned to each instance the majority class of the respective task, namely `not-ironic` for task A and `not-sarcastic` for task B.

[7] According to the parameters proposed by Fleiss [1971].
[8] `http://di.unito.it/definitionofsarcasm`.

- *baseline-random* assigned uniformly random values to the instances. Note that for task A, a class is assigned randomly to every instance, while for task B the classes are assigned randomly only to eligible tweets who are marked `ironic` in the previous task.

Table 2.5 shows the results for **Task A**, which attracted 17 total submissions from 7 different teams. The best scores are achieved by the ItaliaNLP team that, with a constrained run, obtained the best score for both the `ironic` and `not-ironic` class, thus obtaining the highest averaged F1-score of 0.731. Among the unconstrained systems, the best F1-score for the `not-ironic` class is achieved by the X2Check team with F = 0.708, and the best F1-score for the `ironic` class is obtained by the UNITOR team with F = 0.733. All participating systems show an improvement over the baselines, with the exception of the only unsupervised system (venses-itgetarun).

team name	**id**	**F1-score**		
		not-iro	**iro**	**macro**
ItaliaNLP	1	**0.707**	**0.754**	**0.731**
ItaliaNLP	2	0.693	0.733	0.713
UNIBA	1	0.689	0.730	0.710
UNIBA	2	0.689	0.730	0.710
X2Check	1	**0.708**	0.700	**0.704**
UNITOR	1	0.662	0.739	0.700
UNITOR	2	0.668	**0.733**	0.700
X2Check	2	0.700	0.689	0.695
Aspie96	1	0.668	0.722	0.695
X2Check	2	0.679	0.708	0.693
X2Check	1	0.674	0.693	0.683
UO_IRO	2	0.603	0.700	0.651
UO_IRO	1	0.626	0.665	0.646
UO_IRO	2	0.579	0.678	0.629
UO_IRO	1	0.652	0.577	0.614
baseline-random		0.503	0.506	0.505
venses-itgetarun	1	0.651	0.289	0.470
venses-itgetarun	2	0.645	0.195	0.420
baseline-MFC		0.668	0.000	0.334

Table 2.5: Results of Task A. Grey background marks unconstrained runs.

Table 2.6 shows the results for **Task B**, which attracted 7 total submissions from 4 different teams. The best scores are achieved by the UNITOR team that with an unconstrained run obtained the highest macro F1-score of 0.520.
Among the constrained systems, the best F1-score for the `not-ironic` class is achieved by the ItaliaNLP team with F1-score = 0.707, and the best F1-score for the `ironic` class is obtained by the Aspie96 team with F1-score = 0.438. The best score for the `sarcastic` class is obtained by a constrained run of the UNITOR team with F1-score = 0.459. The best performing UNITOR team is also the only team that participated in Task B with an unconstrained run. All participating systems show an improvement over the baselines, with the exception of the only unsupervised system (venses-itgetarun).

team name	id	F1-score			
		not-iro	iro	sarc	macro
UNITOR	2	**0.668**	**0.447**	**0.446**	**0.520**
UNITOR	1	0.662	0.432	**0.459**	**0.518**
ItaliaNLP	1	**0.707**	0.432	0.409	0.516
ItaliaNLP	2	0.693	0.423	0.392	0.503
Aspie96	1	0.668	**0.438**	0.289	0.465
baseline-random		0.503	0.266	0.242	0.337
venses-itgetarun	1	0.431	0.260	0.018	0.236
baseline-MFC		0.668	0.000	0.000	0.223
venses-itgetarun	2	0.413	0.183	0.000	0.199

Table 2.6: Results Task B. Unconstrained runs are marked by grey background.

According to the reports provided by the participants, the systems may be compared and described according to the following main dimensions: classification framework (approaches, algorithms, features), text representation strategy, use of additional annotated data for training, external resources (e.g., sentiment lexica, NLP tools, etc.). Noticeably, the system architectures employed in this task are highly varied and heterogeneous, as well as the features engineered and external resources used. To the purpose of the present thesis, it is interesting to highlight that no team made use of morphosyntactic features, let alone no one used *Universal Dependencies* as encoding format.

System architecture. Most submitted runs to IronITA were produced by supervised machine learning systems. In fact, all but one systems used were supervised, although the nature and complexity of their architectures varies significantly. UNIBA and UNITOR used Support Vector Machine (SVM) classifiers, with different parameter settings. UNITOR, in particular, employed a multiple kernel-based approach to create two SVM classifiers that work on the two tasks. X2Check used several models based on Multinomial Naive Bayes and SVM in a voting ensemble. Three systems implemented deep learning neural networks for the classification of irony and sarcasm. Sequence-learning networks were a popular choice, in the form of Bidirectional Long Short-term Memory Networks (used by ItaliaNLP and UO_IRO) and Gated Recurrent Units (Aspie96). The venses-itgetarun team proposed the only unsupervised system submitted to IronITA. The system was based on an extension of the ITGETARUN rule-based fully symbolic semantic parser [Delmonte, 2014]. The performance of the venses-itgetarun system was penalized mainly by its low recall (see the detailed results on the task website).

It is interesting to see how, still nowadays, in the era of deep learning, the results obtained with classical machine learning techniques such as SVMs are definitely comparable with the results obtained by deep learning architectures. It might be due, of course, to the nature and size of the dataset, but it seems then, that for an effective exploration of the irony detection task, it might prove useful to still investigate a variety of different models.

Features. In addition to explore a broad spectrum of supervised and unsupervised architectures, the submitted systems leverage different kinds of linguistic and semantic

information extracted from the tweets. Word n-grams of varying size are used by ItaliaNLP, UNIBA, and X2Check. Word embeddings were used as features by three systems, namely ItaliaNLP (built with word2vec on a concatenation of ItWaC[9] and a custom tweet corpus), UNITOR (built with word2vec on a custom Twitter corpus) and UNIBA (built with Random Indexing [Sahlgren, 2005]) on a subset of TWITA [Basile et al., 2018]. Affective lexicons were also employed to extract polarity-related features from the words in the tweets, by UNIBA, ItaliaNLP and UNITOR and UO_IRO (see the "Lexical Resources" section for details on the lexica). UNIBA and UO_IRO also computed sentiment variation and contrast in order to extract the ironic content from the text. Features derived from sentiment analysis are also employed by the unsupervised system venses-itgetarun. Aspie96 performs its classification based on the single characters of the tweet. Finally, a great number of other features is employed by the systems, including stylistic and structural features (UO_IRO), special tokens and emoticons (X2Check). As anticipated, no team made explicit use of morphosyntactic features. Although, two teams exploited lexical features such as n-grams and char-grams.

Lexical Resources. Several systems employed affective resources, mainly as a tool to compute the sentiment polarity of words in each tweet. ItaliaNLP used two affective lexica generated automatically by means of distant supervision and automatic translation. UNIBA used an automatic translation of SentiWordNet [Esuli and Sebastiani, 2006]. UNITOR used the Distributed Polarity Lexicon by Castellucci et al. [2016]. UO_IRO used the affective lexicon derived from the OpeNER project [Russo et al., 2016] and a polarity lexicon of emojis by Kralj Novak et al. [2015]. venses-itgetarun used several lexica, including some specifically built for ITGETARUNS and a translation of SentiWordNet as well.

Additional training data. Three teams took the opportunity to send unconstrained runs along with constrained runs. X2Check included in the unconstrained training set a balanced version of the SENTIPOLC 2016 dataset, Italian tweets annotated with irony [Barbieri et al., 2016]. UNITOR used for their unconstrained runs a dataset of 6,000 tweets obtained by distant supervision (searching for the hashtags #ironia and #irony). UO_IRO employed tweets annotated with fine-grained irony from TWITTIRÒ [Cignarella et al., 2018a], the resource I developed within the last three years, on which is focused Section 2.1.2 of the present thesis (see below).

Differently from previous sub-tasks on irony detection in Italian language proposed as part of the previous SENTIPOLC shared tasks [Basile et al., 2014, Barbieri et al., 2016], having Sentiment Analysis as reference framework, the IronITA tasks, for the first time for Italian, focused specifically on irony and sarcasm identification. By comparing the results for irony detection obtained within the SENTIPOLC sub-task (the best performing system in the 2016 edition reached $F = 0.5412$ and in 2014 $F = 0.575$) with the ones obtained in IronITA, it is worth to notice that a dedicated task on irony detection led to a remarkable improvement of the scores, with the highest value here being $F = 0.731$.

[9] https://www.sketchengine.eu/itwac-italian-corpus/

Evan more surprisingly, scores for Italian are in line with those obtained at SemEval 2018-Task3 on irony detection in English tweets [Van Hee et al., 2018a], even if a lower amount of linguistic resources is available for Italian than for English, especially in terms of affective lexica, a type of resource that is frequently exploited in this kind of task. Actually, some teams used resources provided by the Italian NLP community also in the framework of previous EVALITA's edition (e.g., additional information from annotated corpora as SENTIPOLC Barbieri et al. [2016] and HaSpeeDe Bosco et al. [2018]). The good results obtained in this edition can be read also as a confirmation that linguistic resources for Italian language are increasing in quantity and quality, and they are helpful also for a very challenging task as irony detection.

As a contributor of the *Universal Dependencies* project, in the following sections I will present another kind of linguistic resource that might prove useful in NLP tasks, which is focused on the annotation of morphosyntactic information, i.e., treebanks. In particular I will shed some light on the panorama of available morphosyntactic resources for Italian, also describing the creation of one of them, for which I was personally involved.

In IronITA, another interesting factor is the use of the fashionable deep learning techniques, mirroring the growing interest in deep learning by the NLP community at large. Indeed, the best performing system is based on a deep learning approach revealing its usefulness also for irony detection. The high performance of deep learning methods is an indication that irony and sarcasm are phenomena involving more complex features than n-grams and lexical polarity. This inspired the experiments will be presented later in this thesis.

Inspired by the approaches proposed in this shared task, and pairing it with my research interest towards morphology and syntax, the natural follow-up has been that of proposing an experimental setting where I present a new method based on dependency syntax for the detection of irony. In Section 2.3.2, in fact, I will present such setting, exploiting the training and test set created within this shared task as first case study, and I will compare the obtained results with the official rankings of Table 2.5.

2.1.2 The creation of an ironic corpus: TWITTIRÒ

For a great part of the time during my PhD studies I have been involved also in the creation of a corpus annotated for irony. In its origins, was thought to become the primary source for conducting experiments and to carry out an in-depth study regarding irony detection also using novel features with respect to state-of-the-art systems, such as those that participated in the shared tasks I described above.

The availability of resources, and in particular annotated corpora, is crucial for the advancement in this area, even if it involves a very time-consuming activity and the work of several people. In this section, I will therefore focus on the description of the TWITTIRÒ corpus that I started to develop during the work conducted for my master thesis [Cignarella, 2017], and I continued to further enhance within the duration of my PhD studies [Cignarella et al., 2017, 2018a]. The corpus has also been distributed for training participating systems in the shared task on *IronITA 2018: Irony Detection in*

Italian tweets mentioned in Section 2.1.1. As a consequence of being exploited as part of the training data in a shared task , the corpus received feedback from the participants on a variety of levels, it has undergone many phases of correction, and nowadays it can be considered a solid benchmark in the field of irony detection. Later on, after the evaluation phase and the shared task event were over, I enhanced the available annotations by adding a new layer containing morphosyntactic information which allowed the development of the syntax-based approach for sentiment analysis tasks, i.e., the achievement of the main goal of this thesis. The corpus has been converted to the CoNLL-U format[10] adopted by the Universal Dependencies framework, thus resulting in the TWITTIRÒ-UD treebank, which is my main contribution to the Universal Dependencies project [Cignarella et al., 2019b].

In the most recent works, I tried indeed to capture the relation between the two layers of annotation mentioned above: the pragmatic/semantic information (irony) and morphosyntactic information (UD). I annotated indeed the so-called *irony activators* (i.e., the specific cue words that serve as triggers in the production of irony) by taking advantage of the structure of the CoNLL-U format imposed by the UD framework, which allowed to annotate relevant words at a token level [Cignarella et al., 2019c, 2020c]. The idea of annotating such cue words, comes from previous work of Karoui et al. [2017] in which the authors proposed an annotation scheme on four different layers where the last level consisted in the lexical annotation of irony "triggers", which was mainly done in an automatic fashion but was never carried on and deepened as study. In this regard, my work was done in a more extended way, taking advantage of the fact that the ironic corpus TWITTIRÒ has also been enriched with tokenization, lemmatization, part-of-speech and dependency relation labels. The application of the UD format to the TWITTIRÒ corpus, in fact, allowed to find and tag those cue words in a systematic way (see following section).

This section should serve as a detailed zoom in on the issues that may arise while dealing with a complex pragmatic phenomenon such as that of irony, especially within the delicate process of the creation of a multi-layered corpus, which entails annotation, discussion among human annotators and correction. Following, I will describe the process that led to the annotation of the 'sentiment analysis' layer, the 'morphosyntactic' layer, and the annotation of 'ironic cues'.

Composition of the corpus. Based on previous research oriented towards the creation of corpora annotated accordingly to irony, I decided to follow similar approaches in the collection of data. In particular, I have been greatly inspired by the work of Karoui et al. [2015], who not only worked on irony detection, but also proposed guidelines for a multi-layered scheme of annotation validating its efficacy on datasets in three different languages: English, French and Italian [Karoui et al., 2017]. In order to collect the French and English datasets the Twitter APIs were used, by filtering tweets through specific *hashtags* exploited by users to self-mark their ironic intention (*#irony*, *#sarcasm*, *#sarcastic*). They explained that this technique could not be applied for Italian because although Italian users do exploit a series of humorous hashtags, no long-term single hashtag is established and shared among them (such as *#irony* for English). Nevertheless,

[10] https://universaldependencies.org/format.html.

in the last few years several Italian corpora from Twitter (SENTI-TUT [Bosco et al., 2013], SENTIPOLC 2014 [Basile et al., 2014]), and LA BUONA SCUOLA CORPUS [Stranisci et al., 2016, 2015]) where the presence of irony is marked, have been made available. From those I could extract the tweets to include in the corpus I created (TWITTIRÒ). The origin from each source corpus is detailed in Table 2.7.

source corpus	number of tweets
TW-SPINO	378
TW-STPC14	527
TW-BS	519
TWITTIRÒ	**1,424**

Table 2.7: Origin of TWITTIRÒ's tweets from the different sources.

• **TW-SPINO** is a portion of SENTI-TUT [Bosco et al., 2013] which contains tweets collected from the satirical blog *Spinoza.it*.[11] The language used is grammatically correct and featured by a high register and style, while the topics are varied with a clear preference for jokes concerning the world of politics and general news.

Ex.19 Pubblicata la classifica mondiale della libertà di stampa. Non possiamo dirvi altro. [giga]
The world ranking for freedom of printing competition has been published. We can not say anything else. [giga]

• **TW-STPC14** is a portion of SENTIPOLC 2014 [Basile et al., 2014] and contains tweets generated by common users and, therefore, it is less homogeneous than TW-SPINO, with a frequent use of creative hashtags, mentions, repetitions of laughters. We selected here the political tweets with reference to the government of Monti between 2011 and 2012.

Ex.20 Mario Monti? non era il nome di un antipasto? #FullMonti #laresadeiconti #elezioni #308.
Mario Monti? Wasn't it the name of an entree? #FullMonti #laresadeiconti #elezioni #308.

• **TW-BS** is a portion of LABUONASCUOLA CORPUS [Stranisci et al., 2015, 2016] and it contains tweets on the debate of the reform of Italian School "Buona Scuola". Devices typically exploited in computer-mediated communication are shown and, being the reform of the education system a highly criticized one, the use of sentences written in ALL CAPS (to decode shouting) is wide.

Ex.21 @fattoquotidiano Quest'anno è peggio del solito: oltre all'amianto c'è anche #labuonascuola.
@fattoquotidiano This year is worse than usual: in addition to asbestos there is also #labuonascuola.

[11] http://www.spinoza.it/.

The different linguistic styles and composition of the three sub-corpora TW-SPINO, TW-STPC14, and TW-BS make TWITTIRÒ heterogeneous and also reveal interesting research paths to be followed, as I will address in the following sections.

2.1.2.1 The annotation scheme of TWITTIRÒ

As anticipated, this part of my work has been greatly inspired by the previous work of Karoui et al. [2017]. In particular, the authors, working on irony detection as well, proposed an annotation scheme based on four different layers, dedicated at annotating the phenomenon on irony in all its various levels of articulation (see Figure 2.1).

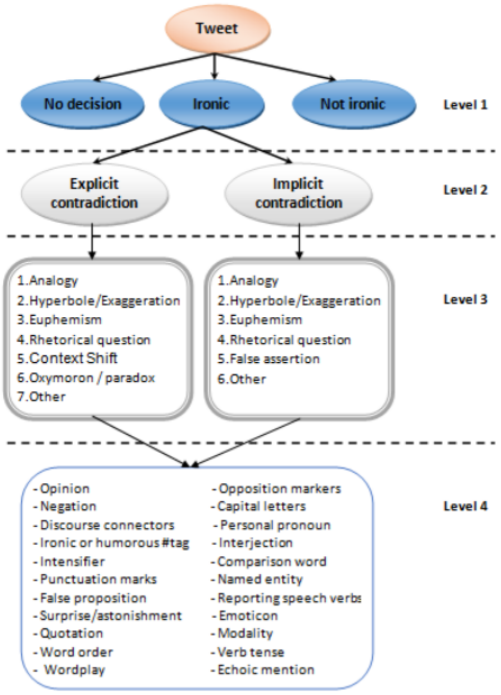


Figure 2.1: Annotation scheme for irony in tweets Karoui [2017].

Therefore, in this section I will describe how I applied Karoui's scheme to TWITTIRÒ and how I also went further with my investigation. First, I produced a complete morphosyntactic annotation (in UD format) which also delves into level 4 of the scheme, and secondly, I annotated the triggers of irony on the parse trees by delving into level 2 and 3.

Application of the multi-layered annotation scheme to TWITTIRÒ

As previously mentioned in the sections above, as the sources from which I derived the TWITTIRÒ corpus were already provided with a binary annotation for the presence of irony, my first aim - once collected the corpus - has been that of applying a fine-grained layer of annotation.

I decided to apply to the corpus the multi-layered annotation scheme based on the work of Karoui et al. [2015], because it had already been applied to English and French ironic tweets and had also been successfully applied to a smaller portion of Italian ironic tweets [Karoui et al., 2017]. The main advantage of the scheme is that it guarantees a representation of irony inspired by the issues raised in linguistic literature about this topic. For achieving this goal, the scheme includes four different levels of annotation organized as follows.

LEVEL 1: CLASS.

The first level concerns the classification of tweets into **ironic** or **not ironic**, but it does not apply in principle to my case where the corpus only includes ironic tweets because of the methodology applied in collection.

LEVEL 2: ACTIVATION TYPE.

As stated from various linguistic theories [Grice, 1975, Sperber and Wilson, 1981, Clark and Gerrig, 1984], irony is often exhibited through the presence of a clash or a contradiction between two elements. In tweets, these elements, henceforth named P1 and P2, can be found both as two lexicalized clues belonging to the internal context or can be one in the utterance and the other outside, as part of some pragmatic context external to the tweet.

According to [Karoui et al., 2015], I annotate the **activation type** such that, if the contradiction relies exclusively on the lexical clues internal to the utterance, as EXPLICIT, while if the contradiction that combines lexical clues with an additional pragmatic context external to the utterance, as IMPLICIT.

Explicit contradiction:
It can involve a contradiction between proposition P1 and proposition P2 that have e.g., opposite polarities, like in the example below where the opposition is between *liberate* (freed) and *processate* (processed).

Ex.22 [**Liberate**]$_{P1}$ Greta e Vanessa. Saranno [**processate**]$_{P2}$ in Italia. [@maurizioneri79]
Greta and Vanessa have been [***freed***]$_{P1}$. *They will* [***undergo trial***]$_{P2}$ *in Italy. [@maurizioneri79].*

Implicit contradiction:
Irony occurs because the writer believes that his audience can detect the disparity between P1 and P2 on the basis of contextual knowledge or common background shared with the writer.

Ex.23 ["Se davvero abbiamo pagato è uno schifo" ha detto Salvini guardando la laurea di Renzo Bossi.]$_{P1}$ [faro]
"It's a shame that we really paid for this" said Salvini, looking at Renzo Bossi's Masters degree. [faro]
$\rightarrow P_2$: Renzo Bossi got his Master's degree by paying with his father's party money (Lega Nord).

There are cases in which irony is activated in multiple ways inside a tweet. It might occur that on one superficial layer irony is explicitly activated from lexicalized cue words, and on a second "hidden" layer there is a deeper level of irony, inferable only through additional pragmatic knowledge. In this case, the tweet has to be annotated as ironic and the activation type as IMPLICIT.

LEVEL 3: CATEGORIES.

Both forms of contradictions can be expressed through different rhetorical devices, patterns or features that are grouped under different labels (i.e., analogy, euphemism, false assertion, oxymoron/paradox, context shift, hyperbole, rhetorical question and other).

Analogy:

In this category are summoned also other figures of speech that comprehend mechanisms of comparison, such as *simile* and *metaphor*. In (Ex.24) an analogy is drawn between the footballer Lionel Messi and the Italian minister Maria Elena Boschi because of their authoritarian fathers.

Ex.24 Leo Messi: "Firmo quello che mi dice papà". Pure la Boschi. [notturnoconcertante]
Leo Messi: "I sign what daddy tells me". Also Minister Boschi. [notturnoconcertante]

Euphemism:

It is a figure of speech which is used to reduce the facts of an expression or an idea considered unpleasant in order to soften the reality. In (Ex.25) is used the common device of punctuation such as quotations to soften one's way to express their own opinion.

Ex.25 "“" buona scuola "”".
"“" *good school* "”"

False assertion (implicit only):

Indicates that a proposition, fact or an assertion fails to make sense against the reality. The speaker expresses the opposite of what he thinks or something wrong with respect to a context. External knowledge is fundamental to understand irony (it is, in fact, implicit only). In the following example, the sentences written in correspondence of the right arrow are propositions, which are not lexicalized, but must be inferred from the reader to understand irony.

Ex.26 Vedo che c'è molta disinformazione sul referendum del 17 maggio. [@MisterDonnie13]
I see there is a lot of misinformation on the referenum of May, 17th. [@MisterDonnie13]
$\longrightarrow$ The referendum, in fact, was on April, 17th, not May.

Oxymoron/paradox (explicit only):

This category is equivalent to the category FALSE ASSERTION except that the contradiction, this time, is explicit. Also in this category, the sentences written in correspondence of the right arrow are propositions, which are not lexicalized, but must be inferred from the reader to understand irony.

Ex.27 Legge elettorale, il Pd si divide. Non vedono l'ora di provarla. [@maurofodaroni]
Electoral law, the Democratic Party is divided. They can't wait to try it. [@maurofodaroni]
$\longrightarrow$ It is absurd to think that the Italian political party PD has undergone an internal division in order to try a new electoral law they promoted.

Context shift (explicit only):
It occurs by the sudden change of the topic/frame in the tweet, as in (Ex.28), where the first clue word is about a Roma encampment, while the second about a safari journey.

Ex.28 L'auto di Salvini assalita al **campo rom**. Rovinato il **safari**. [@paniruro]
Salvini's car assaulted at the Roma camp. The safari is runined. [@paniruro]

Hyperbole/exaggeration:
It is a figure of speech which consists in expressing an idea or a feeling with an exaggerated way. It can be expressed either through the use of superlative adjectives or with the use of hyperbolic expression as **aberrazione** (aberration) in (Ex.29), or indefinite collective adjectives (or pronouns).

Ex.29 @masechi Si è già assistito a Porta a Porta alla simulazione di un governo Monti con ministri La Russa e Bindi. Aberrazione audiovisiva.
@masechi At Porta a Porta we have already seen a simulation of Monti's government with La Russa and Bindi ministers. Audiovisual aberration.

Rhetorical question:
It is a figure of speech in the form of a question asked in order to make a point rather than to elicit an answer. It can be direct and explicit as in (Ex.30), or it can be an indirect rhetorical question.

Ex.30 Mario Monti? non era il nome di un antipasto? #FullMonti #laresadeiconti #elezioni #308.
Mario Monti? Wasn't it the name of an entree? #FullMonti #laresadeiconti #elezioni #308.

Other:
This last category represents ironic tweets, which can not be classified under one of the other seven previously described categories. It can occur, for example, in case of humor or situational irony. It is also applied when there is a number of overlapping categories, and thus, it is hard to define which one should be tagged first. In some cases this category is in practice a way for introducing hints about the presence of a class that is not included in the current schema but can be added in the future, e.g., pun or alliteration.

LEVEL 4: CLUES.
The fourth level of the scheme aims at annotating clues. They represent words that can help annotators to decide to which category belongs a given ironic tweet, such as **like** for analogy, **very** for hyperbole/exaggeration. Clues include also negation words, emoticons, punctuation marks, interjections, named entity (and mentions). Since the extraction of the information about this level can be done to a great extent by automatic tools, I did not address this specific task by manual annotation in this first phase, but I rather focused on the annotation of triggers of irony, by exploiting the availability of the morphosyntactic annotation including dependency-based trees (see following section).

2.1.2.2 From TWITTIRÒ to TWITTIRÒ-UD

As previously described in Section 1.2, in recent years UD has become the standard for morphosyntactic annotation [De Marneffe et al., 2014, Nivre et al., 2016] and the repository of UD projects enlarges by the day, also including data for under-resourced languages and less studied varieties. Its popularity derives from the fact that its creators proposed an improved taxonomy to capture grammatical relations across languages. They enumerate a set of broadly attested universal grammatical relations, in which existing dependency schemes for several languages can be mapped. As far as Italian is concerned, we focused on two main UD resources :

- UD_Italian-ISDT[12] (henceforth referred as UD-Italian) that entails standard texts drawn from newspapers, legal codes and Wikipedia
- UD_Italian-PoSTWITA[13] (henceforth referred as POSTWITA-UD) that entails texts from social media, namely Twitter

Two other resources developed for this language were not considered in our experiments. They are UD_Italian-VIT,[14] which has been only recently released for the first time and not enough validated until now, and UD_Italian-ParTUT,[15] which is a small parallel Italian, French and English corpus whose Italian section is part of UD-Italian.

In this section I will describe the application of the Universal Dependencies morphosyntactic scheme to the corpus TWITTIRÒ, which led to the creation and official release of TWITTIRÒ-UD in CoNLL-U format [Cignarella et al., 2019b].

The inspiring work of Karoui foresaw a fourth level of their scheme with a superficial analysis of morphology and syntax, intended to be useful for irony detection, but it was never studied further. In this part of my work on Italian, I went further on with Karoui's intuition and applied a complete and in-depth analysis on morphology and syntax both. The output of the application of the UD format to TWITTIRÒ is an essential part of my research, which, in my opinion, leads to more reliable results and also has the added value that it allowed to validate the starting hypothesis of the present thesis, i.e., that the morphosyntactic information might be useful to detect irony.

The enhancement of the morphosyntactic layer (*Universal Dependencies*)
In order to create TWITTIRÒ-UD, I applied the full pipeline of tokenization, lemmatization, PoS-tagging and dependency parsing provided by the tool *UDPipe*[16] [Straka and Straková, 2017]. For this purpose, I trained UDPipe on the two Italian resources we described above, namely PoSTWITA-UD [Sanguinetti et al., 2018] (6,712 tokens) and UD-Italian [Simi et al., 2014] (14,167 tokens). While the former has been selected because of its size, the latter instead because of the text typology it includes, which is

[12] https://universaldependencies.org/treebanks/it_isdt/index.html.
[13] https://universaldependencies.org/treebanks/it_postwita/index.html.
[14] https://github.com/UniversalDependencies/UD_Italian-VIT/tree/master.
[15] https://github.com/UniversalDependencies/UD_Italian-ParTUT/tree/master.
[16] The parsing is performed using Parsito (http://ufal.mff.cuni.cz/parsito).

the same of TWITTIRO. Considering the typology of text (i.e., tweets) and the features of ironic messages, I indeed followed the PoSTWITA-UD tenets, in particular for what concerns segmentation, which is at tweet level rather than at sentence level. From the manual correction of this dataset, through which I exploited the *Dependency Grammar Annotator*[17] I have learnt some interesting lessons.

Tokenization. Several tokenization errors depend on misspelled words (i.e., not correctly separated by spaces) or punctuation irregularly used, like in the following example:

sent_id = 516493351034826752
twittiro = EXPLICIT RHETORICAL QUESTION
sarcasm = 0
text = @User #labuonascuola deve riconoscere il merito di chi ha superato il concorso...solo in Italia chi vince perde?#dalleparoleaifatti

1	@User	@User	SYM	SYM	_	4	vocative:mention	_	_
2	#labuonascuola	#labuonascuola	SYM	SYM	_	4	nsubj	_	_
3	deve	dovere	AUX	VM	Mood=Ind|Number=Sing|...	4	aux	_	_
4	riconoscere	riconoscere	VERB	V	VerbForm=Inf	0	root	_	_
5	il	il	DET	RD	Definite=Def|Gender=Masc|...	6	det	_	_
6	merito	merito	NOUN	S	Gender=Masc|Number=Sing	4	obj	_	_
7	di	di	ADP	E	_	8	case	_	_
8	chi	chi	PRON	PR	PronType=Rel	6	nmod	_	_
9	ha	avere	AUX	VA	Mood=Ind|Number=Sing|...	10	aux	_	_
10	superato	superare	VERB	V	Gender=Masc|Number=Sing|...	8	acl:relcl	_	_
11	il	il	DET	RD	Definite=Def|Gender=Masc|...	12	det	_	_
12	concorso...solo	concorso...solo	NOUN	S	Gender=Masc|Number=Sing	10	obj	_	_
13	in	in	ADP	E	_	14	case	_	_
14	Italia	Italia	PROPN	SP	_	12	nmod	_	_
15	chi	chi	PRON	PR	PronType=Rel	17	nsubj	_	_
16	vince	vincere	VERB	V	Mood=Ind|Number=Sing|...	15	acl:relcl	_	_
17	perde?#dalleparoleaifatti	perde?#dalleparoleaifatti	CCONJ	CC	_	4	cc	_	SA=\n

Table 2.8: Tweet represented in CoNLL-U format, with errors.

In line 12 and line 17 of the tweet[18] we find "concorso...solo" and "perde?#dalleparoleaifatti", which should be split in three different tokens each. In order to avoid that the failures in tokenization propagate in the other annotation levels, before tokenization I applied an automatic data cleaning which consists in always adding a white space between words and punctuation signs (with the exception of the apostrophe which is left attached to the preceding token). I only manually corrected the remaining cases of misspelled tokens, that is not separated by the necessary white space. The result of the correction of the example above can be seen below (where I also corrected the PoS tags).

Lemmatization and PoS-tagging. Misspelled forms often occurring in social media contents can not be recognized by lemmatizers and their analysis may result in a failure. Here, as it was done in the annotation of PoSTWITA-UD [Sanguinetti et al., 2018] I associated the non-standard forms with the lemmas of their normalized versions, thus allowing a correct PoS-tagging.

[17] http://medialab.di.unipi.it/Project/QA/Parser/DgAnnotator/.

[18] Translation: @User #labuonascuola needs to acknowledge the merit of whose who passed the competition...only in Italy who wins also loses? #fromwordstofacts.

1	@User	@User	SYM	SYM	_	4	vocative:mention	_	_
2	#labuonascuola	#labuonascuola	SYM	SYM	_	4	nsubj	_	_
3	deve	dovere	AUX	VM	Mood=Ind|Number=Sing|...	4	aux	_	_
4	riconoscere	riconoscere	VERB	V	VerbForm=Inf	0	root	_	_
5	il	il	DET	RD	Definite=Def|Gender=Masc|Number=Sing|...	6	det	_	_
6	merito	merito	NOUN	S	Gender=Masc|Number=Sing	4	obj	_	_
7	di	di	ADP	E	_	8	case	_	_
8	chi	chi	PRON	PR	PronType=Rel	6	nmod	_	_
9	ha	avere	AUX	VA	Mood=Ind|Number=Sing|...	10	aux	_	_
10	superato	superare	VERB	V	Gender=Masc|Number=Sing|...	8	acl:relcl	_	_
11	il	il	DET	RD	Definite=Def|Gender=Masc|Number=Sing|...	12	det	_	_
12	concorso	concorso	NOUN	S	Gender=Masc|Number=Sing	10	obj	_	SA=No
13	...	...	PUNCT	FS	_	10	punct	_	SA=No
14	solo	solo	ADV	B	_	16	advmod	_	_
15	in	in	ADP	E	_	16	case	_	_
16	Italia	Italia	PROPN	SP	_	4	obl	_	_
17	chi	chi	PRON	PR	PronType=Rel	19	nsubj	_	_
18	vince	vincere	VERB	V	Mood=Ind|Number=Sing|...	17	acl:relcl	_	_
19	perde	perdere	VERB	V	Mood=Ind|Number=Sing|...	4	acl:relcl	_	SA=No
20	?	?	PUNCT	FS	_	19	punct	_	SA=No
21	#dalleparoleaifatti	#dalleparoleaifatti	SYM	SYM	_	4	parataxis:hashtag	_	SA=\n

Table 2.9: Tweet represented in CoNLL-U format, corrected.

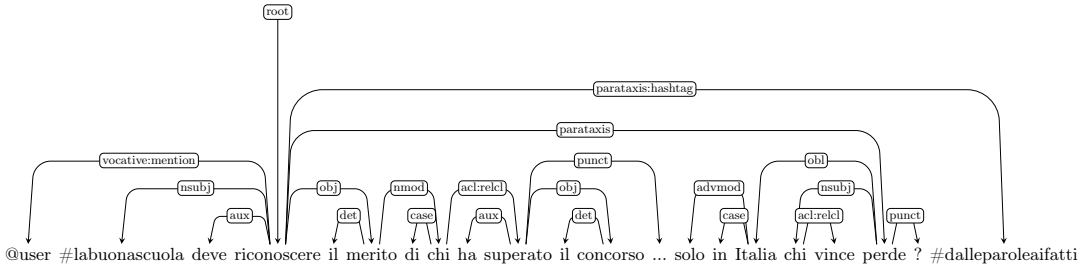


Figure 2.2: Tweet represented in the form of a dependency-based syntactic tree.

Following this principle, I thus assigned the corresponding lemma to the various cases of abbreviation, capitalization, typos and grammatical errors, and word lengthening. Some examples follow below:

- anema ⇒ anima (*soul*)
- ke ⇒ che (*that*)
- X ⇒ per (*for*)
- nooo ⇒ no (*no*)
- h ⇒ ora (*hour*)

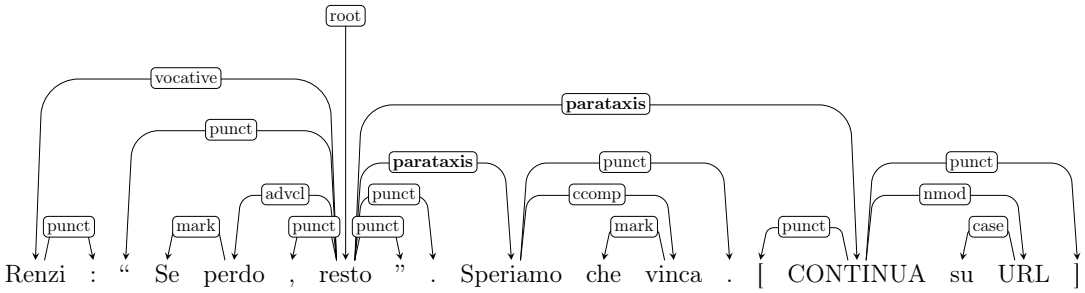
Emoticons, emojis, URLs, email addresses, and Twitter marks (hashtags and mentions) have been instead labelled with the tag `SYM`.

Despite the annotation efforts made in this phase of the creation of the corpus, back in 2017, some changes might now be needed. It is important to point out that recently[19] the issue of how to best solve problems related to the annotation in UD format of social media content, Twitter in particular, has been addressed in a recent paper published at LREC 2020. In that paper, in fact, we propose some annotation guidelines (perfectly

[19]The time of writing is December 2020.

compatible with the UD restrictions) that are emerging from an ongoing debate among researchers active in the UD community, who also aim at solving the idiosyncrasies found in social media content, in a handful of languages [Sanguinetti et al., 2020].

Dependency Relations Attachment. For what concerns the issues encountered in the last step of the annotation pipeline, I was able to identify two main problems: sentence splitting and selection of the correct root, and label attachment on Twitter marks. As said above, following the strategy applied in POSTWITA-UD, I did not perform any sentence splitting in the novel dataset. Each syntax tree of TWITTIRÒ-UD corresponds to a tweet [20] in its entirety, and may consist of multiple sentences too. At the same time, provided that the UD scheme poses a single-root constraint, the internal connections between different sentences occurring in a tweet have to be annotated and labeled by the dependency relation `parataxis`.[21] This relation is quite hard to be provided by the parser, which often fails in recognizing this kind of structure, even though it is trained on POSTWITA-UD where `parataxis` is annotated following the same strategy. From a syntactic point of view, as far as dependency relations and user-generated content are concerned, the selection of the root and the individuation of paratactic structures are not easy tasks. The TWITTIRÒ-UD dataset is rich of tweets that contain more than one sentence. To represent this syntactic phenomenon I widely exploited the `parataxis` label. See for instance, Figure 2.3 where I display a tweet containing more paratactic structures.



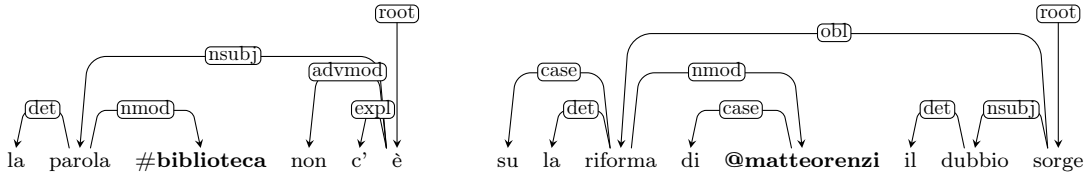
(*“If I lose, I stay”. Let’s hope he wins then. [Follows on URL]*)

Figure 2.3: Example of tweet containing multiple sentences.

Another issue I encountered, is related to the wide presence of Twitter marks in this kind of texts. The current limited amount of adequate training data prevents the parser from dealing with them successfully. Within the manual correction phase, I resort to the label `vocative:mention` for Twitter mentions, the label `discourse:emo` for emojis, and `dep` for URLs.
Moreover, hashtags and mentions could be either used appended at the end of the tweet, to create more emphasis, or with a full syntactic function. In the first case, I resort to

[20] All the tweets are less than 140 characters long, since the tweets composing the TWITTIRÒ-UD dataset were retrieved before Twitter allowed the 280-character limit.

[21] https://universaldependencies.org/u/dep/parataxis.html.



(*There is no word #library.*) (*about the reform of @matteorenzi a doubt rises.*)

Figure 2.4: Tweets containing a hashtag and a mention with syntactic function.

the relation (`parataxis:hashtags` and `vocative:mention`), while in the second case I annotate accordingly to the syntactic role, see for example in Figure 2.4 the hashtag #biblioteca and the mention @matteorenzi are labelled as `nmod`.

		PoSTWITA-UD	**TWITTIRÒ-UD**
hashtags	`parataxis:hashtag`	40.89%	54.79%
	`nmod`	19.64%	11.55%
	`nsubj`	13.48%	8.59%
	other	25.99%	25.07%
mentions	`vocative:mention`	92.37%	87.41%
	other	7.63%	12.59%

Table 2.10: Distribution of deprel labels for hashtags and mentions.

It is interesting to observe the quantitative distribution of hashtags and mentions and the syntactic role they cover. Table 2.10 shows the distribution of the dependency relations and confirms that there is a syntactic correlation of the peculiar semantic role that hashtags and mentions play in tweets.
The labels that are mostly exploited for linking the hashtags to the sentence structure in PoSTWITA-UD and in TWITTIRÒ-UD are mostly two: `nmod` and `nsubj`. This behaviour is confirmed in both datasets. In fact, `nmod` is 19.64% in PoSTWITA-UD and 11.55% in TWITTIRÒ-UD, while `nsubj` is 13.48% PoSTWITA-UD and 8.59% in TWITTIRÒ-UD. All the other occurrences are below the threshold of statistical significance, therefore I do not report them (encompassing them in *other* category in the table). On the other hand, user mentions seems to play most of the time the dependency role of `vocative:mention` (92.37% in PoSTWITA-UD and 87.41% in TWITTIRÒ-UD). This label, that specializes `vocative`, has been indeed introduced exactly for marking this case in PoSTWITA-UD.

In Table 2.10 it can also be appreciated that the behaviour and the distribution of these two Twitter marks is really similar between the two social media treebanks PoSTWITA-UD and TWITTIRÒ-UD. I did not report the values concerning the texts in the UD-Italian [Simi et al., 2014]), because being a treebank of standard text, and not of UGC, it does not contain neither hashtags nor mentions.

A quantitative analysis over Italian treebanks. Once I have manually corrected the

dataset, I have detected the following frequencies of dependency relations. In Table 2.11 I display the distribution of dependency relations in UD-Italian, i.e., the resource mainly representing the features of standard Italian text, PoSTWITA-UD and TWITTIRÒ-UD. Despite the sparseness of relations, we can observe how their frequency and distribution characterizes the language exploited in the social media data collected in TWITTIRÒ-UD and PoSTWITA-UD with respect to the standard language collected in UD-Italian.

	UD-It	**PoSTW**	**TWIT**		**UD_It**	**PoSTW**	**TWIT**
acl	0.99	0.48	0.65	flat	0.19	0.35	0.10
acl:relcl	1.06	0.68	0.71	flat:foreign	0.05	0.28	0.05
advcl	1.26	1.00	0.90	flat:name	1.17	2.18	0.85
advmod	3.53	4.85	4.21	goeswith	0.00	0.03	-
amod	5.59	2.75	3.49	iobj	0.23	0.75	0.52
appos	0.31	0.43	0.16	list	-	0.22	-
aux	2.02	1.67	1.80	mark	2.11	2.23	2.10
aux:pass	0.75	0.12	0.18	nmod	8.01	6.84	5.68
case	14.03	9.42	10.23	nsubj	4.30	4.50	4.40
cc	2.73	2.26	1.80	nsubj:pass	0.77	0.16	0.26
ccomp	0.49	0.80	0.67	nummod	1.20	0.88	0.93
compound	0.25	0.17	0.27	obj	3.43	4.10	4.64
conj	3.39	2.95	1.72	obl	5.77	4.03	4.80
cop	1.15	1.75	1.54	obl:agent	0.38	0.12	0.13
csubj	0.11	0.17	0.07	orphan	0.01	0.05	-
csubj:pass	0.00	-	-	parataxis	0.14	4.02	4.62
dep	0.00	2.34	0.89	parataxis:appos	-	0.10	0.01
det	15.54	10.97	10.98	parataxis:discourse	-	0.02	0.01
det:poss	0.63	0.48	0.31	parataxis:hashtag	-	1.81	2.15
det:predet	0.14	0.12	0.11	parataxis:insert	-	0.03	-
discourse	0.02	1.18	0.75	parataxis:nsubj	-	0.03	-
discourse:emo	-	0.59	0.13	parataxis:obj	-	0.07	-
dislocated	0.01	0.11	0.01	punct	11.36	12.08	17.24
expl	0.73	0.85	0.96	root	4.75	5.39	4.77
expl:impers	0.14	0.15	0.13	vocative	0.03	0.38	0.09
expl:pass	0.13	0.05	0.04	vocative:mention	-	2.06	2.89
fixed	0.32	0.19	0.30	xcomp	0.76	0.76	0.78

Table 2.11: Dependency relations' distribution across the three main Italian treebanks. The values are expressed in percentage %.

As expected, meaningful differences emerge for parataxis and punctuation. Punctuation is indeed exploited more extensively in the two social media datasets (12.08% and 17.24%) than in UD-Italian (11.36%), and the frequency of the `parataxis` deprel is 4.02% and 4.62% in PoSTWITA and TWITTIRÒ-UD, while it is only 0.14% in UD-Italian, marking a significant difference. The distributions of the relations `vocative:mention` and `parataxis:hashtag` especially features the two social media treebanks. The mentions' deprel is 2.06% in PoSTWITA-UD and 2.89% in TWITTIRÒ-UD, while the hashtags are respectively 1.81% and 2.15%. Furthermore, it is interesting to notice how the use of passive voices (`aux:pass`) is 0.75% in the UD-Italian treebank while only 0.12% in PoSTWITA-UD and only 0.18% in TWITTIRÒ-UD, indicating a preference

for the exploitation of active voices in the language used in social media, as it happens in spoken language.

A parsing experiment. In order to preliminarily evaluate the similarities between the three datasets, I performed an evaluation based on syntactic and morphological analysis, applying UDPipe on the TWITTIRÒ-UD gold corpus as a test set. The following three settings were exploited:

1) training UDPipe using only UD-Italian (UD_It),
2) training UDPipe using only PoSTWITA-UD (PoSTW),
3) and training UDPipe using both resources (UD_It+PoSTW).

For evaluation I used the script made available for the CoNLL 2018 Shared Task[22] with the default setting parameters. Table 2.12 surveys the resulting scores for precision (P), recall (R) and averaged F1-score (F1).

	UD_It			**PoSTW**			**UD_It+PoSTW**		
	P	R	F1	P	R	F1	P	R	F1
Tokens	66.85	67.28	67.07	66.50	65.15	65.82	67.62	67.63	67.62
Sentences	66.18	66.18	66.18	66.18	66.18	66.18	66.18	66.18	66.18
Words	66.73	67.12	66.92	66.36	65.01	65.67	67.54	67.56	67.55
UPOS	57.10	57.44	57.27	62.71	61.44	62.07	65.75	65.77	65.76
XPOS	56.30	56.63	56.47	62.23	60.97	61.59	65.59	65.61	65.60
Feats	59.35	59.70	59.52	62.17	60.91	61.53	65.64	65.66	65.65
AllTags	55.11	55.43	55.27	60.59	59.36	59.97	65.04	65.06	65.05
Lemmas	60.88	61.23	61.05	62.17	60.91	61.53	65.48	65.50	65.49
UAS	66.73	67.12	66.92	66.36	65.01	65.67	67.54	67.56	67.55
LAS	50.12	50.42	50.27	54.07	52.97	53.51	56.84	56.85	56.85

Table 2.12: Evaluation of UDPipe.

First of all, it is interesting to notice the variation of the Unlabelled Attachment Score (UAS) and Labelled Attachment Score (LAS). For what concerns UAS, the first setup, where only the data from UD-Italian have been used for training, allowed a better result than the second one, where PoSTWITA-UD is the training dataset. But the opposite can be seen for LAS. We can hypothesize that the larger amount of data in UD-Italian allowed to build a more representative statistical model. Nevertheless, training on a resource which includes the same typology of data may be crucial for collecting an adequate knowledge about the specific relations exploited. This motivates the best scores for LAS and UAS, which were obtained in the third setup benefiting of both the resources for training. This encourages us to develop more and better gold standard treebanks from social media content to be used for training and evaluating NLP tools.

The irony activators. Finally, as described in an earlier section, in some work from 2019 I tried to capture the relation between the two layers of annotation mentioned in the

[22] https://universaldependencies.org/conll18/evaluation.html.

sections above. On the one hand the pragmatic/semantic information regarding irony captured by the annotation described in Section 2.1.2.1, and on the other hand morphosyntactic information captured by applying the UD annotation scheme ad described in Section 2.1.2.2.

This part of my research, is decisively linked with the overly-cited work of Karoui, especially trying to link levels 2 and 3 (type and category) of the annotation scheme of irony, with the 4th level (cues) (see Section 2.1.2.1). With this investigation on the cue words of irony, which I afterwards called "irony activators", I was finally able to give my own original contribution to the scheme of Karoui.

This part of my research on triggers has for now been left a bit aside, and studied only in a qualitative fashion, and, for now, has not impacted on the experiments on irony detection. However, I will surely use this additional information, which links semantic triggers and their morphosyntactic features, for further investigation and I will explore its application on other corpora and across other languages. In particular, my intuition is that such trigger words, when combined with syntactic and semantic information, may shed some light on the discovery of significant patterns revealing irony.

Therefore, in this section I will describe how I annotated the so-called *irony activators* (i.e., the specific cue words that serve as triggers in the production of irony) by taking advantage of the structure of the CoNLL-U format imposed by the UD framework, which allowed to annotate significantly relevant words at a token level [Cignarella et al., 2019c, 2020c], but I will not linger much on it, since as I said, it has been until now just a study from a qualitative perspective and has not been of any use in the experimental settings.

It is important to emphasize that I have adopted the annotation scheme proposed by Karoui, firstly to make an exploratory qualitative investigation of the data I had collected in an even more fine-grained perspective. Secondly, I wanted to test the scheme on Italian (i.e., a language on which it had been applied only partially). Finally I was able to annotate the so-called "triggers" at a token level, by making use of the presence of a complete and detailed morphosyntactic annotation, validated by hand, which was not done even for French.

To the best of my knowledge, the TWITTIRÒ-UD corpus is one of the few linguistic resources where irony annotation and morphosyntactic annotation are applied within the same framework. By taking advantage of the many layers of annotation I was interested in whether *there could be any syntactic pattern that can help to automatically detect irony*. The intuition that I followed in this work is that: if such "syntactic patterns" which activate irony do actually exist, therefore, they should be particularly evident in the syntactic context of certain lexical elements that create a semantic clash in a text. For this reason, I thought of annotating specific irony activators in the TWITTIRÒ-UD corpus, taking advantage of the fact that the annotation format I adopted for the syntactic annotation allows us also to label specific activators at token level and retrieve dependency relations connected to them. In doing so, I was led to think whether *there could be an effective way to annotate irony activators*, and I tried to propose one. In this section I will describe the guidelines for the annotation of irony activators and walk the reader through the process of their evaluation and discussion.

As previously mentioned, irony is activated by the presence of a clash or a contradiction between two elements or two propositions (P1 and P2), which are indeed the triggers of the activation of irony. According to the scheme proposed by Karoui et al. [2017], which I followed, there are two kinds of *activation types*: EXPLICIT when both these elements are lexicalized in the message, or IMPLICIT otherwise. The concept of "*irony by clash*" has been also thoroughly studied in a work by Van Hee et al. [2018b], where the authors address the challenge of modeling implicit or prototypical sentiment in the framework of automatic irony detection. They conclude that most classification errors on the *irony by clash* category mainly include tweets where a polarity contrast is difficult to perceive, even if implicit sentiment is taken into account.

In this step of my work, I focused on the manual annotation of irony activators and on providing annotation guidelines that could be useful also for other datasets in different languages, within the same multilingual project. Indeed, the starting point of the present work is connected to the PhD thesis of Karoui [2017], on a French dataset, in which the author tried to annotate at tweet level some elements that are responsible for the activation of irony. In that approach, each tweet had to be annotated using the Glozz tool [Widlöcher and Mathet, 2009], in terms of units and relationships between units (if the relationship existed). Three types of relationship were taken into account: 1) relation of comparison, 2) relation of explicit contradiction, and 3) relation of cause/consequence.

With respect to my work I opted for a finer-grained annotation also taking advantage from the availability of tokenized data and a full syntactic analysis in UD format. My aim is to annotate irony activators in the whole TWITTIRÒ-UD corpus. Differently from what proposed in [Karoui, 2017], in which the elements creating an ironic contrast (P1 and P2) could be words, phrases or even full sentences, in this work, since I want to highlight the interaction between the pragmatic phenomenon of irony and its syntactic representation, I define as irony activators a pair of words T1 and T2 that must correspond to nodes of the syntactic dependency tree.

Given an ironical utterance (in this case a tweet) and its dependency-based syntactic representation, where each node in the tree structure represents a word, T1 and T2 is a pair of words – regardless of their grammatical category – such that:

- either they are both lexicalized (in explicit irony) or one of them is left unspecified (implicit irony);
- they act as triggers by signaling the presence of an ironic device.

The intuition behind this choice is inspired by the work of Saif et al. [2016], in which the authors underline the importance of contextual and conceptual semantics of words when calculating their sentiment, which in turn comes from the popular dictum "You shall know a word by the company it keeps!" [Firth, 1957]. My idea is, in fact, to proceed in two steps: firstly, to annotate irony triggers at token level, and subsequently to retrieve the other tokens that "keep company" to them by means of the dependency relations available from the UD annotation.

Therefore, as I have already highlighted, if any kind of "syntactic pattern" that can help us to automatically detect irony does actually exist, we assume this will be partic-

ularly evident in the "syntactic circle" around the lexical elements that create a contradiction and are the lexical activators of the ironic realization, namely T1 and T2.[23]

For instance, in the syntax tree of the tweet "Spero sia colite. Ma ho paura sia amore." (*I hope it's colitis. But I'm afraid it's love.*), we have annotated T1 and T2 according to the guidelines of the category ANALOGY (see below in this section). The syntactic tree looks like this:

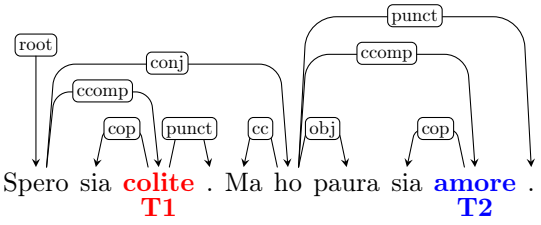


Figure 2.5: Syntactic tree in UD flat format, with highlighted irony activators.

According to my intuition, if "syntactic patterns" that help to detect irony do exist, they should be particularly evident in the syntactic context of certain lexical elements that create a semantic clash in a text (i.e., T1 and T2). After extracting automatically the "sub-tree" surrounding the irony activators, we would have this tree representation:

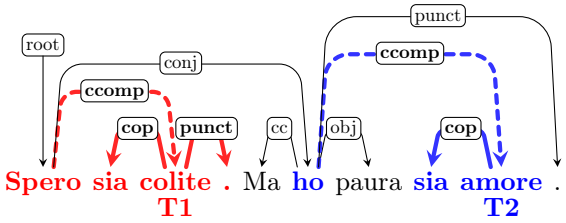


Figure 2.6: Highlighted irony activators T1 and T2 and their respective sub-trees.

Concerning the tweet above, the tokens directly connected through dependency relations are:

$$T1 = [spero, sia, colite, .] \text{ and } T2 = [ho, sia, amore].$$

From a dependency relation viewpoint T1 and T2 are connected by means of:

$$T1 \rightarrow T2 = [T1 \rightarrow ccomp \rightarrow conj \rightarrow ccomp \rightarrow T2].$$

Such information could be exploited as feature in the implementation of automatic systems for the detection of irony, but it could also be useful to gain new insights on patterns that may underlie the activation of irony.

[23] T1 and T2, replacing P1 and P2 used by [Karoui, 2017], stand for "token".

The decision of annotating exclusively a single token corresponding to each irony activator found in the tweet lies in a twofold motivation. Firstly, managing only a pair of tokens per tweet is computationally easier and more homogeneous. Secondly, each node of a dependency tree can be seen as a head of a sub-tree, i.e., a word usually semantically richer with respect to its dependents and, therefore, more interesting for my analysis. Furthermore, in this way we are also able to analyze morphological similarities between irony activators and across irony types and categories.

2.2 From feature-based approaches to deep learning

After having worked for such a long time on the development of a linguistic resource for irony detection, exploitable in NLP, and thus having acquired a better knowledge of the phenomenon of irony itself and its computational characteristics, I finally moved to its automatic detection.

In most of the studies relying on Twitter data, irony detection has been modeled as a binary classification problem, where mostly tweets labeled with certain hashtags (i.e., *#irony*, *#sarcasm*, *#sarcastic*, *#not*) have been considered as ironic utterances. Following this framework, different approaches have been proposed [Davidov et al., 2010, González-Ibáñez et al., 2011, Reyes et al., 2013, Riloff et al., 2013, Barbieri et al., 2014, Ptáček et al., 2014, Hernández Farías et al., 2015]. The authors proposed models that exploit mainly features related to textual-content such as: punctuation marks, emoticons, part-of-speech labels, discursive terms, specific patterns [Riloff et al., 2013].

For what concerns the affective information, some approaches already used in their models sentiment and emotional information. Reyes et al. [2013] considered some features to characterize irony in terms of elements related to sentiments, attitudes, feelings and moods exploiting the Dictionary of Affect in Language (DAL) proposed by [Whissell, 2009]. Barbieri et al. [2014] considered the amount of positive and negative words by using SentiWordNet. Hernández Farías et al. [2015] exploited two widely applied sentiment lexicons (Hu&Liu and AFINN)[24] as features in their model. More recent works focused specifically on studying the role of affective information in a comprehensive manner, by exploring the use of a wide range of lexical resources available for English, reflecting different aspects of this multi-faceted phenomenon [Hernández Farías and Rosso, 2016].

Another key characteristic for irony is *unexpectedness* [Attardo, 2000]. According to many theoretical accounts, people infer irony when they recognize an incongruity between an utterance and what is known (or expected) about the speaker and/or the so-called pragmatic shared context. Recent approaches started to more specifically address such issue, taking into account information about context [Rajadesingan et al., 2015, Bamman and Smith, 2015, Wallace et al., 2015].

While the majority of the systems in the above-mentioned shared tasks are based on classical machine learning techniques (see Section 2.1), researchers have recently started to exploit approaches based on deep learning. In fact, similarly to other NLP tasks, also

[24] Hu&Liu: http://www.cs.uic.edu/~liub/FBS;
AFINN: http://github.com/abromberg/sentiment_analysis/blob/master/AFINN.

in irony detection, deep learning approaches have been widely used and proved successful even in the shared tasks described in the section above. For instance, the best-scoring team for Task 3 at Semeval 2018 used a densely connected LSTM based on pre-trained word embeddings and sentiment and PoS tag features [Wu et al., 2018]; they built a multi-task model to predict the missing irony hashtag, whether a tweet is ironic or not and the fine-grained type of irony. A similar multi-task learning approach was adopted by Cimino et al. [2018] at IronITA 2018, which introduced a 2-layer Bidirectional Long Short-Term Memory (BiLSTM) that exploits also additional information, such as automatically-generated sentiment polarity lexica, word embedding lexica and PoS tags (see Section 2.1.1). The irony detection system proposed by González et al. [2019] at IroSVA 2019 was based on the Transformer Encoders model. At DEFT 2017 the winning team of the shared task for French proposed an approach based on varying the size of the layers of a Convolutional Neural Network (CNN), combined with three different sentiment-based word embeddings [Rouvier and Bousquet, 2017]. In another approach, Huang et al. [2017] applied attentive Recurrent Neural Networks (RNNs) that capture specific words which are helpful in detecting the presence of irony in a tweet, and Zhang et al. [2019] took advantage of recent advancements in transfer learning techniques. Finally, a multilingual scenario is also proposed by Ghanem et al. [2020], that describe both feature-based models (with Random Forest showing the best results) and a CNN architecture with multilingual embeddings, and test their approach on Arabic, English and French.

While trying to generalize and draw some conclusions and main ideas derived from precedent work, it still seems that 'classical' machine learning architectures can still compete with brand new neural models, and obtain comparable results. One big issue that needs to be stressed is the fact that some results, obtained through systems that have a high number of manually engineered features, still seem to be more explainable than the results obtained with most of the deep learning methods. On this regard, prioritizing the explainability of results must be the path to follow, with respect to the obtaining of higher performances.

Nonetheless, it is important to keep experimenting on detection tasks such as irony detection also with the newest models applicable to NLP such as CNNs and Transformers. Not surprisingly, in fact, in the last years there has been also wide proliferation of models that are based on the Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al., 2018], which proved to be the state-of-the-art technique for a variety of NLP tasks. The latest trend is, indeed, that of inputting linguistically motivated engineered features inside the transformer architecture. For instance, Srivastava et al. [2020] implemented a hierarchical BERT architecture for sarcasm detection, that first, extracts the local features from the words in a sentence, and then uses a convolution module to summarize all the sentences in a context to detect whether the response is sarcastic or not. Another BERT-based approach is the one from Baruah et al. [2020], who participated in the FigLang 2020 workshop, and proposed an approach which uses the conversational context to detect sarcasm, by varying the amount of context used along with the response. The amount of context used includes either zero context, the last one to three utterances, or all previous utterances. Lastly, at the same workshop,

Thenmozhi et al. [2020], presented a work in which they compare traditional machine learning approaches, a deep learning approach (RNN-LSTM) and BERT for identifying sarcasm with the input of different sizes of contextual information.

To resume, the variety of approaches used nowadays to address the task of irony detection is still very wide. It ranges from SVM with the addition of very simple lexical features, it passes through neural architectures that rely on both linguistic features and external resources such as lexica and word embeddings, finally reaching the exploitation of BERT-based language models as classifiers. The latest trend is trying to "infuse" additional features in pre-trained architectures. In many works the addition of new features it has seemed to *mislead* the original classifier.

Finally, more and more researchers are addressing detection problems such as this one, from a multilingual perspective, trying to level performances to those obtained in monolingual settings.

As we can infer from the last two sections, the majority of the research on irony detection has been addressed in English, although interest for such research in other languages is deifinitely growing, such as: Dutch [Kunneman et al., 2015], Italian [Bosco et al., 2013, Cignarella et al., 2020c], Czech [Ptáček et al., 2014], French [Karoui et al., 2015], Portuguese [Carvalho et al., 2009], Chinese [Tang and Chen, 2014, Xiang et al., 2020] and Arabic [Ghanem et al., 2019b, Farha and Magdy, 2020].

2.3 Irony detection using dependency syntax

Some research already explored different kinds of syntactic features and their interaction in several NLP tasks, showing their effectiveness. For example, Sidorov et al. [2012] exploited syntactic dependency-based n-grams for general-purpose classification tasks, Socher et al. [2013] investigated sentiment and syntax with to the development of a sentiment treebank, and Kanayama and Iwamoto [2020] showed a pipeline method that makes the most of syntactic structures based on Universal Dependencies, achieving high precision in sentiment analysis for 17 languages. Morphology and syntax have also been proved useful in a number of other tasks, such as rumour detection [Ghanem et al., 2019a], authorship attribution [Posadas-Duran et al., 2014, Sidorov et al., 2014] and humor recognition [Liu et al., 2018a].

To the best of my knowledge, very few studies use syntactic information specifically for irony detection. For this reason I consider this as a novel aspect of my research, and in this section I will present two important steps of the research I conducted towards the detection of irony exploiting morphology and syntax information. On the one hand I will describe one preliminary work [Cignarella and Bosco, 2019], where I participated in the IroSvA 2019 irony detection shared task in Spanish variants [Ortega et al., 2019] by submitting a system employing a Support Vector Classifier (SVC) combined with shallow features based on morphology and dependency syntax.

On the other hand, I will present my most recent work [Cignarella et al., 2020a]: an in-depth investigation of the effectiveness of dependency-based syntactic features on the irony detection task in a multilingual perspective (English, Spanish, French and Italian).

I provide three distinct experimental settings. In the first, I explore a variety of syntactic dependency-based features combined with classical machine learning classifiers. In the second scenario, I combine two types of word embeddings trained on parsed data and tested against gold standard datasets. In the third setting, I combine dependency-based syntactic features into the Multilingual BERT architecture. The results suggest that fine-grained dependency-based syntactic information is highly informative for the detection of irony.

2.3.1 Participation in the *IroSVA 2019* shared task

In the present section I describe my participation in the IroSvA 2019 shared task at IberLEF 2019 [Ortega et al., 2019], which is focused on Irony Detection in Spanish Variants, addressing the identification of irony as a classical binary classification task. My approach is mainly oriented in performing a preliminary test of the importance of morphosyntactic information in the task of irony detection. For this reason, I exploited a straightforward methodology: a Support Vector Classifier with a linear kernel, combined with shallow features based on morphology and dependency syntax. For the representation of such kind of knowledge I relied on the application on the data of the *Universal Dependencies* format, as extensively explained in this chapter.

Task description. The task is structured into three subtasks, each one for predicting whether text messages are ironic or not in three different variants of the Spanish language, i.e., those spoken respectively in Spain, Mexico and Cuba. The aim is that of investigating whether a short message, written in the Spanish language, is ironic or not with respect to a given context.

	Training		Test	
	ironic	not-iro	ironic	not-iro
Spain (es)	800	1,600	200	400
Mexico (mx)	800	1,600	200	400
Cuba (cu)	800	1,600	200	400
	7,200		1,800	

Table 2.13: Data distribution.

The organizers provided a different training and test set for each Spanish variant where the items were distributed as shown in Table 2.13. The Spanish and Mexican sets are composed by tweets, while the sets from Cuba included news comments from three popular Cuban news sites. With respect to the previous tasks on irony detection, the messages here are not considered as isolated texts but together with a given context (e.g., a headline or a topic), as shown in Table 2.14.

As far as the distribution of irony, each training set contained 800 ironic and 1,600 not ironic texts. The same proportion of approximately 33%-66% has been also maintained in the test set, counting respectively 200 ironic and 400 not-ironic texts in each subset.

	Training		Test	
	ironic	not-iro	ironic	not-iro
DigitalTV	137	275	32	65
Sports	108	219	28	55
E-Quality	100	201	25	51
E-Mobile	92	185	23	47
Transport	91	184	23	46
TechSociety	85	172	22	44
IC-Trade	74	150	19	38
Economy	57	103	14	26
Science	56	111	14	28
Total	800	1,600	200	400

Table 2.14: Training and Test distribution on the Cuban variant data.

Methodology. Because of my interest in features related to syntax and their contribution in figurative language detection, I did not take into account lexical features also considering that they can be too much influenced by the involved Spanish varieties. Therefore, I trained the automatic system on the three datasets altogether (7,200 texts) and tested the same model on the three different test sets, regardless of the three variants of Spanish.

Preprocessing. I applied two preprocessing steps: the stripping of URLs from texts all normalized to lowercase letters, and the morphosyntactic analysis. I trained the *UDPipe* pipeline (which includes tokenization, Part of Speech tagging and parsing) on the UD_Spanish-GSD corpus [McDonald et al., 2013] for generating for each item of the dataset a CoNLL-U tree in Universal Dependencies format.

The system. After having performed experiments with Random Forest (RF), Decision Trees (DT) and Support Vector Machine (SVM), I finally implemented a system with a linear kernel with the last classifier, which resulted the best performing one. I proposed a straightforward approach with two different types of features: a) common *"baseline"* features, widely explored in sentiment analysis tasks and in irony detection tasks too; and b) new syntactic *"dependency-based"* features.

The novel contribution of my work mainly consists in the exploitation of these latter features to create vectorial representations of texts; all them (listed in point b) have been made available thanks to the application of the UDpipe pipeline together with the subsequent generation of the dependency trees corresponding to the items of the datasets. Figure 2.7 shows a tweet where the UD format has been applied.

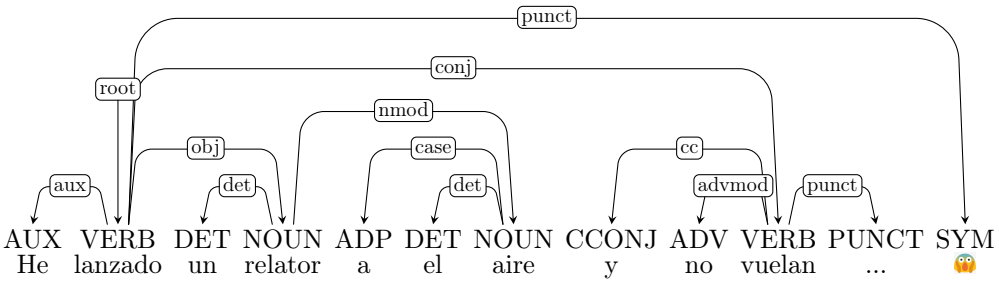
a) Baseline features

• *Bag of words (Bow):* each tweet was pre-processed to convert it to lowercase letters. Then we extracted unigrams, bigrams and trigrams to create a binary representation.

• *Bag of Char-grams (BoC)*: we considered the sequence of char-grams in a range from 2 to 5 characters.

b) Dependency-based features

• *Bag of Dependency Relations (BoDeprel):* following the approach described in Ghanem et al. [2019a], we considered the sets from 5 to 7 dependency relations as occurring in the linear order of the sentence from left to right.



(*I have launched a reporter in the air and they do not fly...* 😱)

Figure 2.7: Example of dependency tree.

• *Bag of SyntaxPath's Word Forms (Path_ Form):* starting from the intuition of Sidorov et al. [2013, 2014] we created a Bag of Word Forms (tokens), considering the bigrams that can be collected following the syntactic tree structure (rather than the bigrams that can be collected reading the sentence from left to right). For instance, the Path_Form corresponding to the sentence in Figure 2.7 includes ['lanzado', 'he'], ['lanzado', 'relator'], ['relator', 'un'], ['relator', 'aire'], ['aire', 'a'], ['aire', 'el'], ['lanzado', 'vuelan'], ['vuelan', 'y'], ['vuelan', 'no'], ['vuelan', '...'].

• *Bag of SyntaxPath's Deprels (Path_ Deprel):* we created a Bag of Deprels, collecting the dependency relations occurring in the structure of the syntactic tree, i.e., following the syntactic paths, thus creating a vectorial space based on bigrams, combining dependency relations in pairs. For instance, the Path_Deprel corresponding to the sentence in Figure 2.7 includes ['root', 'aux'], ['root', 'punct'], ['root', 'conj'], ['conj', 'cc'], ['conj', 'advmod'], ['conj', 'punct'], ['root', 'obj'], ['obj', 'det'], ['obj', 'nmod'], ['nmod', 'case'], ['nmod', 'det'].

The model is available at: `https://github.com/AleT-Cig/ATC_IroSvA_2019`.

Results. In Table 2.15 are shown the official results together with the four baselines proposed by the organizers. We can observe how, on average, the dependency-based approach I propose performs better than shallow lexical techniques, such as *word n-grams*, but is not stronger than more refined approaches, such as those implemented in the word embedding approach of *word2vec* and *LDSE* [Rangel et al., 2018].

A fine-grained observation of results[25] shows that the system I proposed performs

[25]Detailed results on the official website: `https://www.autoritas.net/IroSvA2019/`.

	ES	MX	CU	AVG
LDSE	0.6795	**0.6608**	**0.6335**	**0.6579**
W2V	**0.6823**	0.6271	0.6033	0.6376
Word n-grams	0.6696	0.6196	0.5684	0.6192
MAJORITY	0.4000	0.4000	0.4000	0.4000
my approach	0.6512	0.6454	0.5941	0.6302

Table 2.15: Official results and baselines obtained over the test set.

better on the Spanish and Mexican datasets (F_{avg} ES = 0.6512, F_{avg} MX = 0.6454) and slightly worse in the Cuban variety (F_{avg} CU = 0.5941). I recall this might be connected to the nature of the datasets, in fact, the first two are composed by tweets while the Cuban dataset is composed by news comments which may be featured by a slightly different lexical and syntactic structure.

From the numbers shown in Table 2.15, the system based on a linear SVM including syntactic features (*my approach*) outperforms a system using only word n-grams. This is true on average and for the Mexican and Cuban datasets, but not for the Spanish dataset. In addition, the results for the syntax-enriched model are, in Spanish and Cuban, lower than the results for the systems using word embeddings, but for the Mexican dataset. Generally speaking, the results of this set of experiments are not consistent and thus it would be misleading to say that syntax is beneficial overall in this task. Although, recently, I conducted further investigation on the importance of dependency-based syntax and word embeddings standing alone, and also syntactic features combined with word embeddings, showing interesting results [Lai et al., 2021].

Finally, in future work, it might be interesting to apply the method developed in the above-cited latest work to the datasets exploited in the work depicted in Table 2.15 (IroSvA 2019 datasets) and compare the obtained results with state-of-the-art ones.

The results show that focusing on syntactic features, namely *Bag of SyntaxPath's Word Forms* and *Bag of SyntaxPath's Deprels* (i.e., the novel contribution of this task participation), produced a good contribution to the Irony Detection task in Spanish Variants.

The participation in this shared task, and the application of syntax-based features allowed to understand the way in which syntax patterns play with irony, and to investigate in a deeper way the relationships between words (especially those that are distant among each other). For instance, by referring again to Figure 2.7, where the ironic content is produced by the semantic clash between the words '*lanzado*' (thrown) and '*vuelan*' (fly), it is possible to grasp their connection only with longer dependency-based features. If we had only considered approaches such as word n-grams, that take into account only the words that are in the immediate proximity of each other, the ironic meaning would have been lost. In fact, with dependency-based features is possible to capture the information in which words precede, follow and are syntactically related to each other.

Considering that the results seem quite promising, and that the dependency-based features deserve a finer-grained study, in the next section I will further investigate them observing their behaviour in other languages and paired with state-of-the-art architec-

tures. In particular, thanks to the great adaptability of the UD format across different languages, I could test these new features in a multilingual scenario too.

2.3.2 Multilingual irony detection with neural models

In this section I will go one step further with respect to what I have presented in Section 2.3.1 regarding my automatic system submitted to IroSVA 2019. Here I will focus for the first time on the development of syntax-aware irony detection systems in a multilingual perspective (English, Spanish, French and Italian), providing an in-depth investigation of the impact of different sources of syntactic information when used on top of several machine learning models, including Recurrent Neural Networks and Transformers. It is important to note that my aim is not to outperform the current state of the art on monolingual irony detection, but to investigate whether irony detection based on *syntax alone* can achieve comparable results with existing systems when evaluated in *multiple languages*.

I believe this is an important first step before moving to complex scenarios where both syntax and pragmatic knowledge are incorporated into deep learning models which could lead to explicitly model the syntax-pragmatic interface of irony.

Furthermore, as anticipated in Section 2.3.1, an approach based on dependency syntax could shed some light on the explainability of the computational model of a difficult pragmatic phenomenon such as that of irony. It is indeed, through the application of the UD format to ironic tweets, that I was able to observe the apparently hidden connections between words in a text and, thus, better grasp the semantic clash that produces an ironic meaning.

To this end, and as a summarizing in the concluding section regarding the chapter on irony detection, I aim at addressing the following questions. Firstly, I would like to investigate *whether features derived from morphology and syntax could help to address the task of stance detection* [RQ-1] (see Section 1.3), and following *to what extent does using resources such as treebanks for training NLP models improve the performance in stance detection* [RQ-2] (see Section 1.3).

Similarly to other NLP tasks, neural approaches in irony detection have been widely used, and proved successful even in the mentioned shared tasks. For instance, the best-scoring team for Task 3 at Semeval 2018 used a densely connected LSTM based on pre-trained word embeddings together with sentiment and PoS tag features [Wu et al., 2018]; they built a multi-task model to predict the missing irony hashtag, whether a tweet is ironic or not and the fine-grained type of irony. A similar multi-task learning approach was adopted in Cimino et al. [2018] at IronITA 2018, which introduced a 2-layer BiLSTM that exploits also additional information, such as automatically-generated sentiment polarity lexica, word embedding lexica and PoS tags. The irony detection system proposed in González et al. [2019] at IroSVA 2019 was based on the Transformer Encoders model. At DEFT 2017 the winning team of the shared task for French Rouvier and Bousquet [2017] proposed an approach based on varying the size of the layers of a CNN, combined with three different sentiment-based word embeddings. Finally, a multilingual scenario is also proposed in Ghanem et al. [2020], that describe both feature-based models (with Random

Forest showing the best results) and a CNN architecture with multilingual embeddings, and test their approach on Arabic, English and French (for the latter, the DEFT 2017 dataset was used, as in the present work).

As mentioned in Section 2.3, to the best of my knowledge, very few studies use syntactic information specifically for irony detection. While in the work described in this section I tested novel models in a multilingual setting (with partially overlapping languages with respect to Ghanem et al. [2020]), the novelty of my approach consists in using features which represent syntactic knowledge that can be extracted from text by applying morphological and syntactic analysis. *Universal Dependencies* played a key role in this respect, providing a set of broadly attested universal grammatical relations, in which existing dependency schemes for different languages can be mapped.

Nevertheless, the genre of social media texts can be an issue in the annotation pipeline, as they may be especially rich in non-standard forms and, therefore, harder to parse. This, in turn, motivated the development of several treebanks in different languages [Kong et al., 2014, Seddah et al., 2012], among which a large number created by using the UD format [Blodgett et al., 2018, Bhat et al., 2018, Cignarella et al., 2019b, Rehbein et al., 2019, Sanguinetti et al., 2018], or developed by conversion from previous formats [Albogamy and Ramsay, 2017, Liu et al., 2018b]. Such proliferation of social media-based treebanks in a short time span highlights the need for a consistent representation of linguistic phenomena, not only for typical contrastive studies, but also - and more importantly in this context - when using such resources for downstream applications.

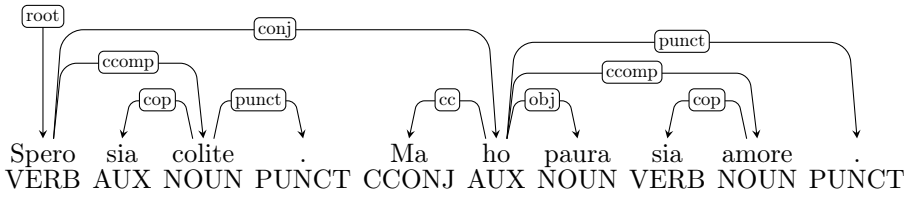
Data and experimental setting. In this section I provide a description of the methods and data I used for addressing the research questions. I followed a binary irony detection task, testing an automatic system on four different languages: English, Spanish, French and Italian. This allowed to investigate whether syntactic structures, independently from the target language, may provide information useful to understand if a message is ironic or not.

In the multilingual experimental setting I took advantage of four datasets that have been made available during the last few years within evaluation campaigns, and all featured at least by a binary annotation for irony. In Table 2.16 for each dataset, I report the target language, the name of the shared task in which it was released and its overview paper, the number of tweets for each class (*ironic* vs. *not ironic*) and the total number of instances, for both training set and test set. The aim is, thus, to determine whether a given tweet is *ironic* or *not ironic*.

language	**dataset**	**train**			**test**		
		ironic	not	total	ironic	not	total
English	SemEval-2018 Task 3 [Van Hee et al., 2018a]	1,923	1,911	3,834	311	473	784
Spanish	IroSvA 2019 [Ortega et al., 2019]	1,600	5,600	7,200	599	1,201	1,800
French	DEFT 2017 [Benamara et al., 2017]	1,947	3,906	5,853	488	976	1,464
Italian	IronITA 2018 [Cignarella et al., 2018b]	2,023	1,954	3,977	435	437	872

Table 2.16: Benchmark datasets used for irony detection (binary task).

Provided that the availability of all the morphological and syntactic knowledge is crucial for performing the experiments described in the rest of this section, I needed to obtain a representation of all the datasets in UD format. With the exception of TWITTIRÒ-UD, described in Section 2.1.2, I obtained the dependency-based annotation for the other corpora by applying *UDPipe*[26] for tokenization, PoS-tagging and parsing. An example drawn from TWITTIRÒ-UD is provided in Figure 2.8.



(*I hope it's colitis. but I fear it's love.*)

Figure 2.8: The dependency-based syntax tree accordingly to the UD format.

Considering that all the datasets used in this part of my work consist of Twitter data, whenever possible, I used resources where this genre, or at least user-generated content of some kind was included as training data for parsing. More precisely, the model for English has been trained on the EWT treebank [Silveira et al., 2014], that for Spanish on both GSD-Spanish [McDonald et al., 2013] and ANCORA corpora [Taulé et al., 2008]. The model for Italian – for the remaining part of IronITA[27] – was trained on PoSTWITA-UD [Sanguinetti et al., 2018] and ISTD treebanks [Simi et al., 2014], while that for French on the GSD-French corpus [McDonald et al., 2013]. I am aware that there actually exists a Twitter treebank for English, i.e., TWEEBANK V2 [Liu et al., 2018b], but it is not fully compliant with the UD format specifications (e.g., it violates the single root constraint posed in UD format). I thus opted for the EWT in order to preserve annotation consistency among resources. The different amount of data used for training UDPipe, the variety of text genres among the datasets, and the fact that the UD annotation of only one dataset has undergone a manual correction – i.e., TWITTIRÒ-UD – can make the quality of the UD data used in this section not entirely homogeneous.

In spite of such disparity in the annotation reliability, and bearing in mind that a higher accuracy in this regard can be crucial, I considered the output provided by the parser for all the languages reasonably satisfactory for the purposes of our study.

Methodology. The main aim of the experiments I present in this section consists in evaluating the contribution to irony detection made by the linguistic information provided in the datasets described above. The task I address is, therefore, a straightforward binary classification task on irony detection, that is, the task for which literature offers baselines (e.g., shared tasks and competitions) and fair-sized annotated datasets for a variety of languages.

[26] http://ufal.mff.cuni.cz/udpipe.

[27] Approximately 1,400 out 4,849 tweets from IronITA are also part of the TWITTIRÒ-UD corpus.

For addressing the task, I performed a set of experiments where several models were implemented exploiting classical machine learning algorithms, deep learning architectures and state-of-the-art language models implemented with the Python libraries *scikit-learn*[28] and *keras*.[29] I tested different sets of pre-trained word embeddings to initialize the neural models, namely *fastText*[30] and a dependency-based *word2vec* proposed by Levy and Goldberg [2014] (*word2vecf*). The latter were trained on the concatenation of all the treebanks available in the UD repository for each considered language.

In order to combine these methods with syntactic features inspired by Sidorov et al. [2014], I used data where not only a binary annotation for irony is applied, but also a morphological and syntactic analysis is available (see Section 2.3.2).

Pre-processing and features. I stripped all the URLs and normalized all characters to lowercase, as it is usually done before the application of sentiment analysis tools.

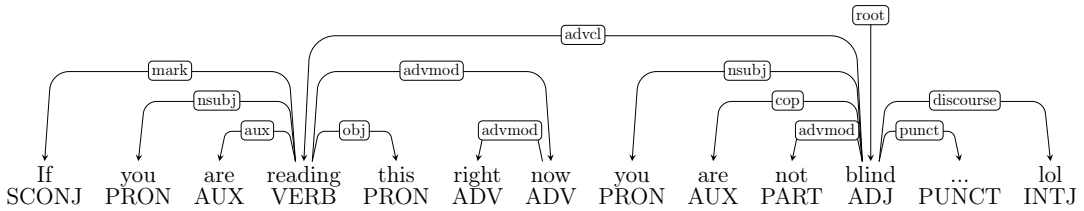


Figure 2.9: Dependency-based syntactic tree of an English tweet.

The description of features as well as the content of the vectors for the syntactic features we developed, referring to the tweet in Figure 2.9, are as follows:

• n-grams: I extracted unigrams, bigrams and trigrams of tokens; e.g., *[If, you, are, reading, ..., If you, you are, are reading, ..., If you are, you are reading, are reading this, ...]*;

• char-grams: I considered the sequence of char-grams in a range from 2 to 5 characters; eg *[If, fy, yo, ou, ..., Ifyou, fyoua, youar, ouare, uarer, ...]*;

• deprelneg: I considered the presence of negation in the text, relying on the morphosyntactic cues present in the UD format. When a negation was present, I appended the correspondent dependency relation in the feature vector. For instance in Figure 2.9, I spot a negation in *[... are **not** blind ...]*, the dependency relation of "not" is `advmod`, therefore, I append it in the feature vector;

• deprel: I built a bag of words of 5-grams, 6-grams and 7-grams of dependency relations as occurring in the linear order of the sentence from left to right; e.g., *[mark nsubj aux obj advmod, nsubj aux obj advmod advmod, ..., advmod advmod nsubj cop advmod root punct, advmod nsubj cop advmod root punct discourse]*;

• relationformVERB: I create a feature vector with all the tuples of tokens that are connected with a dependency distance = 1, by starting from a verb and at the same time I blank the verb itself. For instance, in the example the first verb is "*reading*" and some of the tuples of tokens connected through this verb are, e.g., *[IfVERBthis, youVERBthis,*

[28] https://scikit-learn.org.
[29] https://keras.io/.
[30] https://fasttext.cc/.

areVERBthis, IfVERBnow, youVERBnow, ...];
• relationformNOUN: I applied the same procedure of the feature above but considering nouns as starting points for collecting tuples;
• relationformADJ: in the same fashion of the two features above, I repeated the same procedure for adjectives too;
• Sidorovbigramsform: I created a bag of word-forms (tokens), considering the bigrams that can be collected following the syntactic tree structure (rather than the bigrams that can be collected reading the sentence from left to right).[31] Such that: e.g., *[blind reading, blind you, blind are, blind not, reading if, reading you, ...]*;
• Sidorovbigramsupostag: as the feature above, I created a bag of part-of-speech tags;
• Sidorovbigramsdeprel: as the two features above, I created a bag of words based on dependency relations (*deprels*).

Models. Having as primary goal the exploration of the features described in the previous section and as a case study testing their effectiveness in irony detection, I implemented a variety of models,[32] including the following:

Logistic Regression (LR) - I used the Logistic Regression classifier with the default parameters, exception made for the maximum number of iterations which I set to 5.
Random Forest (RF) - I used the Random Forest classifier with its default parameters.
Gated Recurrent Unit (GRU) - I used a straightforward architecture and set the following hyper-parameters: epochs = 10, learning rate = .0001. The GRU is initialized with random weights and the word embeddings are learned during the training.[33]
GRU+fastText - With the same hyper-parameters as above, I initialized the neural network with specific *fastText* word embedding pre-trained models [Joulin et al., 2016] for each language.
GRU+dependency-based embeddings - In the same way, and maintaining the same setting, I fed the neural network with the dependency-based *word2vec* word embeddings by [Levy and Goldberg, 2014]. These word embeddings are able to capture syntactic information during their training, therefore producing embeddings that are more sensitive to functional similarity than traditional co-occurrence-based word embeddings like *fastText*. I trained a different word embedding model for each language using the concatenation of available UD treebanks for that language.
Multilingual BERT (M-BERT) - Since a multilingual version of the BERT language model is also available for 104 languages including the four I take into account in this study [Cignarella et al., 2020a], I also performed a set of experiments using it. I set the hyper-parameters such that the batch size = 8, the initial learning rate = $1e-5$. I did not set any fixed number of epochs, but rather relied on the *EarlyStopping* function, setting the value of patience = 5.
M-BERT+syntax - With the same hyper-parameters as above, I concatenated the dependency-based features from Section 2.3.2 to those extracted from M-BERT and fed

[31] Please refer to [Sidorov et al., 2013] and [Sidorov, 2014] for more details on this regard.
[32] I also experimented with the SVC using different kernels and the Multilayer Perceptron (MLP), but I report here only the best performing ones.
[33] I experimented also with other RNN architectures but the experiments were not conclusive.

them both into a LSTM neural network.

M-BERT+best_feats - Finally, I experimented maintaining the same setting as the two previous models, here I concatenated a smaller set of features, and not all of them. Namely, I selected only the *best features* resulting from experiments with classical machine learning models, as I will explain in Section 2.3.2.

I performed different sets of experiments training the models on the training sets and evaluating them against the gold test sets made available in the shared tasks referred to in Table 2.16. I drew three main scenarios: a) experiments to select the best features; b) experiments to measure the impact of dependency-based word embeddings; and finally c) experiments conducted with the bidirectional encoder BERT in its multilingual variant infused with syntactic knowledge.

a) Selection of best features. Firstly, in order to understand which of the features provided in 2.3.2 are relevant for the present task, I performed experiments with classical machine learning algorithms. I carried out an evaluation of four models (SVM, LR, RF and MLP) by combining them with all the possible permutations of the feature set and I evaluate them with respect to the macro F1 score (the averaged score between the F1 of the ironic class and the F1 of the non-ironic one).[34] This first step has been crucial for establishing which single feature or group of combined features performs better for each of the involved languages, as summarized in Table 2.17.

language	macro F1	model	n-grams	char-grams	deprel	deprelneg	relation form VERB	relation form NOUN	relation form ADJ	Sidorov bigrams form	Sidorov bigrams deprel	Sidorov bigrams upostag
English	.683	RF		✓			✓		✓		✓	✓
Spanish	.539	RF							✓	✓		
French	.641	LR		✓			✓	✓		✓		✓
Italian	.702	RF		✓	✓		✓	✓		✓	✓	

Table 2.17: Features exploited in the best runs with classical ML algorithms in each language scenario.

From Table 2.17 it emerges how in all the configurations used for achieving the best score at least one dependency-based syntactic feature was exploited and in particular those based on Sidorov's work, i.e., the last three columns of the table.[35] This provides evidence for positively answering to our first research question, since those are the features where the real structure from root to branches of syntactic trees is encoded.

Moreover, Italian is the language where the best score was obtained (F1 = .702). This might provide some hints about our second research question, suggesting that the higher quality of the resource from where syntactic knowledge has been drawn might positively influence the performance. In fact, as previously mentioned, the one for Italian is the only dataset used for this study previously submitted to a careful manual check, after the processing done with UDPipe. On the contrary, the datasets for the other three languages are obtained automatically without further manual revision.

[34] Supplementary material such as code, detailed features and exhaustive tables of results are accessible here: https://anonymous.4open.science/r/16545061-5f63-4d52-a6f7-971128c10e4f/#.

[35] Please refer to Sidorov et al. [2012] and Sidorov et al. [2013] for further details.

Interestingly, in all the configurations *n-grams* and *deprelneg* are not used. Regarding the latter, I noticed that it was a weak feature, and very sparse, thus not able alone to capture a complex phenomenon as irony. Concerning *n-grams*, although, the fact that they are not exploited in the best configurations might point towards the fact that to detect irony a lexical approach is not really sufficient. On an opposite note, it can also be seen how there is no single syntactic feature that was selected in the best performing model for each of the four languages.

One conclusion that can be easily drawn from Table 2.17 is that the field of multilingual research implies a large heterogeneity in results. And this is also one of the huge limitations of working in a multilingual perspective. Indeed, proposing a model that can reach high results on many languages at once is still a more difficult task to address, with respect to similar research that goes in the direction of monolingual approaches.

In some prior work, it is highlighted how while conducting research with datasets in different languages, they could be very different from each other both in how they were collected and in how they were aggregated and selected [Kim et al., 2010, Hajjem et al., 2014]. Multilingual corpora are, indeed, widely exploited within several tasks of NLP. These corpora are principally of two sorts: comparable and parallel corpora. The comparable corpora gather texts in several languages dealing with analogous subjects but are not translations of each other such as in parallel corpora [Karima and Smaili, 2016]. As parallel corpora are more expensive to obtain, because they need a bigger human effort to align corresponding sentences, it makes sense that comparable corpora (sharing the same topic) are still nowadays the best alternative, as it has been done here.

Despite the difficulties and limitations that may arise by working in a multilingual setting, in this step of my research I was able to consider these combination of characteristics as "*best feature sets*" for each language and to exploit them in further experiments (see Section 2.3.2 below in this section).

b) Measuring the impact of dependency-based word embeddings. In the second set of experiments I explored, instead, the potential of two different kinds of word embeddings with regard to each language, namely *fastText* and the dependency-based word embeddings version of *word2vec*, which I trained on the four datasets taking advantage of the UD representation (*dep-based we*).

language	**GRU**		
	—	**+fastText**	**+dep-based we**
English	.648	**.650**	.552
Spanish	.494	**.500**	.404
French	.522	**.567**	.447
Italian	.649	.652	**.659**

Table 2.18: Results obtained combining a GRU architecture and word embeddings.

As it can be seen from Table 2.18, for each language, the best results in terms of macro F1 are obtained adopting either *fastText* or the *dependency-based word embeddings*. Fur-

thermore, I want to highlight that *fastText* seems overall the best configuration in this set of experiments (English, Spanish and French), exception made for Italian in which the best result is obtained with the *dependency-based word embeddings* configuration. I interpret this as a confirmation that external lexical knowledge (fastText) and structured syntactic information (dependency-based word embeddings) are indeed useful for the detection of irony. The positive result obtained in Italian with the *dependency-based word embeddings* configuration gives some hints about the impact that a good quality of the morphosyntactic annotation might provide as previously observed.

The Italian dataset is, I recall, the only one amongst the four other languages that has been manually checked and corrected, as the dataset used to perform experiments almost completely overlaps with one the official UD Italian treebanks (see Section 2.1.2). The other pareses for the other three languages have been automatically obtained through the application of UDpipe pipeline to the datasets. One naïve conclusion that could be derived from this result, is how the quality of syntactic annotation (and manual correction) directly propagates on the results of downstream tasks.

Finally, the most significant insight that we can get from this second set of experiments is that deep learning architectures (in this case GRUs) seem to benefit from the addition of syntactic features. It is my belief that fine-grained syntactic information such as the features I implemented in Section 2.3.2, when added to an already robust architecture, could capture important structures of language and, therefore, boost the performance of the system. I try to make this hypothesis evident by building the third and last experimental setting, as follows.

c) Syntactically-informed BERT for irony detection. Lastly, I performed experiments with the state-of-the-art BERT language model. For each language, I ran the straightforward M-BERT model as anticipated in Section 2.3.2. In a second phase of this setting, I implemented the base architecture by adding the dependency-based syntactic features detailed in 2.3.2 in three different ways in order to have a clear-cut evidence on the actual contribution derived from dependency syntax to irony detection.

language	**shared task best (report & score)**		***SVC***	**M-BERT**		
			+1-grams		**+syntax**	**+best_feats**
English (SemEval-2018)	[Wu et al., 2018]	.705	*.649*	.655	.682 (↑ .027)	.694 (↑ .039)
Spanish (IroSvA 2019)	[González et al., 2019]	.683	*.613*	.663	.668 (↑ .003)	.677 (↑ .014)
French (DEFT 2017)	[Rouvier and Bousquet, 2017]	.783	*.617*	.770	.785 (↑ .015)	.772 (↑ .002)
Italian (IronITA 2018)	[Cimino et al., 2018]	.731	*.578*	.699	.703 (↑ .004)	.687 (↓ .012)

Table 2.19: Results obtained combining M-BERT and dependency-based syntactic features. Green values and arrows pointing up show an increment in performance, while red values and arrows pointing down indicate a performance reduction, with respect to results obtained by the bare architecture.

In Table 2.19 I report the results of the best participating system in each one of the shared tasks (with the reference to their working notes). Furthermore, as a baseline reference measure, I also added the results obtained with a SVC and a bag of words of

unigrams, as it was a baseline proposed in all competitions. Each of the experiments with M-BERT has been performed 5 times with the hyper-parameters previously described in Section 2.3.2 in order to take into account the differences of random initialization, and the average macro F1 score of such number of runs is reported.

It is interesting to see that models implemented with the addition of dependency-based syntactic features obtain results in line with the state of the art in all four language scenarios (see shared tasks results). Moreover, in all of them, the addition of syntactic knowledge (*M-BERT+syntax*) determined an improvement of scores with respect to the models where syntax is not taken into account (*M-BERT*), as highlighted with the green values. Therefore, it seems that syntax plays an important role in the detection of irony and surely deserves further investigation exploring more complex neural architectures.

I finally carried out a third set of experiments, which produced the scores reported in the table as *M-BERT+best_feats*. In these experiments I paired the M-BERT architecture with the best set of features extracted in the very first setting (see bullet point a). What we can observe in this case is that the extraction of the best features, and the subsequent reduction of the dimension of the feature space, is beneficial for English and Spanish. As a matter of fact, using the reduced feature set, we have an increment of the macro F1 score from .682 to .694 (Δ +.012) in English and from .668 to .677 (Δ +.009) in Spanish. On the other hand, French and Italian do not seem to profit from the reduction of available features.

All the above-mentioned conditions prove themselves interesting and surely lead in the direction of further investigation, pointing mainly towards a better understanding of features' behaviour when stacked in a pre-trained language model such as BERT. Furthermore, taking into account that irony is a pragmatic device inherently related to culture and language, the above-listed findings stress the importance of investigating features monolingually in order to provide a solid background for enhancing a multilingual system.

Error Analysis. From a lexical point of view, a qualitative analysis of results shows that even in the presence of clear lexical clues, like specific hashtags, classification errors occur in all datasets. This piece of evidence does not come unattended. It is indeed a confirmation that the classification method proposed here –almost exclusively based on morphosyntactic information– which I developed in order to highlight the impact of syntax alone in the task of irony detection, is not especially influenced by the presence of such clues.

On the other hand, an error analysis based on the results generated from the best runs in the four languages also provides useful evidences about the impact of syntactic features to the detection of irony. The distribution of part-of-speech tags and dependency relations in the whole test sets with respect to their distribution in the misclassified tweets is indeed unbalanced. Namely, a higher number of `SYM`[36] and `X`[37], with respect to PoS-tags, and

[36]See: https://universaldependencies.org/u/pos/SYM.html.
[37]See: https://universaldependencies.org/u/pos/X.html.

`parataxis`[38], `flat`[39] and `expl`[40], with respect to dependency relations, characterizes tweets that are wrongly classified (either false positives or false negatives). Regardless of the variation of the frequency of these morphological categories and syntactic relations in the four different languages, their incidence can be observed in the results. For instance, the presence of tokens tagged as `SYM` in the English misclassified tweets is higher by +22% with regard to the average distribution, in Spanish is +7%, in Italian is +4%, while it does not seem to have particular relevance in the French data.

The categories and relations observed are typically found in user-generated texts and are moreover known as inherently hard to parse for the state-of-the-art UD parsers which, still nowadays, are mostly trained on standard texts and do not take into account idiosyncrasies of social media texts (emojis, hashtags, non-standard word forms, etc.). A debate regarding this one and other related issues is indeed ongoing in the UD community and a discussion can be seen in a recent work by Sanguinetti et al. [2020], in which I also participated as active member of the UD community (especially those people treating with UD and user-generated content), as described in Chapter 1.

Indeed, the authors discuss various UGC-related issues such as: simplification, contraction, oversplitting, medium-dependent phenomena and context-dependency. It is acknowledged how some texts, which may be particularly hard to tokenize, lemmatize and/or parse could be treated within the UD framework. Among the examples described in the paper, the sentence "*I want sumthn that's gonna last*" for instance, reports an example of an unsplit contraction: *gonna* [Sanguinetti et al., 2020, Fig. 2]. Depending on the case whether the creators of a treebank decide to split or not such cases, and therefore provide two different choices according to the UD annotation framework, some semantic information regarding the minimal parts of 'gonna' (i.e., 'go' + 'to') might be saved or lost. Therefore, imagining that saving or losing that kind of information could be essential also for the effectiveness of performing other downstream tasks (irony detection for instance), the authors open up a debate and provide some advice and preferable guidelines to be followed, stressing the importance of making choices to reflect the different goals of each treebank creator.

2.4 Concluding remarks on irony detection

In the present chapter, I have focused on the problem of irony detection, that I took into account as first case study, in order to investigate the impact of morphosyntactic information in Sentiment Analysis tasks.

Firstly, I surveyed the related work on the topic, devoting particular attention to the description of shared tasks organized within evaluation campaigns organized in the last few years. In the same section, I provided a broad panorama of approaches and machine learning techniques that are typically used in this field, naming the innovative approaches exploited by participants of the above mentioned shared tasks and outlining

[38] See: https://universaldependencies.org/u/dep/parataxis.html.
[39] See: https://universaldependencies.org/u/dep/flat.html.
[40] See: https://universaldependencies.org/u/dep/expl.html.

the state-of-the-art models. After having stressed the meaningful impact of shared tasks in the field, in the second section, I presented an overview of the IronITA 2018 shared task (*Irony Detection in Italian tweets*), organized within EVALITA 2018.

Thirdly, I focused on the description of the TWITTIRÒ corpus, which I developed within my PhD studies, and that has also been in part distributed for training participating systems in the above-mentioned shared task on *Irony Detection in Italian tweets*. Starting from a nucleus of some hundreds of tweets collected during the development of my master thesis, I have later on enhanced the annotation of this corpus within the duration of my PhD up to now, i.e., the TWITTIRÒ-UD treebank, which is my contribution to the Universal Dependencies project.

My direct involvement in the organization of a shared task and in the development of a novel resource allowed a meaningful improvement of my awareness of the importance of formalizing the irony detection problem, turning it into a computational task, also paving the way for a deeper understanding of the major approaches applied to irony detection and for my participation in some shared task.

Lastly, in a more technical section, I provided the implementation details and evaluation of machine learning systems that exploit morphosyntactic knowledge by taking advantage from the Universal Dependencies annotation format. Therefore, I described my participation in the IroSVA 2019 irony detection shared task (*Irony Detection in Spanish Variants*) and presented the first introductory work that moved in the direction of a syntax-based approach for dealing with irony. Following this, I described one of my last works on the further implementation of the above-mentioned system (see Section 2.3.2). I provided more accurate descriptions of feature engineering and system development, together with a wider scenario of results and discussion points regarding primarily the impact of syntax on the task of irony detection, which is one of the contributions that I present with this thesis.

In particular, throughout the whole chapter I have laid the useful background in order to fully investigate the phenomenon of irony and to answer the following research questions:

- **RQ-1** *Could features derived from morphology and syntax help to address the task of irony detection?*
- **RQ-2** *To what extent does using resources such as treebanks for training NLP models improve the performance in irony detection?*

Current findings provide a meaningful support to the hypothesis that morphosyntactic knowledge extracted from treebanks can be usefully exploited for addressing the irony detection task. In particular, they pave the way for a further investigation where the combination of a dependency-based syntactic approach and state-of-the-art neural models can be explored. The novel contribution of my work mainly consists in the exploitation of these features to create vectorial representations of texts; all of them have been made available thanks to the application of the UDpipe pipeline together with the subsequent generation of the dependency trees corresponding to the items of the datasets. The

application of UD allowed to investigate whether syntactic structures, independently from the target language, may provide information useful to understand if a message is ironic or not, according to a perspective of improving the explainability of results.

Thanks to dependency syntax it is possible to grasp connections among words that are not captured by n-grams or word embeddings. If we had only considered approaches such as those that take into account only the words that are in the immediate proximity of each other, the ironic meaning caused by the clash of two words that are fare away from each other, would have been lost. In fact, with dependency-based features is possible to capture the information in which words are syntactically related to each other also if they have a long-distance relation. Indeed, the importance of capturing the syntactic dependency between words that are distant from one another has been highlighted in a previous section (see Section 2.3.1) through the following example: '*He lanzado un relator al aire y no vuelan...*' (Trans: I have launched a reporter into the air and they do not fly...), where the relation between the words '*lanzado*' (launched) and '*no vuelan*' (not fly) is crucial to intend the ironic meaning of the sentence.

The results suggest that fine-grained dependency-based syntactic information is indeed informative for the task of irony detection, in fact the addition of syntactic knowledge in a complex architecture such as multilingual BERT determined an improvement of scores with respect to the models where syntax is not taken into account (see Section 2.3.2). I believe that dependency syntax, not only allows to reach better results in terms of metrics, but could surely shed some light on the explainability of a difficult pragmatic phenomenon such as irony.

In the future, in order to validate these findings, I would like to propose a wider experimental setting, taking into account other language scenarios. For instance, testing the approach on other languages for which both irony-annotated datasets and UD resources are available. A starting point towards an expansion could be Arabic, for which a dataset annotated for irony is available [Ghanem et al., 2020] and also a treebank of user-generated content exists [Seddah et al., 2020]. Furthermore it would be interesting to deal with a non-European language. Another option could be to expand our investigation to German, since at least one corpus[41] and a treebank of German tweets [Rehbein et al., 2019] exist, and can be used for training a dependency-based model.

With this brief paragraph, I conclude the presentation of the work done regarding the automatic detection of irony, in particular investigating the contribution that might be given by the usage of morphosyntactic information. In the following chapter, I will present the second task taken as a case study for testing the hypothesis of the present thesis: automatic stance detection. Once again, I will describe the task in detail, and I will explore the impact of morphosyntactic information and provide an exploration of stance in a multilingual scenario.

[41] http://kti.tugraz.at/staff/rkern/courses/kddm2/2018/reports/team-27.pdf.

Chapter 3

Stance detection

---

stance

noun [C]

UK /stɑːns/ US /stæns/

A PARTICULAR WAY OF THINKING ABOUT SOMETHING, ESPECIALLY WHEN THOSE OPINIONS ARE EXPRESSED PUBLICLY OR OFFICIALLY

(CAMBRIDGE DICTIONARY)

In this chapter, I will focus on the second case study that I took as object of analysis in the present thesis: Stance Detection. Similarly to what I have done in the previous chapter on irony detection, also here in the first place I survey the related work on the topic, dedicating peculiar attention to evaluation campaigns and shared tasks organized in the last decade. After that, I will focus on the availability of corpora annotated accordingly to stance; we will see in detail how stance detection is a broad topic, that embraces many similarities with the more general-purpose task of Sentiment Analysis.

Secondly, I will focus on the overview of the *SardiStance 2020* shared task, which I organized together with some colleagues from the Università degli Studi di Torino and from the Universitat Politècnica de València in the framework of the *7th Evaluation campaign of Natural Language Processing and speech tools for the Italian language (EVALITA 2020)* within the last year of my PhD studies. I will describe this task in order to have a zoom in on the issues that may arise while dealing with the creation of a dataset, the application of an annotation scheme and the organization of a shared task. I thus outline the strengths and difficulties of such activity (see Section 3.1.1). My direct involvement in the organization of a shared task and in the development of a novel resource allowed a meaningful improvement of my awareness of the importance of the formalization of the stance detection problem, turning it into a computational task, also paving the way for a deeper understanding of the main approaches applied in this field.

Later on, in Section 3.2, I will focus on the vast panorama of approaches and machine learning techniques that have been used to address this problem, providing a description

of the state-of-the-art models, which also allow us to tackle the problem of stance detection in a multilingual perspective. I will also provide a more detailed description of my participation in the *StanceCat* shared task at *IberEval 2017*, in which I had the first opportunity to get in touch with a formal modeling of stance detection as a multi-class classification problem (see Section 3.2.1). The participation in the shared task, occurred in the very first months of my PhD, and together with other peer students, I had the first experience of creating an automatic system for resolving a sentiment analysis task; an experience that paved the way for successive more elaborate studies on the matter.

In the third section of this chapter regarding the topic of stance detection, I focus on the approaches exploiting the knowledge made available in the Universal Dependencies format. In particular, I firstly describe my participation in the *RumorEval* shared task at *SemEval 2019* in which a preliminary study of features based on dependency syntax has been proposed. Therefore, I provide the implementation details and performances of the systems that I submitted as a participant, especially stressing the contribution given by morphosyntactic information (see Section 3.3.1).

Finally, in Section 3.3.2, I mirror the multilingual experiments previously done for irony detection (see Chapter 2, in particular Section 2.3.2), and apply the same framework to the problem of stance detection. Therefore, precisely as it has been done in the previous chapter on irony detection, I present the results obtained with regard to experiments performed in four different languages: English, Spanish, French and Italian. In this chapter I experiment also with a fifth language, i.e., Catalan, due to the availability of a benchmark dataset also in this language (extracted from the dataset of the same shared task that also made the Spanish dataset available). Furthermore, I study the phenomenon of stance with respect to six different targets – one per language, and two different for Italian (*Constitutional Referendum* and *Sardines Movement*) – with a variety of complex architectures that primarily exploit morphosyntactic knowledge represented throughout the format of Universal Dependencies.

3.1 Shared tasks and corpora for stance detection

Recently, the task of monitoring people's opinion towards particular targets in political topics or public life debates has grown, thus leading to the creation of a novel area of investigation named *Stance Detection* (SD). Research on this topic has an impact on different aspects of everyone's life such as public administration, policy-making, marketing strategies and security. In fact, through the constant monitoring of people's opinion, desires, complaints and beliefs on political agenda or public services, administrators could better meet population's needs.

Exactly like Sentiment Analysis, also SD has been applied in several domains, for instance, to discover the reputation of an enterprise, what is the general public thought regarding a political reform, if costumers of a fashion brand are happy about the customer service etc.. Nevertheless, whereas the aim of sentiment analysis is at categorizing texts according to a notion of polarity (positive, negative or neutral), the aim of SD consists in classifying texts according to the attitude they express towards a given target of interest.

Figure 3.1, taken from Küçük and Can [2020], illustrates the relationships occurring among the different tasks and subtasks in the field of Sentiment Analysis, putting at the center of attention SD.

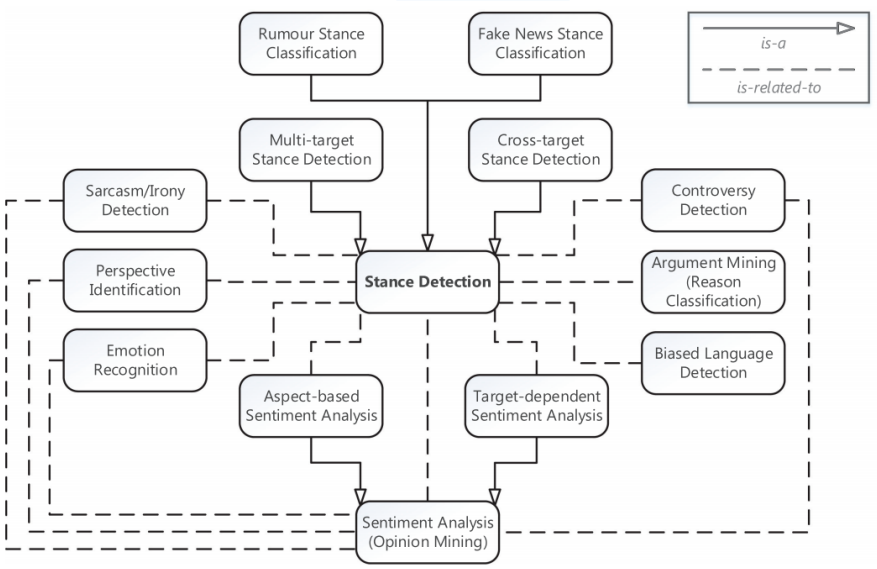


Figure 3.1: Problems and subproblems in SA related to stance detection.

In literature, we distinguish at least two different tasks that are both referred to as "Stance Detection". The first is the TARGET-SPECIFIC STANCE CLASSIFICATION, described as the task of automatically determining whether the author of a text is in favour, against, or neutral towards a given target. The first shared task of this type, from which we also inherit the formal definition of the task, was held for English at SemEval in 2016, i.e., *Task 6 "Detecting Stance in Tweets"* [Mohammad et al., 2016]. The second type of stance classification, more general-purpose, is the OPEN STANCE CLASSIFICATION, usually indicated with the acronym SDQC, by referring to the four categories exploited for indicating the attitude of a message with respect to a rumour: Support (S), Deny (D), Query (Q) and Comment (C) [Aker et al., 2017].

It is also worth mentioning that, nowadays, the spreading of fake news and hoaxes online has become a huge problem. Stance detection is considered as a useful first stage to determine whether a given story is real or fake. The Fake News Challenge [FNC, 2017], in a first stage (FNC-I), proposed a classification problem where the stance towards a claim of a news headline should be identified. The input was in the form of news headline and a news body pair (where the headline and body parts may belong to different news articles). The output should be in the form of a category label from this set: Agrees, Disagrees, Discusses (the same topic), Unrelated.

Not surprisingly the SD tasks are mostly based on data extracted from social media or user-generated content and they are about politics and public life topics. On the one hand, it is on social media that people spontaneously express opinions, desires, com-

plaints, beliefs and outbursts. On the other hand, politics and public life are among the topics mainly discussed by users in social media. In these choices the possible relevance of SD techniques for policy makers and public administrators is also mirrored, e.g., for better meeting population's needs and preventing feelings of dissatisfaction and extreme reactions of hostility and anger. A further motivation for collecting texts especially from social media, in particular Twitter and Reddit, is that they are a great source of freely available data. Beside the usefulness, it would be naive not to mention the other side of the coin of applications based on stance detection. Indeed, such applications could surely also be exploited by authoritarian states as a form of mass' control, which, of course, might be a very dangerous scenario.

Although SD is a fairly recent research topic, considerable efforts have been devoted to the creation of stance-annotated datasets, not only as training and test sets for organizing contests, but also for independent researches. Through a recent survey on SD (see Table 6 in Küçük and Can, 2020) I came across the existence of datasets (of different text types such as tweets, posts in online forums, news articles, or news comments) for at least eleven languages: Arabic, Catalan, Chinese, Czech, English, English-Hindi, Italian, Japanese, Russian, Spanish, and Turkish; most of which are made publicly available.

Although some efforts have recently been made to develop annotated data in other languages, there still is a telling lack of resources to facilitate multilingual and cross-lingual research on stance detection. Furthermore, as stance is a highly domain- and topic-specific phenomenon, the need for annotated data is specially demanding. In a very recent work, Zotova et al. [2021] present a method to obtain multilingual datasets for stance detection in Twitter. Instead of manually annotating on a per tweet basis, they leverage user-based information to semi-automatically label large amounts of tweets.

The creation of the field of SD as an independent task, detached from sentiment analysis, is broadly attested by the continuous organization of shared tasks focusing on this topic within NLP evaluation campaigns in several languages, during the last years. In this section, I will outline the most significant events that took place regarding it, following a chronological order. Below, in Table 3.1, I summarize the most relevant information regarding the shared tasks mentioned in this section, grouping them by language.

The earliest competition on SD is SemEval-2016 Task 6 [Mohammad et al., 2016]. It consisted in detecting the orientation in favour or against six different targets of interest: "Hillary Clinton", "Feminist Movement", "Legalization of Abortion", "Atheism", "Donald Trump", and "Climate Change is a Real Concern". The competition had two subtasks: in subtask A (supervised stance detection), an annotated training dataset of 2,814 tweets and a test dataset of 1,249 tweets are provided for a total of six targets, while in subtask B (weakly supervised stance detection), only a large unlabeled dataset (of approximately 78,000 tweets) and a smaller test data (of 707 tweets) for another target are provided to the participants for training and testing, respectively, without any annotated training data. In the same year, another SD shared task similar to that of SemEval-2016 has been organized at the International Conference on Natural Language Processing and Chinese Computing (NLPCC-ICCPOL 2016) for detecting stance in Chinese microblog texts (from the Sina Weibo platform) as described in Xu et al. [2016]. Here, two subtasks

were organized: the first, allowing submission of a supervised stance detection system, and the second requesting a submission of a unsupervised system trained on a set of unlabeled microblog texts.

The following year, 2017, an evaluation for SD systems was proposed at *IberEval 2017* for both Catalan and Spanish: *StanceCat 2017* [Taulé et al., 2017] where the target was only one, i.e., "Independence of Catalonia". The organizers provided 5,400 tweets in Spanish and 5,400 in Catalan, in both languages they divided 75% for training and the remaining 25% for testing.

language	dataset	focus	source
English	SemEval-2016 Task 6 [Mohammad et al., 2016]	target-specific	Twitter
	SemEval-2017 Task 8 [Derczynski et al., 2017]	rumours	Twitter
	SemEval-2019 Task 7 [Gorrell et al., 2019]	rumours	Twitter, Reddit
	Fake News Challenge Stage 1 [FNC, 2017]	fake news	news headlines
Chinese	NLPCC-ICCPOL 2016 - Task 4 [Xu et al., 2016]	target-specific	Sina Weibo
Spanish and Catalan	StanceCat - IberEval 2017 [Taulé et al., 2017]	target-specific	Twitter
	MultiStanceCat - IberEval 2018 [Taulé et al., 2018]	target-specific	Twitter
Spanish and Basque	VaxxStance - IberLEF 2021	target-specific	Twitter
Italian	SardiStance - EVALITA 2020 [Cignarella et al., 2020b]	target-specific	Twitter

Table 3.1: Shared tasks and datasets for stance detection

Always in 2017 at SemEval the first shared task of the type 'open stance classification' in the context of rumour detection was presented at *RumorEval 2017*. The organizers provided 5,568 tweets in English, for the classification according the SDQC scheme (support, deny, query and comment).

In 2018 the follow-up edition of StanceCat was organized, regarding the target "*Catalan 1st of October Referendum*" – i.e., *MultiStanceCat* [Taulé et al., 2018] – encouraging stance detection with multimodal approaches. The organizers proposed three settings where different sources of information were allowed: a) only the information appearing in the tweet under evaluation; b) the information included in the message and the contextual information, and c) the message and contextual information, as well as images downloaded from the authors timeline. One year later, in 2019, the sequel of the *RumorEval* shared task at SemEval 2019 has been organized. In this edition the organizers also provided Reddit posts, and not only tweets, thus offering more diversity in the types of users, more focused discussions and longer texts. The first pioneer SD task for Italian has been organized in the second semester of 2020.[1] *SardiStance* has been presented, in December 2020, within EVALITA 2020, the *7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*. As organizing this event improved my awareness of the problems that can arise within the automatic detection of stance, I dedicate the following section (Section 3.1.1) to describe the process of its organization, the participating systems, focusing on the new state-of-the-art approaches for Italian and the lesson learned.

To sum up, SD has become a standalone task detached from sentiment analysis in

[1] http://di.unito.it/sardistance2020.

2016, after a dedicated competition was organized as SemEval 2016. Furthermore, some researchers, recently have also proven that stance and sentiment present an orthogonal relationship [AlDayel and Magdy, 2021].

In the last five years both the NLP scientific community as well as the network science community have invested many efforts to the development of datasets on different topics and on different languages. Stance detection resulted being a very important task, sharing a variety of characteristics that could be of help also to other research fields, such as fake news analysis and rumor analysis.

Nowadays, the main languages that have been involved in the study of stance detection are currently: English, Spanish, Catalan, Chinese, Italian and Arabic. Although, the most recent shared task regarding stance detection, is currently being organized for Spanish and Basque on a very controversial and trendy topic, namely, the *Antivaxxers movement* and will take place in September 2021.[2] The organizers propose three different participation tracks: open, closed and zero-shot. Inspired by the recently held *SardiStance 2020* shared task, the so-called "closed track" will include two evaluation settings per language: *textual SD* and *contextual SD*. With the organization of this last competition, which is – among other things – investigating the role of contextual information in stance detection, it seems that the research community is more and more rowing towards the merger of techniques derived from NLP and network science. Indeed, the joint efforts of these two communities has already proven beneficial in late work (see Section 3.2).

3.1.1 Organization of *SardiStance 2020*

With this task proposal we wanted to invite participants to explore features based on the textual content of the tweet, such as structural, stylistic, and affective features, but also features based on contextual information that does not emerge directly from the text, such as for instance knowledge about the domain of the political debate or information about the user's community. Overall, we proposed two different subtasks:

- **Task A - Textual Stance Detection:**
  The first task is a three-class classification task, where the system has to predict whether a tweet is in FAVOUR, AGAINST or NONE towards the given target (following the guidelines below, as in Mohammad et al. [2016]), exploiting only textual information, i.e., the text of the tweet.

- **Task B - Contextual Stance Detection:**
  The second task is the same as the first one: a three-class classification task where the system has to predict whether a tweet is in FAVOUR, AGAINST or NONE towards the given target. Here participants could access a wider range of contextual information based on the **post** such as: the number of retweets, the number of favours, the number of replies and the number of quotes received to the tweet, the type of posting source (e.g., iOS or Android), and date of posting. Furthermore we shared (and encouraged its exploitation) contextual information related to the

[2] https://vaxxstance.github.io/.

user, such as: number of tweets ever posted, user's bio, user's number of followers, user's number of friends. Additionally we shared users' contextual information about their **social network**, such as: friends, replies, retweets, and quotes' relations. The personal ids of the users were anonymized but their network structures were maintained intact.

Collection and annotation of the data. We chose to gather the data from the social networking Twitter due to the free availability of a huge amount of users' generated data and because it allowed us to explore different types of relations among the users involved in a debate. We collected around 700K tweets written in Italian about the "Movimento delle Sardine" (*Sardines movement*[3]), retrieving tweets containing the keywords "sardina", "sardine", and the homonymous hashtags.

Furthermore, we collected all the conversation threads in which the said tweet belongs to, iteratively following the reply's tree. We also collected the quoted tweets and the list of all the retweets of each previously recovered tweet, obtaining about 1M tweets. Finally, we collected the friend list of all the users included in the annotated dataset.

The tweets were gathered between the 46th week of 2019 (November) and the 5th week of 2020 (January), corresponding to a 12 weeks time-window. Through the experience matured as participants in previous shared tasks of SD, and in order to reduce noise in text, we collected data taking into account the following constraints: only one tweet per author for each week, no retweets, no replies, no quotes, no tweets containing URLs, no tweets containing pictures or videos.

Then, we included only Italian tweets posted using a limited number of "sources" (utilities used to post the tweet, such as iOS, Android, etc...) in order to avoid to include pre-written tweets posted using a *Tweet button*.[4] Furthermore, we validated that all the collected tweets presented a *Jaccard similarity coefficient* < 0.8. From about 25K filtered tweets, we finally randomly selected around 300 tweets for each week (only the first week of 2020 does not reach 300 tweets), thus obtaining 3,600 tweets in total.

We created a web platform for annotation purposes (see Figure 3.2) in order to facilitate the labelling task to the annotators, unifying the visualization mode and shuffling the tweets in a random order.[5] Twelve different native Italian speakers with an interest for news and politics were involved in the annotation, according to detailed guidelines completed with examples[6] we provided for supporting them. We randomly shuffled the annotators and matched them into 66 pairs in which each pair would annotate 55 tweets. As a result, each annotator labelled 605 tweets independently and each tweet was annotated by two annotators, who had to choose among four different labels: AGAINST, FAVOUR, NONE/NEUTRAL and OUT OF TOPIC.

Furthermore, as it can also be seen in Figure 3.2 (*Tonight we are all sardines in Bologna #bolognanonsilega*), we asked the annotators to mark whether the tweet was

[3] https://en.wikipedia.org/wiki/Sardines_movement.

[4] https://developer.twitter.com/en/docs/twitter-for-websites/tweet-button/overview.

[5] In this way, each annotator was seeing emojis – which, we believe are essential in order to understand the correct stance – in the same way of the other annotators independently of the device used.

[6] Guidelines for annotation: http://di.unito.it/stanceannotationguidelines.

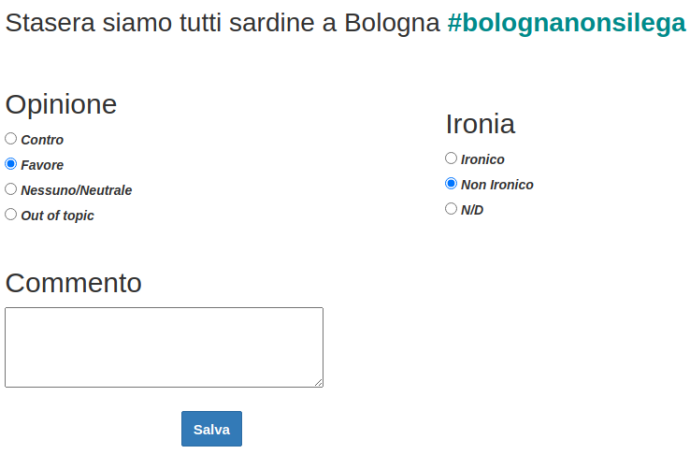


Figure 3.2: Platform for the annotation of tweets.

IRONIC or NOT IRONIC. Ultimately, we were not able to obtain satisfactory results on this end, so we did not include information regarding irony in the task. After the shared task took place,[7] I spared some time in order to complete the annotation of irony in parallel with the already existing annotation of stance, and it is something that I will explore and describe in Chapter 4 of the present thesis, where I connect the dots and finally link the pragmatic phenomenon of irony and stance detection.

At the end of a first phase of annotation we obtained: 2,256 tweets in agreement, with a clear decision on one of the three main classes, 917 tweets in *light disagreement* (i.e., FAVOUR vs. NEUTRAL or AGAINST vs. NEUTRAL), and the remaining 457 tweets in *strong disagreement* (i.e., FAVOUR vs. OUT OF TOPIC) or considered as out of topic by the majority of annotators. While the latters were immediately discarded, we proceeded in the resolution of those 917 tweets, whose disagreement was deemed "light", in order to obtain a bigger dataset, by assigning them to novel annotators of the same group.

In Table 3.2 the distribution of such instances is shown accordingly to the training set and the test set and in Table 3.3 some tweets are reported as example for each class.

TRAINING SET			TEST SET		
AGAINST	FAVOUR	NONE	AGAINST	FAVOUR	NONE
1,028	589	515	742	196	172
2,132			**1,110**		

Table 3.2: Distribution of tweets.

Participants, results and evaluation. A total of 12 teams, both from academia and industry participated in at least one of the two tasks of SardiStance. In Table 3.4 the teams are listed in alphabetical order.

Submissions have been ranked by the averaged F1-score over the two classes, accord-

[7] EVALITA 2020, and *SardiStance* final workshop were held on December 17th, 2020.

tweet_id	tweet_text	stance_label
1	Non ci credo che stasera devo andare in teatro e non posso essere fra le #Sardine #Bologna #bolognanonsilega *I cannot believe that tonight I have to go to the theater and I cannot be together with the #Sardines #Bologna #bolognanonsilega*	FAVOUR
2	LE SARDINE IN PIAZZA MAGGIORE NON SONO ITALIANI SE LO FOSSERO NON SI METTEREBBERO CONTRO LA DESTRA CHE AMA L'ITALIA E VUOLE RIMANERE ITALIANA *THE SARDINES IN PIAZZA MAGGIORE ARE NOT ITALIAN IF THEY WERE THEY WOULD NOT FIGHT AGAINST THE RIGHT THAT LOVES ITALY AND WANTS TO REMAIN ITALIAN*	AGAINST
3	Mi sono svegliato nudo e triste perché a Bologna, tra salviniani e antisalviniani, non mi ha cagato nessuno. *I woke up naked and tired because in Bologna, among salvinians and antisalvinians, no one paid attention to me.*	NONE

Table 3.3: Some examples extracted from the dataset.

team name	institution	report	task
deepreading	UNED, Spain	Espinosa et al. [2020]	A, B
GhostWriter	You Are My Guide, Italy	Bennici [2020]	A, B
IXA	UPV/EHU, Spain	Espinosa et al. [2020]	A, B
MeSoVe	ISASI, Italy	-	A
QMUL-SDS	QMUL-SDS-EECS, UK	Alkhalifa and Zubiaga [2020]	A, B
SSN_NLP	CSE Department/SSNCE, India	Kayalvizhi et al. [2020]	A
SSNCSE-NLP	SSN College of Engineering, India	Bharathi et al. [2020]	A, B
TextWiller	UNIPD, Italy	Ferraccioli et al. [2020]	A, B
UNED	UPV/EHU and UNED, Spain	Espinosa et al. [2020]	B
UninaStudents	UNINA, Italy	Moraca et al. [2020]	A
UNITOR	UNIROMA2, Italy	Giorgioni et al. [2020]	A
Venses	UNIVE, Italy	Delmonte [2020]	A

Table 3.4: Participants and reports.

ing the following equation: $F1_{avg} = (F1_{favour} + F1_{against})/2$. Furthermore, we computed a baseline using a simple machine learning model, for Task A: a Support Vector Classifier based on token unigram features. We computed a second baseline for Task B: a system based on our previous work on stance detection: a Logistic Regression classifier paired with token n-grams features (unigrams, bigrams and trigrams), plus features based on a binary one-hot encoding representation of the communities extracted from the network of retweets and the network of friends (see the best system for Italian, in Lai et al. [2020a]).

Table 3.5 shows the results for the textual stance detection task, which attracted 22 total submissions from 11 different teams. Since the only two systems in an unconstrained setting were submitted by the same team we decided not to create a separate ranking for them, but rather to include them in the same ranking, and marking them with a different color (gray in Table 3.5).

The best results are achieved by the UNITOR team that, with an unconstrained, ranked

team name	run	F1-score			
		AVG	AGAINST	FAVOUR	NONE
UNITOR	1	**.6853**	.7866	**.5840**	.3910
UNITOR	1	**.6801**	.7881	**.5721**	.3979
UNITOR	2	.6793	**.7939**	.5647	.3672
DeepReading	1	.6621	.7580	.5663	**.4213**
UNITOR	2	.6606	.7689	.5522	.3702
IXA	1	.6473	.7616	.5330	.3888
GhostWriter	1	.6257	.7502	.5012	.3810
IXA	2	.6171	.7543	.4800	.3675
SSNCSE-NLP	2	.6067	.7723	.4412	.2113
DeepReading	2	.6004	.6966	.5042	.3916
GhostWriter	2	.6004	.7224	.4784	.3778
UninaStudents	1	.5886	.7850	.3922	.2326
baseline		*.5784*	*.7158*	*.4409*	*.2764*
TextWiller	1	.5773	.7755	.3791	.1849
SSNCSE-NLP	1	.5749	.7307	.4192	.3388
QMUL-SDS	1	.5595	.7091	.4099	.2313
QMUL-SDS	2	.5329	.6478	.4181	.3049
MeSoVe	1	.4989	.7336	.2642	.3118
TextWiller	2	.4715	.6713	.2718	.2884
SSN_NLP	1	.4707	.5763	.3651	.3364
SSN_NLP	2	.4473	.6545	.2402	.1913
Venses	1	.3882	.5325	.2438	.2022
Venses	2	.3637	.4564	.2710	.2387

Table 3.5: Results Task A.

as 1st position with $F1_{avg} = 0.6853$. The best result for the constrained runs is achieved once again by the UNITOR team with $F1_{avg} = 0.6801$. The best results for the two main classes AGAINST and FAVOUR are obtained by the three best systems of the ranking, which are all submissions by the team UNITOR. On the other hand, though, the Deepreading team, ranking as 4th, has obtained the best F1-score for the NONE class, with $F1_{none} = 0.4213$.

Analogously, Table 3.6 shows the results for the contextual stance detection task, which attracted 13 total submissions from 7 different teams.
The best scores are achieved by the IXA team that with a constrained run obtained the highest score of $F1_{avg} = 0.7445$. The best F1-score for the main classes AGAINST and FAVOUR is achieved by the team ranked 1st, IXA, team with $F1_{against} = 0.8562$, and $F1_{favour} = 0.6329$, respectively. Once again, the Deepreading team, ranking 3rd and 4th, has obtained the best F1-score for the NONE class, with $F1_{none} = 0.4251$. Almost all participating systems show an improvement over the baseline, which was computed using a Logistic Regression classifier paired with token n-grams features (unigrams, bigrams and trigrams), features based on the network of retweets, and features based on the network of friends [Lai et al., 2020a].

Discussion. The exploration and subsequent discussion of the participating systems has been particularly interesting. Exploring the state-of-the-art approaches for SD in Italian, having co-organized this shared task, gave me the opportunity of collecting experiences

team name	run	F1-score			
		AVG	AGAINST	FAVOUR	NONE
IXA	3	**.7445**	**.8562**	**.6329**	.4214
TextWiller	1	.7309	.8505	.6114	.2963
DeepReading	1	.7230	.8368	.6093	.3364
DeepReading	2	.7222	.8300	.6143	**.4251**
TextWiller	2	.7147	.8298	.5995	.3680
QMUL-SDS	1	.7088	.8267	.5908	.1811
UNED	2	.6888	.8175	.5600	.2455
QMUL-SDS	2	.6765	.8134	.5396	.1553
SSNCSE-NLP	2	.6582	.7915	.5249	.3691
SSNCSE-NLP	1	.6556	.7914	.5198	.3880
baseline		*.6284*	*.7672*	*.4895*	*.3009*
GhostWriter	1	.6257	.7502	.5012	.3810
GhostWriter	2	.6004	.7224	.4784	.3778
UNED	1	.5313	.7399	.3226	.2000

Table 3.6: Results Task B.

from different researchers and study firsthand new methodologies applied to such task, and thus, be inspired for my personal research on SD.

System architecture – Among all submitted runs in Task A, I counted a great variety of architectures, ranging from classical machine learning classifiers, to recent neural approaches, and also more old-fashioned statistically-based models. For instance, regarding the use of classical ML, the team UninaStudents used a SVM, and the team MeSoVe used Logistic Regression in one run. Regarding the use of neural networks, the QMUL-SDS team used bidirectional-LSTM, a CNN-2D, and a bi-LSTM with attention. Also SSN_NLP exploited the LSTM neural network.

Four teams exploited different variants of the BERT model: Ghostwriter used AlBERTo[8] trained on Italian tweets, IXA used GilBERTo[9] and UmBERTo[10], while UNITOR adopted only this latter model. Finally the Deepreading team made use of transformers such as BERT XXL[11] and XML-RoBERTa[12], paired together with linear classifiers. TextWiller is the only team to have exploited the *xg-boost* algorithm, and ItVenses relied on supervised models, based on statistics and semantics. The UNED team proposed instead a voting system among the output of different models.

For what concerns Task B, almost all the teams enriched the models they submitted in Task A, by enhancing them with contextual information that was made available in Task B. UNED, DeepReading, and TextWiller exploited the *xg-boost* algorithm selecting different features from contextual data. The language model BERT was used in different variants by SSNCSE-NLP, DeepReading, and IXA. In particular, the last two teams proposed three voting based ensemble methods that use two or more models that also exploit the *xg-boost* algorithm. Furthermore, the neural network framework proposed by QMUL-SDS

[8] https://github.com/marcopoli/AlBERTo-it.
[9] https://github.com/idb-ita/GilBERTo.
[10] https://huggingface.co/Musixmatch/umberto-commoncrawl-cased-v1.
[11] https://huggingface.co/dbmdz/bert-base-italian-xxl-cased.
[12] https://huggingface.co/transformers/model_doc/xlmroberta.html.

exploits and combines four different embedding methods into a dense layer for generating the final label using a *softmax* activation function.

The wide and varied exploration of such architectures by the participating teams, and their respective results obtained, allowed me to understand how all kinds of approaches could be valid for the detection of stance: from classical ML algorithms, spacing through deep learning architectures and till the newest language models.

Features – Besides having explored a variety of system architectures, the teams participating in Task A, also used many different textual features, in the most of cases based on word n-grams or character n-grams. MeSoVe and TextWiller additionally engineered features based on emoticons. The team UNED, in one of their runs, proposed a system relying on psychological and social features, while UninaStudents proposed features based on unigrams of hashtags. Interestingly, UNITOR added special tags to the texts, which are the result of a classification with respect some so-called "auxiliary task". In particular, they trained three classifiers based respectively on SENTIPOLC 2016 [Barbieri et al., 2016] for sentiment analysis classification, on HaSpeeDe 2018 [Bosco et al., 2018] for hate speech detection, and on IronITA 2018 [Cignarella et al., 2018b] for irony detection; and they added three tags to each instance of the SardiStance datasets with respect to these three dimensions: sentiment, hate and irony. ItVenses proposed features collected automatically from a unique dictionary list, frequency of occurrence of emojis and emoticons, and semantic features investigating propositional level, factivity and speech act type.

Surprisingly, in Task B, not every team took full advantage of contextual information. For example, SSNCSE-NLP only exploited the number of friends and the number of quotes and friends. UNED exploited some features based on the metadata of the tweets (such as length of text, date/time of posting, etc...) in addition to psychological and emotional features based on the valence-arousal scheme and on the "big five" approach [Espinosa et al., 2020]. Other teams exploited different approaches for learning the vectorial representations of the nodes of the available networks (friends, quotes, replies, retweets). DeepReading, IXA, and UNED, exploiting contextual information, proposed a feature that computes the mean distances of each user to the rest of users, whose stance is known. TextWiller experimented with multi-dimensional scaling (MDS) for retaining the first and second dimension for each of the four networks. QMUL-SDS exploited *Node2vec* and *deepwalk*, for learning a vectorial representation of the nodes of the networks. Also in Task B, no team made use of neither morphological nor syntactic features.

The comparison between the approaches respectively used for dealing with Task A and Task B, clearly highlights the benefits of exploiting information from different and heterogeneous sources. In fact, by a quick glance at Table 3.5 and Table 3.6 it is noticeable how the systems submitted in Task B overall obtain consistently higher scores with respect to systems submitted in Task A, where only textual information was usable.

No team resorted to the use of morphological nor syntactical features, let alone dependency-based syntactical features. It will be even more interesting, then, to compare the results obtained by the novel approach I have engineered as a main contribution to this thesis, that exploits dependency syntax, with those that participated in this task (see Section 3.3.2). Among all participating teams, the UNITOR team proposed an approach

that particularly captured my attention, as - in its own fashion - proposes a method that connects stance and irony (among other characteristics) speculating the first might be a good indicator for retrieving the latter [Giorgioni et al., 2020]. On this regard, I will discuss more in Chapter 4, where I try to summarize the many discussions opened in the present thesis, by finally connecting irony and stance as well.

Additional training data – The only team who participated in the unconstrained setting of *SardiStance* has been UNITOR. They proposed two unconstrained runs in addition to other two constrained ones. For the unconstrained setting, they downloaded and labeled about 3,200 tweets using distant supervision and used the additional data to train their systems. In particular they created the following subsets:
- 1,500 AGAINST: tweets from 2019 containing the hashtag: #gatticonsalvini[13];
- 1,000 FAVOUR: tweets from 2019 containing the hashtags: #nessunotocchilesardine, #iostoconlesardine, #unmaredisardine, #vivalesardine and #forzasardine[14];
- 700 NONE/NEUTRAL: texts derived from news titles. These were retrieved by querying to Google news with the keyword "sardine".

Other resources – Five teams declared to have used also other resources such as lexica, word embeddings, or others. In particular, GhostWriter used a grammar model to rephrase the tweets. MeSoVe exploited SenticNet [Cambria et al., 2014] and the "Nuovo vocabolario di base della lingua italiana".[15] QMUL-SDS took advantage of temporal embeddings and *fastText* embeddings, while only one team, UninaStudents, used a sentiment lexicon: AFINN [Nielsen, 2011b]. Lastly, Venses used a proprietary lexicon of Italian, enriched with conceptual and semantic information. Similarly TextWiller's approach relied on a self-created vocabulary[16] and trained word-embeddings on the Italian corpus of web texts PAISÀ [Lyding et al., 2014].

Seven teams participated in Task B, for *contextual* stance detection, submitting a total of 13 runs. Most teams extensively explored the additional features available for Task B. Notably, the only three runs with a score lower than the baseline (see Table 3.6) are those that have not benefited from any features based on the users' social network.

In particular, it is interesting to observe that all the teams that participated in both tasks, also produced better results in the second setting. Experimenting with different classifiers trained with the textual content of the tweets, as well as with features based on contextual information (additional info on the tweets, on users, or their social networks), seems therefore to allow to obtain overall better results.

Furthermore, among the 6 teams that participated in both tasks, only 4 fully explored the social network relations of the author of the tweet. The only two runs that overcome the baseline without investigating the structures of the social graphs are those submitted by the SSNCSE-NLP team. Only one team participated in both tasks exploiting the exact

[13]Translation: #kittenswithsalvini.

[14]Translation: #nobodytouchesthesardines, #iamwiththesardines, #aseaofsardines, #hoorraytosardines and #gosardines.

[15]https://www.internazionale.it/opinione/tullio-de-mauro/2016/12/23/il-nuovo-vocabolario-di-base-della-lingua-italiana.

[16]https://www.dropbox.com/s/4df128teotrit8y/word_emb_Evalita?dl=0.

same architecture (with no difference in hyperparameters). This, allowed to compare the F1-scores obtained in the first setting with those obtained in the second, highlighting that adding contextual features could increase performance of +0.2432, in terms of $F1_{avg}$.

Additionally, we calculated the increment in performance between the score obtained by the run ranked as 1st position in Task A (UNITOR, $F_{avg} = 0.6853$) and the score of the run ranked as 1st position in Task B (IXA, $F_{avg} = 0.7445$), showing that taking advantage of contextual features could increase performance up to 8.6% in terms of $F1_{avg}$.

The organization of the *SardiStance* shared task, and the inherent creation of a new stance-annotated dataset, allowed us to encourage the NLP scientific community to start working on the field of SD also in Italian. Furthermore, it has given me the opportunity of experimenting firsthand the evaluation of newly created systems, relying on state-of-the-art techniques. As I have mentioned before, no team used neither morphological nor syntactic information, and thus, it will be one of my novel contributions to this thesis, that of experimenting with them on the new dataset for SD for Italian (see Section 3.3.2). Additionally, as mentioned earlier one of the participating teams, proposed an approach for detecting stance that speculates that finding irony first, might be useful for the detection of stance in a following moment. As this is my opinion too, and one of the fields I aim at investigating in the present thesis, on this regard, I will discuss more in Chapter 4, where I finally connect irony and stance as well.

3.2 Machine learning approaches for stance detection

The other side of the coin, with respect to the organization of shared tasks and the creation of annotated resources is that of task participation and systems' implementation. In this section I will give an overview of the main techniques used for tackling the task, mainly analyzing the contribution of approaches based on NLP not only in Italian, but rather in a variety of languages.

To the best of my knowledge, Somasundaran and Wiebe [2009] were the first ones to focus on detecting the stance towards a target rather than the polarity of a sentence. They presented (in an unsupervised framework) a stance recognition method for debate-side classification from web blogs. After that, a series of approaches for SD has been proposed in shared tasks, as mentioned in the previous section. On this regard, *SemEval-2016 Task 6* has been the first competition entirely dedicated to SD. Its participants overall employed traditional feature-based machine learning, deep learning, and combined (ensemble) methods. The best performing system for subtask A has been a Recurrent Neural Network (RNN)-based system [Zarrella and Marsh, 2016] and obtained an F1-score of 0.6782 in subtask A, while the best system for subtask B has been a system based on Convolutional Neural Networks (CNN) achieving a macro F1-score of 0.5628 [Wei et al., 2016].

One year later, the organizers of the task, investigated the importance of exploiting the sentiment expressed in a given text in order to improve SD [Mohammad et al., 2017]. They proposed themselves a system, which includes n-grams, char-grams, sentiment features coming from different lexica such as EmoLex [Mohammad and Turney,

2013], Hu&Liu lexicon [Hu and Liu, 2004], and MPQA Subjectivity Lexicon [Wilson et al., 2005]. Besides, they also considered the presence/absence of the target of interest in the tweet, the frequency of part-of-speech tags, emoticons, hashtags, uppercase characters, elongated words, and punctuation marks. The combination of these features together with a support vector machine classifier allowed them to outperform the scores achieved by all the participating systems in *SemEval-2016 Task 6*, obtaining a macro F1-score across all targets of 0.7640.

In the last four years, after the end of the contest, the dataset released for the official competition has been considered as a benchmark for Stance Detection in English and therefore, was exploited to carry on research regarding SD in English tweets by several research groups. Among them, Augenstein et al. [2016] proposed a neural approach based on bidirectional conditional encoding, Dey et al. [2018] implemented a two-phase LSTM using attention, Wei et al. [2018] explored the performances of a biderectional Long Short-Term Memory neural network (biLSTM), and Zhou et al. [2019] used a condensed CNN with attention over self-attention. Another very recent work worth mentioning is that of Zhang et al. [2020], in which the authors combine external knowledge derived from semantic and emotion lexica as a bridge to enable transfer learning across different targets of interest. Then, the semantic-emotion graph representation is fully integrated into a BiLSTM.

Lai et al. [2017b] proposed an approach for detecting stance that relies on the knowledge of the domain and of the context surrounding a target of interest. The approach was evaluated selecting only two targets from the original dataset, i.e., Hillary Clinton and Donald Trump. Three groups of features were considered. *Structural* features include hashtags, mentions, punctuation marks, etc.. *Sentiment* features are extracted using a set of four lexica to cover different facets of affect ranging from prior polarity of words such as AFINN [Nielsen, 2011a] and Hu and Liu Lexicon, to fine-grained emotional information such as LIWC [Pennebaker et al., 2001] and the Dictionary of Affect in Language [Whissell, 2009]). Finally *Context-based* features attempt to capture the information surrounding a given target, looking around at the concepts of "friends" and "enemies" as the entities related to the target, defining a set of relationships between the target and the entities around it. Furthermore, the authors also exploited the additional annotation carried out in Mohammad et al. [2017] on the dataset of the shared task. The proposed approach outperforms the state-of-the-art results, showing that information about "enemies" and "friends" of politicians helps in detecting stance towards them. The importance of joining textual and contextual features in the task of SD, has indeed proven to be successful also in *SardiStance* (see Section 3.1.1).

Furthermore, in a recent work, together with my colleagues, we revised the automatic system proposed before in Lai et al. [2017b] and implemented a more refined version that takes into account the same sets of features proposed in combination with deep learning architectures such as LSTM, BiLSTM and a CNN. Our study encompasses a multilingual perspective, also considering Spanish, Catalan, French and Italian, besides English [Lai et al., 2020a].

In the *StanceCat* shared task, as cited in Section 3.1, well-known approaches for

classification, such as Support Vector Machines, and novel techniques, such as deep learning approaches, were applied by the ten different participating teams. For Catalan and Spanish, a system that I have submitted together with my colleagues, i.e., ITACOS [Lai et al., 2017a], resulted the best performing system, which consists in a supervised approach based on three groups of features: *Stylistic*, *Structural*, and *Context-based*. These results validate the relevance of contextual information in SD, as I will describe more carefully in Section 3.2.1.

Several scholars also investigated SD from a "Network Science" perspective, leveraging methods from the computational social science research field. For instance, Lai et al. [2017c] analyzed the role of social relations together with the users' stance towards the BREXIT referendum. Furthermore, taking into account that people may change their stance after some particular event, happening when the debate is still active, they also explore stance from a diachronic perspective. The authors collected a set of English tweets containing the hashtag *#brexit*, and provided an annotated corpus where diachronic triplets of tweets posted by 600 users active in the debate have been annotated for stance. The outcomes show two main results that may be of particular interest for addressing SD: that users sharing the same stance towards a particular issue tend to belong to the same social network community, and users' stance diachronically evolves [Lai et al., 2020b].

A similar experiment has been performed by Lai et al. [2018] analyzing the political debate on Twitter about the *Italian Constitutional referendum* held in 2016. The authors analyzed both the diachronical evolution of the stance and the online social relations of the users involved in the debate. Interestingly, the typology of the relations used for creating the network (retweets, replies, and quotes) highly affects the performance of the SD system [Lai et al., 2019]. The effects of online social network interactions on future attitudes have been thoroughly examined in Magdy et al. [2016], focusing on how a content generated by a user and network dynamics can be used to predict future attitudes and stances in the aftermath of a major event. The authors explored the effectiveness of three types of features for the prediction, namely content features (i.e., the body of the tweets from a user), profile features (i.e., user-declared information such as name, location, and description), and network features (i.e., user interactions with the Twitter community, through mentions, retweets, and replies).

Concerning SD in tweets, in Rajadesingan and Liu [2014], the authors implement a semi-supervised framework coupled with a supervised classifier to identify users with differing opinions. The authors exploit a retweet-based label propagation, based on the observation that if many users retweet a particular pair of tweets within a reasonably short period of time, then it is highly likely that the two tweets are similar in some aspect. In their work, they label tweets either as "for" or as "against" on the basis of the similarity with the values of the labels surrounding each tweet. Similarly, in the work of Raghavan et al. [2007], a label propagation algorithm is used for community detection. Their approach is particularly simple and efficient, in fact, in their iterative algorithm each node adopts the label that most of its neighbors currently have and it seems to work really well in unsupervised contexts.

An interesting work regarding the concept of "hompohily", i.e., the tendency of indi-

viduals to associate and bond with similar others, which could prove very useful for the task of SD, is that of DellaPosta et al. [2015]. Their work, although describes opinions and aggregating circles from a sociological perspective is very much connected with the world of SD. In fact, the authors carried out computational experiments on a case study taking into account the political and the ideological alignments. Their aim was to analyze how homophily influence the stereotyped perception of the world.

As mentioned in the previous section, within the shared task on Stance Detection in Italian tweets (*SardiStance at EVALITA 2020* [Cignarella et al., 2020b]), the participants resorted to a great variety of features that could be grouped in the following categories: stylistic, textual, psychological/social and propositional. The best-performing team of the shared task, Giorgioni et al. [2020] proposed an original approach in which they tackle SD as a more general task, constituted by three different subtasks: sentiment analysis, irony detection and hate speech detection. In their system they firstly assign a value to each instance of the datasets regarding these three different dimensions, and following they assign a proper label for stance (favour, against, none/neutral). The other approach that seems definitely beneficial is the employment of word-embeddings trained on corpora of a textual genre that is similar to the instances that need to be classified. For instance, Similarly TextWiller trained word-embeddings on the Italian corpus of web texts PAISÀ [Lyding et al., 2014] and obtained fairly good results.

The state of the art for SD in Italian tweets, as well as it happens in other fields and NLP tasks, is dominated by a variety of approaches that are inherited from BERT and RoBERTa transformer architectures. Within the *SardiStance* task, the best results over all were indeed obtained by those teams that submitted systems exploiting AlBERTo [Polignano et al., 2019], GilBERTo or UmBERTo.

Some recent advances in stance detection that are worth referring to and citing are, for example, the work of Darwish et al. [2020], who presented a highly effective unsupervised framework for detecting the stance of highly prolific Twitter users using dimensionality reduction to and clustering. Also Rashed et al. [2020] propose an unsupervised method for target-specific stance detection in a polarized setting (Turkish politics), achieving 90% precision in identifying user stances, while maintaining more than 80% recall. Such recent work has shown that performing stance detection on Twitter users who are very active and thus write many tweets on a single target, can be highly accurate. However, such methods perform poorly or fail completely for less active users, who may have written only a few tweets about a pre-defined target. Some researchers proposed an effective method based on expanding the tweets of a given user using their Twitter timeline, even though it might not be topically relevant [Samih and Darwish, 2021].

To the best of my knowledge, beside my own research, Sun et al. [2016] are the only authors that have published some work, in which syntactic features are explored for SD. In their approach, as participants in the NLPCC-ICCPOL 2016 Task 4, Xu et al. [2016] exploit both morphological features, connecting each word to its PoS tag and syntactic features, exploiting dependency trees, connecting two words that are in a dependency relation between themselves and also exploiting the "paths" that connect the root of a sentence to each "leaf" of the syntactic tree. Their research was conducted solely on

data in Chinese, extracted from Sina Weibo and Twitter. While in the present thesis I work with five different languages and targets, solely with Twitter data. Furthermore, I present a bigger variety of dependency-based features, that can capture different nuances of syntactical structures.

3.2.1 Participation in the *StanceCat 2017* shared task

In this section I describe the ITACOS[17] submission for the *Stance and Gender Detection in Tweets on Catalan Independence* shared task Taulé et al. [2017]. This approach is based on three diverse groups of features: stylistic, structural and context-based. Together with my colleagues we introduced two novel features that exploit significant characteristics conveyed by the presence of Twitter marks and URLs. The results of the experiments are promising and will lead to future tailoring of these two features in a finer grained manner. Since this task participation was early on at the beginning of my PhD (September 2017), I did not exploit any syntactic feature, as my interest towards *Universal Dependencies* was only at its beginning. This experience though, helped me mature significant understanding on the subject of SD and, as I will explain in a later section (3.3.2), I will test other approaches, more significant in terms of the contribution given by dependency-based syntax, on the datasets that were made available after this competition ended, thus carrying more relevant information for the content of the present thesis.

The competition was articulated into two subtasks about information contained in Twitter messages written both in Catalan and Spanish: the first subtask was related to detecting author's stance towards the independence of Catalonia, while the second one aimed at identifying the gender of the author of the tweet. For the sake of relevance, I will describe here only on the first subtask, leaving the interested reader refer to Lai et al. [2017a] for further details on the second subtask as well.

The starting point of our proposal is to be found in the method proposed in Lai et al. [2017b] in which the authors (some PhD students peers) exploited three diverse groups of features: *Structural* such as punctuation and other Twitter marks, *Sentiment* i.e., lexica covering different facets of affect, and finally *Context-based*, which considered the relationship that exists between a given target and other entities in its domain. Then, we defined a set of features distributed as follows:

a) Stylistic Features

• *Bag of Words (BoW)*: each tweet was pre-processed for converting it to lowercase. We used unigrams, bigrams and trigrams with a binary representation.

• *Bag of Part-of-Speech labels (BoP):* we used TreeTagger Schmid [1994, 1995] for extracting both the part-of-speech and lemmas.

• *Bag of Lemmas (BoL):* we used TreeTagger, as in the feature described above.

• *Bag of Char-grams (BoC):* we considered chargrams of 2 and 3 characters.

b) Structural Features

• *Bag of Twitter Marks (BoTM)*: we exploited a Bag of Words considering only the

[17] **I**nterlinguistic **T**rio for the **A**utomatic **C**lassification **O**f **S**tance.

words extracted from multi-word Twitter marks (hashtags and mentions) splitting them by capital letters.
• *Bag of Hashtags (BoH):* we considered the hashtags as terms for building a vector with binary representation.
• *Frequency of Hashtags (freqHash).*
• *Uppercase Words (UpW)*: this feature refers to the amount of words starting with a capital letter.
• *Punctuation Marks (PM)*: we took into account the frequency of dot, comma, semicolon, exclamation and question marks.
• *Length (Length)*: three different features were considered to build a vector: number of words, number of characters, and the average of the length of the words in each tweet.

c) Context Features
• *Language (Lan):* we created a vector exploiting the labels ES for Spanish and CA for Catalan provided by the organizer.
• *URL (Url):* we observed that tweets containing a URL are common in the training dataset. We decided to take advantage of this by considering different aspects extracted from short URLs. First, we identified whether or not the web address of reference was reachable. Second, we retrieved the words contained on the web address, then we built a bag-of-words using this information.

Experimental setting. The organizers provided a dataset of 8,638 tweets written in Spanish and Catalan labelled with stance (AGAINST, FAVOUR, and NEUTRAL). The distribution is skewed towards FAVOUR for Catalan and towards NEUTRAL for Spanish (respectively 30.66% and 29.38%) [Taulé et al., 2017]. Similar trends were found in Bosco et al. [2016b].

Therefore, it appears that language could be a useful feature for SD in the Catalan independence debate concerning a region characterized by a strong bilingualism and a smoldering nationalism. As Millar [2005] pointed out «Language divides and unites us. It [...] impinges upon our identity as individuals, as members of a particular ethnic or national group, and as citizens of a given polity».

In order to assess the performance of the participating systems, a test set of 2,162 unlabelled tweets was provided as for the evaluation metrics, the macro-average of the F1-score of the two main classes (FAVOUR and AGAINST) was used, as in the pioneer task for SD [Mohammad et al., 2016].

ITACOS experiments. In our experiments, we addressed SD as a classification task.[18] We carried out several experiments, evaluating them through 10-fold cross-validation, by combining both the features introduced above together with a set of classifiers composed by: Support Vector Machine (SVM), Random Forest (RF), Logistic Regression (LR), Decision Tree (DT), and Multinomial Naïve Bayes (MNB). Besides, we exploited a Majority Voting (MV) strategy considering the different predictions of the above mentioned classifiers as described in Liakata et al. [2012].

[18]The code is available on github for further exploration and for allow reproducibility of our experiments: https://github.com/mirkolai/iTACOS-at-IberEval2017.

Name	Features list
Set_α	BoW, BoL, BoC, Url, BoTM, freqHash, UpW
Set_β	BoW, BoL, BoP, BoC, Url, BoH, freqHash, Length
Set_γ	BoW, BoL, BoP, BoC, Url, freqHash, Lan, Length
Set_δ	BoW, BoL, BoP, BoC, Url, freqHash, PM, Length
Set_ϵ	BoW, BoL, BoP, BoC, Url, BoH, PM, Lan

Table 3.7: Best-ranked sets of features using the training set.

We analyzed the obtained results and selected the five combinations of features that showed the best performance for the stance detection task. The resulting sets of features are shown in Table 3.7. We participated in the shared task with five different runs for each language and each subtask. Table 3.8 shows the obtained results by using both the features and the classifier used in each of the submitted runs.

Run	**Stance Detection** Features and classifier	F1-score *Catalan*	F1-score *Spanish*
iTACOS.1	Set_α + SVM	**0.680**	0.544
iTACOS.2	Set_ϵ + LR	0.633	0.544
iTACOS.3	Set_β + LR	0.625	**0.548**
iTACOS.4	$5x5^*$	0.636	0.530
iTACOS.5	Set_α + MV	0.657	**0.548**

Table 3.8: Results for stance detection on the training set.

* The final prediction is the most frequent prediction over the 25 combinations between sets of features and machine learning algorithms.

Official results. The approach we proposed ranked first among 10 participating teams in the subtask of SD in both Catalan and Spanish. Table 3.9 shows the official results on the test set. At a first glance, it is possible to observe that our submissions performed better in Catalan, in fact our five runs ranked among the first 8 positions. In Spanish, on the other hand, our less performing run ranked as the 18^{th} position.

Catalan		
Ranking	Run	F1-score
1	**iTACOS.2**	**0.4901**
2	iTACOS.1	0.4885
4	iTACOS.3	0.4685
7	iTACOS.4	0.4490
8	iTACOS.5	0.4484

Spanish		
Ranking	Run	F1-score
1	**iTACOS.1**	**0.4888**
7	iTACOS.2	0.4593
12	iTACOS.3	0.4528
14	iTACOS.4	0.4427
18	iTACOS.5	0.4293

Table 3.9: Official results for stance detection

As shown in Table 3.9, the best result in each language was not achieved by the same run. iTACOS.2 (Set_ϵ + LR) better performs for Catalan, while iTACOS.1 (Set_α + SVM) better suits Spanish. The poorest results in both languages were obtained by

using ITACOS.4 ($5x5$ *approach*) and ITACOS.5 (Set_α + MV). As expected, the best performing runs (ITACOS.1 and ITACOS.2) contain both context-based features, validating the importance of considering contextual information in SD. For example, both runs include the feature *Url* and we are, therefore, interested in evaluating the impact of such feature. For this reason, we carried out experiments on the training set by applying a modified version of iTACOS.1 and ITACOS.2 where the *Url* feature has been removed. Looking at the results, we observed a drop in the performance of -0.029% for Catalan and of -0.002% for Spanish in ITACOS.1; and of -0,004% for Catalan and of -0.002% for Spanish in ITACOS.2 thus marking the importance of such feature.

For what concerns classifiers, LR and SVM achieved the best performance in both languages. Surprisingly, the approach exploiting MV is not performing well.

A linguistic revision. Focusing only on the cases where the results obtained with ITACOS.1 disagree with respect to the golden labels provided by the organizers, I report some examples both in Catalan and Spanish:

Ex.31 #elecciones #catalunya #NO #27S https://t.co/oBuTDnUEHj
→ #elecciones #catalunya #NO #27S https://t.co/oBuTDnUEHj

LANGUAGE: CATALAN

GOLDEN LABEL: AGAINST

ITACOS.1: FAVOUR

Ex.32 Ale @JuntsPelSi, a casa, son solo unas #eleccionescatalanas autonómicas. Mañana a trabajar que es lunes. Seguís teniendo el mismo DNI. #27S
→ @JuntsPelSi, go at home, there is only one autonomous #eleccionescatalanas. Tomorrow, go to work that it'll be Monday. You will have the same DNI (Spanish ID). #27S

LANGUAGE: SPANISH

GOLDEN LABEL: AGAINST

ITACOS.1: FAVOUR

Ex.33 En estas #eleccionescatalanas de decide una posible independencia y un gobierno que vele por los derechos de su pueblo, VOTA @catsiqueespot
→ In these #eleccionescatalanas we decide for a possible independence and a government that fights for the rights of its population, VOTE @catsiqueespot

LANGUAGE: SPANISH

GOLDEN LABEL: FAVOUR

ITACOS.1: AGAINST

Example 1, has been marked as FAVOUR from our classifier ITACOS.1, probably because of the biased token "catalunya", written in Catalan. However, the explicit semantic information carried by the hashtag #NO pointing to AGAINST was ignored, thus leading to a wrong classification. Considering Spanish, Example 2 has been appointed as FAVOUR instead of AGAINST. The presence of the mention @JuntsPelSi (Catalan independence coalition) could have misdirected our classifier. On the other hand, the tweet in example

3 was tagged as AGAINST whereas it should have been FAVOUR as we clearly infer from "VOTA @catsiqueespot" and according to the golden labels.

A similar qualitative analysis helped us to shed some light on the relevance of each single feature we exploited and in the subsequent selection of features to be included in our final sets. Nevertheless, it only provides further contribution to the hypothesis, often found in the literature I cited above about SD, that mostly semantics matters in addressing this task. In the present thesis, my principal goal is to go further, towards a less traveled terrain, and to investigate the contribution that may come from syntax, as I did in the previous chapter for what concerns irony.

In this section I described the ITACOS submission for the *Stance and Gender Detection in Tweets on Catalan Independence* task at IberEval-2017. The proposed approach, chiefly based on *context* and *structural* features, not investigating at all the importance of syntax, and it proved to be highly successful concerning the task of stance in both languages, as our system ranked as the first position among ten participating teams.

In Section 3.3.1 I will start investigating shallow syntactic features applied to SD, and in Section 3.3.2 I will take advantage of the availability of these two same datasets (Spanish and Catalan) among others, to experiment with syntax too. Not only I will experiment with classical ML, but also with neural networks. Thanks to the participation in this task, therefore, I have acquired sensitivity regarding the SD classification problem, I have been guaranteed access to both datasets, which, for their characteristics and dimensions, could be considered the gold standards for stance detection in Spanish and Catalan.

3.3 Stance detection using dependency syntax

As anticipated in Section 3.2, as I was able to verify, beside my own research there are very few works of NLP that have explored the contribution of syntax as feature for SD. The only authors that have published some work on this regard are Sun et al. [2016], who, as participants in the NLPCC-ICCPOL 2016 shared task on SD in Chinese, exploit both morphological and syntactic features, connecting words that are in a dependency relation between each other. Their research was conducted solely on data in Chinese, extracted from Sina Weibo and Twitter. While in the present thesis I work with five different languages in a multi-lingual setting, at least as many targets, and two different textual genres (Twitter data and Reddit posts). Furthermore, I will present a finer-grained variety of dependency-based features, that can capture different nuances of syntactical structures.

Therefore, in this section I describe my participation in the *RumorEval 2019* shared task, on stance and rumour detection, in which I have firstly explored syntactic features. Later on, in Subsection 3.3.2, I will present an unpublished study, born after mirroring the methodology and experiments conducted in Section 2.3.2 for irony, on the case study of SD.

3.3.1 Participation in the *RumorEval 2019* shared task

In this section I describe the UPV-28-UNITO system's submission to the RumorEval 2019 shared task. The approach we applied for addressing both the subtasks of the contest exploits both classical machine learning algorithms and word embeddings, and it is based on diverse groups of features: stylistic, lexical, emotional, sentiment, meta-structural and Twitter-based. Furthermore, a novel set of features, that takes advantage of the syntactic knowledge drawn from text, is introduced in the approach. This represents my first attempt of encoding syntactic information for addressing SD, and thus, it has to be mainly considered an exploration of the field.

The problem of detecting rumours in social media lately has been attracting considerable attention, considering that their diffusion is facilitated by large users' communities, where also expert journalists are unable to keep up with the huge volume of online generated information and to decide whether a news is a hoax [Procter et al., 2013, Webb et al., 2016, Zubiaga et al., 2018]. The main goal of the *RumorEval 2019* shared task is *Rumour Stance Classification*, i.e., the task that intends to classify the type of action expressed by different posts of a same thread [Qazvinian et al., 2011] according to a set of four given categories: Supporting (S), Denying (D), Querying (Q) or simply Commenting (C) on the rumour. For instance, referring to Twitter, once a tweet that introduces a rumour is detected (the "source tweet"), all the tweets having a reply relationship with it, (i.e., being part of the same thread), are collected to be classified. This is the task I have described at the beginning of this chapter as OPEN STANCE CLASSIFICATION, which is usually indeed indicated with the acronym SDQC, by referring to the four categories listed above and exploited for indicating the attitude of a message with respect to the rumour [Aker et al., 2017]. My participation in this task, together with another PhD student, is mainly focused on the investigation of linguistic features of social media language that can be used as cues for detecting rumours. In particular, I tried to introduce, here, for the first time, syntactic features encoded in a preliminary, exploratory way.[19] Nevertheless, while the RumorEval 2019 shared task involved two tasks, Task A (rumour stance classification) and Task B (verification), for the sake of coherence with the topic of the present thesis I will describe only the former, and for what concerns a description of the second I direct the reader to the system report [Ghanem et al., 2019a].

Provided that opinions around a claim can act as proxies for its veracity, and not only of its controversiality, it is reasonable to consider the application of SDQC techniques for accomplishing rumour analysis tasks. A first shared task, concerning SDQC applied to rumour detection, has been organized at SemEval-2017, i.e. *RumorEval 2017* [Derczynski et al., 2017]. Furthermore, several research works have analyzed the open issue of the impact of rumours in social media [Resnick et al., 2014, Zubiaga et al., 2015, 2018], for instance exploiting linguistic features [Ghanem et al., 2018]. Such approaches may be also found in works which deal with the problems of Fake News Detection [Ciampaglia et al., 2015, Hanselowski et al., 2018], that, for several facets, goes hand in hand with rumour SD.

[19] Source code is available on GitHub: `https://github.com/bilalghanem/UPV-28-UNITO`.

The corpus used for the *RumourEval 2019*'s open stance classification Task A contains a total of 8,529 English posts, namely 6,702 from Twitter and 1,827 from Reddit. The portion of data from Twitter has been built by combining the RumorEval 2017 training and development datasets [Derczynski et al., 2017], and includes 5,568 tweets: 325 source tweets (grouped into eight overall topics such as: Charlie Hebdo attack, Ottawa shooting, Germanwings crash...), and 5,243 discussion tweets collected in their threads. The dataset from Reddit, which has been instead newly released this year, is composed by 1,134 posts: 40 source posts and 1,094 collected in their threads.

	Training	Test
Twitter	5,568	1,066
Reddit	1,134	761
Total	6,702	1,827

Table 3.10: Training and test data distribution.

All data have been split in training and test set with a proportion of approximately $80\% - 20\%$ (see Table 3.10).

UPV-28-UNITO Submission. The approach and the features selection we applied is based on a set of manual features described in the paragraphs below. We built moreover another set of features (i.e., second-level features) extracted by using the manual features together with features based on word embeddings (see paragraph "Second-level Features" for a detailed description). For modeling the features distribution with respect to each thread, we used for task B the same features as in task A. Then, in both tasks, we fed the features to a classical machine learning classifier.

• **Manual Features**
For enhancing the selection of features, we investigated the impact of diverse groups of them: emotional, sentiment, lexical, stylistic, meta-structural and Twitter-based. Furthermore, we introduced a novel set of syntax-based features.

Emotional features - We exploited several emotional resources in order to build features for our system. Three lexica: (a) *EmoSenticNet*, a lexicon that assigns six WordNet Affect emotion labels to SenticNet concepts [Poria et al., 2013]; (b) the *NRC Emotion Lexicon*, a list of English words and their associations with the eight Plutchik's basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive) [Mohammad and Turney, 2010]; and (c) *SentiSense*, an easily scalable concept-based affective lexicon for Sentiment Analysis [De Albornoz et al., 2012]. We also exploited two tools: (d) *Empath*, a tool that can generate and validate new lexical categories on demand from a small set of seed terms [Fast et al., 2016]; and (e) *LIWC* a text analysis dictionary that counts words in psychologically meaningful categories [Pennebaker et al., 2001].

Sentiment features - Our sentiment features were modeled exploiting sentiment resources such as: (a) *SentiStrength*, a sentiment strength detection program which uses a lexical approach that exploits a list of sentiment-related terms [Thelwall et al., 2010]; (b) *AFINN*,

a list of English words rated for valence with an integer between minus five (negative) and plus five (positive) [Nielsen, 2011b]; (c) *SentiWordNet*, a lexical resource in which each WordNet synset is associated to three numerical scores, describing how objective, positive, and negative the terms contained in the synset are [Esuli and Sebastiani, 2007]; (d) *EffectWordNet*, a lexicon about how opinions are expressed towards events, which have positive or negative effects on entities (+/-effect events) [Choi and Wiebe, 2014]; (e) *SenticNet*, a publicly available resource for opinion mining built exploiting Semantic Web techniques [Cambria et al., 2014]; and (f) the *Hu&Liu* opinion lexicon.[20]

Lexical features - Various lexical features already explored in similar Sentiment Analysis tasks were employed: (a) the presence of *Bad Sexual Words*, a list extracted from the work of Frenda et al. [2018]; (b) the presence of *Cue Words* related to the following categories: *belief, denial, doubt, fake, knowledge, negation, question, report* [Bahuleyan and Vechtomova, 2017]; the categories *an, asm, asf, qas, cds* of the multilingual hate lexicon with words to hurt *HurtLex* [Bassignana et al., 2018]; (d) the presence of *Linguistic Words* related to the categories of *assertives, bias, fatives, implicatives, hedges, linguistic words, report verbs*; (e) the presence of specific categories present in *LIWC*: *sexual, certain, cause, swear, negate, ipron, they, she, he, you, we, I* [Pennebaker et al., 2001].

Stylistic features - We employed canonical stylistic features, already thoroughly explored in Sentiment Analysis tasks and already proven useful in multiple domains: (a) the count of *question marks*; (b) the count of *exclamation marks*; (c) *length* of a sentence; (d) the *uppercase ratio*; (e) the count of consecutive *characters* and *letters*[21] (f) and the presence of *URLs*.

In addition to the above-listed, common features exploited in Sentiment Analysis tasks, in this work we introduce two novel sets of features: (1) *Problem-specific features* (considering the fact that the dataset is composed by Twitter data and Reddit data) and (2) *Syntactic features* considering the more general aim of my PhD studies, i.e., measuring the impact of syntax in different case studies.

Meta-structural features - Since training and test data are both from Twitter and Reddit, we explored meta-structural features suitable for data coming from both platforms: (a) the *count of favourites/likes*, in which we have two different value distribution (Twitter vs. Reddit), so we normalized them in a range 0-100; (b) the *creation time* of a post, encoded in seconds; (c) the *count of replies*; and (d) the *level*, i.e., the degree of "nestedness" of the post in the thread.

Twitter-only features - Because of the duplicitous nature of the RumorEval 2019 dataset (Twitter and Reddit), some of the several features based on Twitter metadata could not be used in this task.[22] As follows: (a) the presence of *hashtags*; (b) the presence of *mentions*; (c) the count of *retweets*. And also some user-based features: (d) whether

[20] http://www.cs.uic.edu/liub/FBS.

[21] We considered 2 or more consecutive characters, and 3 or more consecutive letters.

[22] For the instances from Reddit, that did not have a representation of one of the following features, the empty values has been filled with a weighted average of the values obtained by other similar instances.

the user is *verified* or not; (f) the count of *followers*; (g) the count of *listed* (i.e., the number of public lists of which this user is a member of); (h) the count of *statuses*; (i) the count of *friends* (i.e., the number of users that one account is following); (l) the count of *favourites*.

Syntactic features - In our system some feature has been also modeled by referring to syntactic information involved in texts. After having parsed[23] the dataset in the *Universal Dependencies* format, thus obtaining a set of syntactic "dependency relations" (*deprel*), we were able to exploit: (a) the *ratio of negation* dependencies compared to all the other relations; (b) the Bag of Relations (BoR_all) considering all the *deprels* attached to *all the tokens*; (c) the Bag of Relations (BoR_list) considering all the *deprels* attached to the tokens belonging to a selected *list of words* (from the lists already made explicit in the paragraph "Lexical Features" in Section 3.3.1); and finally (d) Bag of Relations (BoR_verbs) considering all the *deprels* attached to all the *verbs*, thus fully exploiting morphosyntactic knowledge. As it can be seen here, the features based on dependency syntax are few and specifically tailored for this task. Nevertheless, I imagined that it would be possible to measure their contribution in the following research steps and that a better understanding of them would lead the way for future investigation.

• **Second-level Features**

For the second-level features, we employed (a) the cosine similarity of one instance with regard to its parents and (b) information of the tree structure of a thread, exploiting its "nesting" and depth from the source tweet.

Similarity with parents - In this feature, we used the cosine similarity to measure the similarity between each post with its parents. The parents of a reply are: the direct upper-level post (A) and the source post in the thread (B) (see Figure 3.3).

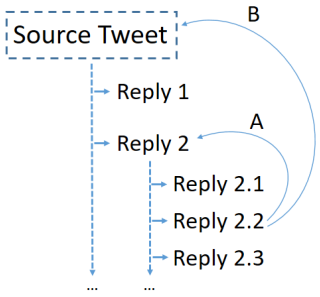


Figure 3.3: An example for reply 2.2 parents.

We extracted the cosine similarity in A and B by using the manual features' final vector and words embeddings average vectors of the posts; the words embeddings average vector for a post is extracted by averaging the embeddings of the post's words[24].

[23]The parsing system we applied is UDPipe, available at: https://pypi.org/project/ufal.udpipe/

[24]We used the pre-trained Google News word embeddings in our system: https://code.google.com/archive/p/word2vec/

SDQC depth-based clusters - We built level-based stance clusters from the posts. For each stance class (SDQC), we extracted all the belonging posts that correspond to one of the four classes and we computed the average value of the feature vectors (as one unique cluster). Since we have four main stances, this process ended with four main clusters. For feature extraction, we measured the cosine similarity for each post with regard to these four clusters. As done in the previous feature described above, we built these clusters by using both the manual features' vectors and word embeddings' vectors of the posts, so each stance cluster is represented in two ways. In these four main clusters, we did not consider the nesting of the posts in the thread.

Also, we obtained the same clusters but instead of averaging all the posts that correspond to a stance, we considered the nesting of the posts in the thread. We split the nesting of the threads into five groups: posts with depth one, two, three, four, five or larger. For each of these levels, we extracted four SDQC clusters (depth-based). For instance, if a post occurs in depth two, we measured the cosine similarity between this post and 1) the four main SDQC clusters,[25] 2) the four depth-based SDQC clusters two.

Experiments. We tested different machine learning classifiers in each task performing 10-fold cross-validation. The results showed that the Logistic Regression (LR) produces the highest scores. For tuning the classifier, we used the Grid Search method. The parameters of the LR are: $C = 61.5$, $penalty = L2$, and since the dataset is not balanced, we used different weights for the classes as COMMENT = 0.10, DENY = 0.35, SUPPORT = 0.20 and QUERY = 0.35. We conducted an ablation test on the features employed in Task A in order to investigate their importance in the classification process. Table 3.11 presents the ablation test results as well as the system performance using 10-fold cross-validation. Set B, for instance, is constituted by all features presented up to now, minus emotional features. Set C is composed by all features minus sentiment features and so on.

Provided that the organizers allowed a maximum of two submissions for the final evaluation, we used all the features (set A) in the first submission and set M for the second submission. The final score, measured in terms of macro F1-score, for our best run is 0.4895.

By comparing the results obtained in the run that exploits the set with all features with the run obtained by using *set E* (all features minus syntactic ones), we observe a drop of 0.2 in terms of macro F1-score ((A)54.9 - (E)54.7). On the other hand, set K, L and M are to be interpreted as sets containing all features *minus* the features that are signed with letters between brackets. So, for instance, set K would be a set containing all features minus (C) sentiment, (E) syntactic and (I) cosine similarity with parents. As it can be seen all three sets K, L and M do not contain syntactic features (marked by the letter identifier *E*), and all three of them present a higher macro F1-score, with respect to the results obtained by the runs that exploit all features (including syntactic).

These two contrasting outcomes could be interpreted such as, that even though with a minor contribution, syntactic features might help in the detection of stance, but only when paired with certain other groups of features. Fore sure in the next sections I will

[25] Four features using the manual features, and another four using the words embeddings.

SET	FEATURE	M-F1
A	All features	54.9
B	A - Emotional features	54.5
C	A - Sentiment features	54.7
D	A - Lexical features	53.6
E	A - Syntactic features	54.7
F	A - Stylistic features	50.1
G	A - Meta-structural features	54.5
H	A - Twitter-only features	54.9
I	A - Cosine similarity with parents	55.3
I.1	I using only manual features	54.9
I.2	I using only words embeddings	54.9
J	A - SDQC depth-based clusters	47.7
J.1	J using only manual features	53.3
J.2	J using only words embeddings	51.1
K	A-(C+E+I)	55.6
L	A-(B+C+E+G)	**55.7**
M	A-(B+C+E+G+I.2)	**55.9**

Table 3.11: Set of features for the ablation test.

try to investigate more on this matter and on the contribution that syntax alone could give to SD.

Error Analysis. A manual error analysis allowed us to see which categories and posts turned out to be the harder to be dealt with. We found out that SUPPORT was misclassified 114 times, DENY 92 times, QUERY 44 times, and COMMENT 57 times. Therefore, SUPPORT seems to be the hardest category to be correctly classified.

		PREDICTED			
		S	D	Q	C
GOLD	S	–	0	13	101
	D	1	–	6	85
	Q	5	1	–	38
	C	5	17	35	–

Table 3.12: Confusion matrix of errors.

This can be seen also in Table 3.12, that reports the confusion matrix of predicted vs. gold labels, and shows also that no errors are present between the two most opposite classes (i.e., SUPPORT vs. DENY). Moreover, by better investigating the gold test set, it should be moreover observed that several semantically empty messages of the test set have been marked using some class, while our system marks them as COMMENT, i.e., selecting the most frequent class when a clear indication of the content is lacking.

To conclude, in this section I presented an overview of my participation in *SemEval 2019 Task 7 - Determining Rumour Veracity and Support for Rumours*, focusing only on the Task A, which consists in an open stance classification exercise applied to English messages from Twitter and Reddit, for which I have submitted two runs. The approach

was based on emotional, sentiment, lexical, stylistic, meta-structural, *depth-based*, and Twitter-based features. Furthermore, I took advantage of the participation in the shared task to introduce and explore *syntactic* features. Being not exactly clear from the experiments performed in this context the specific contribution that they provide in SD, I have, therefore, explored them in an extensive way as I will detail in the next section.

3.3.2 Multilingual stance detection with neural models

Finally, in this section I present the most novel part of my work, which has never been presented in any conference nor published in any journal.

In order to highlight the importance of syntactic information for the task of SD across different datasets and languages, and paired with state-of-the-art techniques, here I mirror the multilingual experiments previously done for irony detection (see Chapter 2, Section 2.3.2), applying the same approach to the problem of SD. Therefore, as it has been done in the previous chapter on irony detection, I present the results obtained with regard to experiments performed on a variety of different corpora, in five different languages (English, Spanish, French, Italian and the addition of Catalan, with respect of what done for irony detection), with a variety of complex architectures that primarily exploit morphosyntactic knowledge, represented throughout the format of Universal Dependencies. Functioning as a summary regarding this chapter on stance detection, I aim at addressing the following research questions. Firstly, I would like to investigate whether *features derived from morphology and syntax could help to address the task of stance detection* [RQ-3] (see 1.3), and following *to what extent using resources such as treebanks for training NLP models improves the performance in stance detection* [RQ-4] (see 1.3).

Mirroring the precedent work done for irony detection in Cignarella et al. [2020a] (reported in Chapter 2), I propose here to address SD as a multi-class classification task, testing an automatic system on the same four languages: English, Spanish, French and Italian. In this framework I also tested the method on a fifth language, since a dataset annotated accordingly to stance was available, having the same target of interest of the Spanish dataset: Catalan. Furthermore, with respect to Italian, I tested the approach on two different datasets with two different targets of interest, namely: the *Constitutional Referendum* [Lai et al., 2020a] and the *Sardines Movement* [Cignarella et al., 2020b], as the latter was just recently made available within the SardiStance shared task at EVALITA 2020, as described in Section 3.1.1 above.

In the multilingual experimental setting I took advantage of some datasets that have been made available during the last few years within evaluation campaigns (*SemEval 2016 - Task 6* [Mohammad et al., 2016], *StanceCat at IberEval 2017* [Taulé et al., 2017] and *SardiStance at EVALITA 2020* [Cignarella et al., 2020b]), and some that have been created *ad hoc* in the research group where I work, for previous studies on SD (Emmanuel Macron and Constitutional Referendum [Lai et al., 2020a]) and are available online for the scientific community.[26] In order to operate in a supervised setting, for what concerns

[26] https://github.com/mirkolai/MultilingualStanceDetection/tree/master/dataset.

the English dataset made available within SemEval 2016, I only selected the portion regarding *Hillary Clinton*, since data about *Donald Trump* were only present in the test set. In this way I carried out experiments in a supervised framework across all languages and targets. The same applies to the French dataset, for which I only considered the tweets having *Emmanuel Macron* as target of interest, excluding the tweets related to *Marine Le Pen*.

language	target	train				test			
		AGAINST	FAVOUR	NONE	TOTAL	AGAINST	FAVOUR	NONE	TOTAL
English	Hillary Clinton	393	118	178	689	172	45	78	295
Spanish	Independencia	335	1,446	2,538	4,319	84	361	636	1,081
Catalan		131	2,648	1,540	4,319	32	663	386	1,081
French	Emmanuel Macron	244	71	109	424	64	20	22	106
Italian	Constitutional Referendum	389	129	148	666	97	34	36	167
	Sardines Movement	1,028	589	515	2,132	742	196	172	1,110

Table 3.13: Benchmark datasets used for target-specific SD.

In Table 3.13 for each dataset, I report the language, the target of interest, the name of the shared task in which it was released and its paper reference, the number of tweets for each class (*against*, *favour*, *none*) and the total number of instances, for both training set and test set. The aim of my task is, thus, to determine the stance expressed by the user with respect to a given target.

Once again, since the morphological and syntactic knowledge is crucial for performing the experiments described in the rest of this section, I needed to obtain a representation of all the datasets in UD format. Considering that all the datasets used in this part of my work consist of Twitter data, whenever possible, I used resources where this genre, or at least user-generated content of some kind was included as training data for parsing. More precisely, the model for English has been trained on the EWT treebank [Silveira et al., 2014], that for Spanish on both GSD-Spanish corpus [McDonald et al., 2013] and the ANCORA corpus [Taulé et al., 2008]. Also the model for Catalan was trained on the ANCORA corpus, while that for French on the GSD-French corpus [McDonald et al., 2013]. Finally, the model for Italian was trained on the POSTWITA-UD corpus [Sanguinetti et al., 2018], on the ISTD treebank [Simi et al., 2014] and on the TWITTIRÒ-UD corpus [Cignarella et al., 2019b].

A higher precision in this phase of the work can be a bottleneck for what concerns the accuracy of the experiments that I will describe in the following sections. In fact, the approach is entirely dependency syntax based and the results strictly depend upon the quality of parsed data. Nevertheless, I considered the output provided by the automatic parser for all the languages to obtain rather satisfactory results for the purposes of my study.

Methodology. The main goal of the experiments I present in this and in the following paragraphs consists in evaluating the contribution of syntax-based linguistic information provided in the datasets described above to the task of SD. As I previously did in Chapter 2 (see Section 2.3), also here I performed a set of experiments where several models were

implemented exploiting classical machine learning algorithms, deep learning architectures and state-of-the-art language models implemented with the Python libraries *scikit-learn* and *keras*. The methodology I propose here is perfectly mirrored from the experiments described in the chapter regarding irony detection, in which a multilingual setting is proposed and neural models are evaluated together with dependency-based features, recalling the idea that syntax could be useful in a variety of language scenarios and in a variety of sentiment analysis tasks.

For the sake of clarity and conciseness, I will list here below the features employed and the models tested, without entering in deep explanations. For more details, the reader may refer to Section 2.3.2, where an exhaustive description of both is provided.

Pre-processing and features
From all the tweets, I stripped the URLs and normalized all characters to lowercase. I investigated the use of the following features aimed at exploiting information conveyed by syntax, studying in particular the impact of the availability of training resources in UD format:

- ngrams, chargrams;
- deprelneg, deprel;
- relationformVERB, relationformNOUN, relationformADJ;
- Sidorovbigramsform, Sidorovbigramsupostag, Sidorovbigramsdeprel.

Models
Having as primary goal the exploration of the features described in the previous paragraph (and in Section 2.3.2) and as a second case study testing their effectiveness also in stance detection, I implemented a variety of models, including the following:

- SVM, LR, RF, MLP;
- M-BERT, M-BERT+syntax, M-BERT+best_feats.

In the setting proposed previously for irony detection, I had also tested different sets of pre-trained word-embeddings to initialize the neural models, namely *FastText* and a dependency-based word2vec proposed by Levy and Goldberg [2014] (*word2vecf*). The latter were trained on the concatenation of all the treebanks available in the UD repository for each considered language. Since the experiments conducted on this end resulted not to be conclusive, and furthermore a very specific type of data was needed (i.e., datasets where not only an annotation for stance was present, but also a morphological and syntactic analysis was available) I decided not to reproduce that setting of those experiments, but to proceed directly in describing what I believe are the most interesting experiments and results.

Experiments and Results. Therefore, in the framework of SD, I propose two different experimental settings: the first one aims at exploring the dependency-based features

listed above paired with classical machine learning algorithms (and thoroughly described in Chapter 2, Section 2.3.2) in order to perform a feature selection and discover the best combination. In the second setting, I experiment with the Multilingual Bidirectional Encoder Representations from Transformers (M-BERT) and different additions of the features explored in the first setting.

a) Selection of best features
Mirroring the previous experimental setting described in the chapter regarding irony, in order to understand which of the features are the most relevant for the SD task, I carried out an evaluation of all the possible combinations of the features, combined with four different models (SVM, LR, RF and MLP) and I evaluated them with respect to the averaged macro F1-score.[27]

features	English Clinton	French Macron	Spanish Independencia	Catalan Independencia	Italian Referendum	Italian Sardines
model	MLP	MLP	MLP	MLP	SVM	MLP
macro F1-score	.673	.596	.493	.497	.967	.651
ngrams			✓		✓	✓
chargrams	✓	✓	✓	✓		✓
deprel	✓		✓	✓		✓
deprelneg	✓	✓		✓		
relationformVERB	✓	✓	✓			
relationformNOUN		✓				
relationformADJ			✓			
Sidorovbigramsform		✓		✓		
Sidorovbigramsdeprel	✓			✓		✓
Sidorovbigramsupostag		✓	✓		✓	✓

Table 3.14: Features exploited in the best runs with classical ML algorithms in each language scenario.

From the observation of Table 3.14 a vastly heterogeneous scenario emerges. Seemingly there is no pattern among language scenarios, regarding the same features exploited for SD. On the contrary, the multilayer perceptron is proven to be the best performing classical ML algorithm across all languages, apart from the setting regarding the Constitutional Referendum in Italian. This has an explanation, that was already found out in precedent work Lai et al. [2020a] and surely a special clarification regarding the nature of the dataset is due.

The same, seemingly 'strange' behaviour applies also for the exploitation of features. In the Italian Constitutional Referendum scenario, in fact, only two features are employed (ngrams and Sidorovbigramsupostag). Following the same deduction as in Lai et al. [2020a], the Italian dataset on the Constitutional Referendum seems to be particularly *sui generis* when compared with the other five. Indeed, within the dataset the exploitation of hashtags is wide and coherent in the whole corpus. For instance the

[27]The average value obtained between the F1-score of the AGAINST class and the F1-score of the FAVOUR class as it was done in [Mohammad et al., 2016].

hashtags #iovotosì (#Ivoteyes) and #iovotono (#Ivoteno) have been exploited almost in each tweet that we took into consideration, and we believe that just their presence (as boolean value) already is a clear manifestation of stance. For this reason the two features ngrams and Sidorovupostags are already sufficient to reach an extremely high F1-score (0.967). And the same applies to the algorithm, as Support Vector Machines are sufficiently good to perform textual classification where such a textual feature is so blatant in indicating stance.

From Table 3.14 it emerges how in all the configurations used for achieving the best score at least one dependency-based syntactic feature was exploited and in particular those based on Sidorov's work, i.e., the last three rows of the table. This provides evidence for **partially** answering to our third research question [RQ-3] (*Could features derived from morphology and syntax help to address the task of stance detection?*), since those are the features where the real structure from root to branches of syntactic trees is encoded.

c) Syntactically-informed BERT for stance detection

Lastly, I performed experiments with the state-of-the-art BERT language model. For each language, I ran the straightforward M-BERT model as anticipated. In a second phase of this setting, I implemented the base architecture by adding the dependency-based syntactic features detailed in previous sections in two different ways in order to have a clear-cut evidence on the actual contribution derived from dependency syntax to SD.

language	**target**	**best run (report and score)**		***SVC***	**M-BERT**		
				+unigrams	—	**+syntax**	**+best_feats**
English	H. Clinton	[Zarrella and Marsh, 2016]	.671	.570	.650	.562 (↓ .088)	.636 (↓ .014)
French	E. Macron	[Lai et al., 2020a]	.687	.526	.511	.511 (= .000)	.533 (↑ .022)
Spanish	Independencia	[Lai et al., 2017a]	.489	.420	.467	.443 (↓ .024)	.463 (↓ .004)
Catalan		[Lai et al., 2017a]	.490	.468	.478	.462 (↓ .016)	.476 (↓ .002)
Italian	Referendum	[Lai et al., 2020a]	.971	.951	.959	.960 (↑ .001)	.960 (↑ .001)
	Sardines	[Giorgioni et al., 2020]	.685	.578	.586	.599 (↑ .013)	.563 (↓ .023)

Table 3.15: Results obtained combining M-BERT and dependency-based syntactic features. Green values and arrows pointing up show an increment in performance with respect to results obtained by the bare architecture. Red values and arrows pointing down indicate a performance reduction, with respect to results obtained by the bare architecture. Orange values show no change.

In Table 3.15 I report the results of the best system exploiting these datasets (with the reference to the working notes). Furthermore, as a baseline reference measure, I also added the results obtained with a SVC and a bag of words of unigrams as only feature, as it is a baseline proposed in most competitions. Each of the experiments with M-BERT has been performed 5 times with the hyper-parameters previously described in Section 2.3.2 in order to take into account the differences of random initialization, and the average macro F1 score of such number of runs is reported.

Firstly, it is interesting to see how, the M-BERT base architecture never surpasses

the results obtained with more complex architectures, proposed by the participants of shared tasks, confirming the complexity of the this task.

Moreover, by having a look at the colorful right-hand side of the table, it can be seen how the addition of syntactic knowledge (M-BERT+syntax) determined a widely varied spectrum of outcomes. By the predominance of the colors orange and red (indicating stasis or loss in terms of performance), it is obvious to state that morphosyntactic information, taken alone and encoded into the M-BERT architecture does not provide strong nor consistent beneficial contribution to the task of SD. Not only the results obtained by the models *M-BERT+syntax* and *M-BERT+best_feats* obtain results lower than the state of the art approaches, but in most cases, they result in being also lower than the results obtained with the base architecture (M-BERT).

3.4 Concluding remarks on stance detection

In this chapter, I have focused on the second case study that I took as object of analysis in the present thesis: Stance Detection. Similarly to what I have done in the previous chapter on irony detection (see Chapter 2), also here in the first place I surveyed the related work on the topic, dedicating peculiar attention to shared tasks organized in evaluation campaigns during last decade. After that, I focused on the availability of corpora annotated accordingly to stance.

Secondly, I focused on the description of the *SardiStance 2020* shared task, which I organized together with some colleagues from the Università degli Studi di Torino and from the Universitat Politècnica de València within the last year of my PhD studies. I carefully described the task in order to have a detailed zoom on the issues that may arise while dealing with the creation of a dataset, the application of an annotation scheme and the organization of a shared task. I outlined the strengths and difficulties of such activity (see Section 3.1.1). My direct involvement in the organization of a shared task and in the development of a novel resource allowed a meaningful improvement of my awareness of the importance of the formalization of the stance detection problem, paving the way for a deeper understanding of the main approaches applied in this field.

In Section 3.2, I focused on the vast panorama of approaches and machine learning techniques that have been used to address this problem, providing a description of the state-of-the-art models, which also allow us to tackle the problem of stance detection in a multilingual perspective. I later provided a detailed description of my participation in the *StanceCat* shared task at *IberEval 2017*, in which I had the first opportunity to get in touch with a formal modeling of stance detection as a multi-class classification problem (see Section 3.2.1). The participation in the shared task, occurred in the very first months of my PhD, and together with other peer PhD students, I had the first experience of creating an automatic model for resolving a sentiment analysis task; an experience that shed some light for following more elaborate studies on the matter.

In the third section of this chapter regarding the topic of stance detection, I focused on the approaches I proposed that exploit some knowledge made available in the Universal Dependencies format.

In particular, I firstly described my participation in the *RumorEval* shared tasks at *SemEval 2019* in which a preliminary study of features based on dependency syntax has been proposed. Therefore, I provided the implementation details and performances of the systems that I submitted as a participant, especially stressing the contribution given by morphosyntactic information (see Section 3.3.1).

Finally, in Section 3.3.2, I mirrored the multilingual experiments previously done for irony detection (see Chapter 2, in particular Section 2.3.2), and applied the same framework to the problem of stance detection. Therefore, precisely as it has been done in the previous chapter on irony detection, I presented the results obtained with regards to experiments performed in four different languages: English, Spanish, French and Italian. In this chapter I experimented also with a fifth language, i.e., Catalan, due to the availability of a benchmark dataset also in this language (extracted from the dataset of the same shared task that also made the Spanish dataset available). Furthermore, I studied the phenomenon of stance with respect to six different targets – one per language, and two different for Italian (*Constitutional Referendum* and *Sardines Movement*) – with a variety of complex architectures that primarily exploit morphosyntactic knowledge represented throughout the format of Universal Dependencies.

In particular, throughout the whole chapter I have laid the useful background in order to fully investigate the task of stance detection and for answering to the following research questions:

- **RQ-3** *Could features derived from morphology and syntax help to address the task of stance detection?*
- **RQ-4** *To what extent does using resources such as treebanks for training NLP models improve the performance in stance detection?*

The outcomes obtained in the investigation proposed in this chapter, surely are a bit disappointing, but they do not come as a total surprise. As previously mentioned in the introduction, when I was formulating the problem statement and the research questions regarding the second case study of SD, I had anticipated that there were no linguistic theories nor research work pointing towards the fact that morphosyntax might prove useful in this task. Furthermore, recalling the example cited in the first chapter, where we observed two simple sentences having opposite stance, but identical syntactic structure:

Ex.34 I love the Sardines Movement.

Ex.35 I hate the Sardines Movement.

we had already anticipated that taking morphology and syntax as only features to detect stance might indeed be calling a long shot.

With the experience matured in the last years and with the findings and all the above-mentioned conditions I have highlighted in the present chapter, we can state that

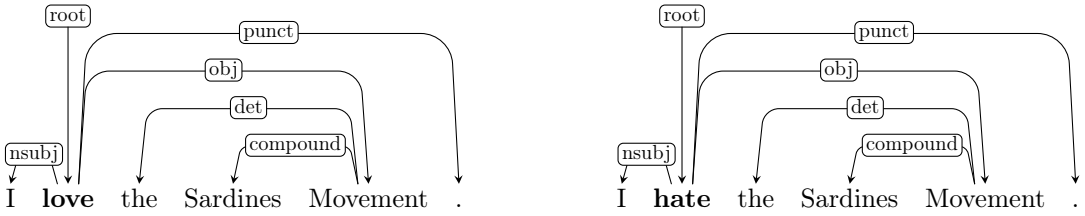


Figure 3.4: Dependency trees of the sentences in Ex.34 (left) and in Ex.35 (right).

– even if we are not obtaining the new state-of-the-art results – the results obtained lead in the direction of further investigation, pointing mainly towards a better understanding of features' behaviour when stacked in a pre-trained language model such as BERT. Furthermore, following the main intuition that led to the creation of my PhD path and the creation of the present thesis, and also following in the footsteps of what anticipated by one of the participating teams in *SardiStance*, who speculated that irony, among other characteristics, might be a useful feature in detecting stance: investigating the relationship between irony and stance might indeed be the turning point to hit in future work. In fact, irony is frequently used when expressing a polarized opinion, in each culture and language, the above-listed findings stress the importance of investigating its role in the manifestation of a certain stance, as I will describe in the next chapter.

Chapter 4

The interaction of irony and stance

After having explored the field of irony detection (Chapter 2) and of stance detection (Chapter 3), especially with the focus of exploring the contribution given by morphology and syntax in those fields, finally, in this chapter I will provide an analysis that might function as *trait d'union* between the two fields.

The lesson learned form the two previous chapters suggests that morphosyntactic cues might prove useful in the automatic detection of irony and that they combine well as features in classical machine learning algorithms, as well as in neural architectures. The same can not be said for the second case study, that of stance detection. In fact, as explained before, the expression of one's stance is frequently a shift that seems to depend more often on semantics rather than on syntactic patterns or constructions. Remember the cited example of "*I love the Sardines Movement*" versus "*I hate the Sardines Movement*".

Pairing with this consideration, within the organization of the *SardiStance* shared task at EVALITA 2020 (described in Chapter 3, Section 3.1.1) my attention was caught by one of the participating system proposed, whose method exploited the information regarding irony as an "auxiliary task" in order to later on detect the author's stance from a tweet, obtaining very good results [Giorgioni et al., 2020]. Subsequently, I was then led to think that the most efficient way of dealing with the two fields of irony detection on one side and stance detection on the other side, was that of treating irony detection as a sort of preprocessing step before detecting stance. We know from literature, in fact, that irony might work as a *polarity reverser* [Basile et al., 2014] when it comes to detect the sentiment of a tweet, so my speculation is that it might play the same role also with regard to stance. As anticipated at the beginning of Chapter 3, when I described the framework of stance detection, sentiment analysis and stance detection are here considered as two different tasks, as it has been stressed in many works how a positive sentiment extracted from textual content, not always goes hand in had with a favourable stance and neither the opposite situation occurs every time (negative sentiment and contrasting stance) [Buytaert, 2018]. This difference between sentiment polarity and stance can be observed, for instance, in the following tweet taken from the dataset of *SemEval-2016 Task 6* [Mohammad et al., 2017] and cited in previous work [Lai et al., 2020a].

Target of interest: Climate change is a real concern
@RegimeChangeBC @ndnstyl It's sad to be the last generation that could change but does nothing. #Auspol
Sentiment: NEGATIVE
Stance: FAVOUR

Taking all of this into consideration, I decided then to analyze more closely the relationship between irony and stance, and for doing so the most useful resource would be that of a dataset that has encoded both phenomena in it. Therefore, in the following sections I will describe the annotation process that led to the enhancement of the *SardiStance* dataset, which was enriched with irony annotations (alongside the labels of stance), and the linguistic analysis performed on it.

4.1 Annotating irony on the *SardiStance* dataset

As described in the previous chapter, I took part in the organization of the first stance detection task for Italian, *SardiStance 2020* within EVALITA (see Section 3.1.1). On the one hand, organizing this event gave me the possibility of looking at the task of SD with a closer look and become more aware about the related issues. On the other hand, it allowed an in-depth investigation of irony detection too, which is the other task analyzed in this thesis (see Chapter 2).

In order to provide some insights on the relationship that occurs between the two tasks, I decided to enrich the dataset created for the *SardiStance* by adding a binary annotation regarding the phenomenon of irony, and to carry on a qualitative analysis. In Table 4.1 is displayed the number of tweets belonging to the *SardiStance* dataset, accordingly to their division into training and test set, and accordingly to their label of stance.

TRAINING SET			TEST SET			TOTAL		
AGAINST	FAVOUR	NONE	AGAINST	FAVOUR	NONE	AGAINST	FAVOUR	NONE
1,028	589	515	742	196	172	1,770	785	687
2,132			**1,110**			**3,242**		

Table 4.1: Distribution of tweets accordingly to stance labels within the portions of the *SardiStance 2020* dataset.

Twelve different annotators labelled the data released for the shared task for the phenomenon of stance (annotating 800 tweets each, see detailed procedure in Section 3.1.1), and, in a first round of annotation, tweets were also annotated for the presence of irony. Of course, after the first phase, a certain degree of disagreement among annotators was present and so a second round of annotations was needed, which has not been applied for releasing also this additional information to participants. If on the one hand the second phase of annotation, i.e., the resolution of disagreement, for stance was conducted several months ago, before the release of the training set,[1] the resolution of disagreement

[1] May 2020.

regarding the dimension of irony was accomplished only in a more recent time, for the purpose of present thesis, by three other skilled annotators.

After the resolution of doubtful cases, whose final label was discussed among annotators, I was finally able to obtain a definitive annotation and, therefore, a gold standard. In Table 4.2 the distribution of irony labels is reported, accordingly to the division into training set and test set.

TRAINING SET		TEST SET		TOTAL	
IRONIC	NOT IRONIC	IRONIC	NOT IRONIC	IRONIC	NOT IRONIC
917	1,215	493	617	1,410	1,832
2,132		**1,110**		**3,242**	

Table 4.2: Distribution of tweets accordingly to irony labels.

Interestingly, even though the original splitting between training test and test set, was made in such way for the purposes of serving the shared task requirements, the labels regarding the absence/presence of irony resulted in having a similar distribution: ironic tweets in the training set represent the 43% in the training set (917 tweets) and the 44% in the test set (493 tweets).

In Table 4.3 and in Figure 4.1 the distribution of *ironic* and *not ironic* tweets can be seen also in comparison with the three different labels of stance.

AGAINST		FAVOUR		NONE	
IRONIC	NOT IRONIC	IRONIC	NOT IRONIC	IRONIC	NOT IRONIC
926	844	197	588	287	400
1,770		**785**		**687**	

Table 4.3: Distribution of ironic tweets accordingly to stance labels.

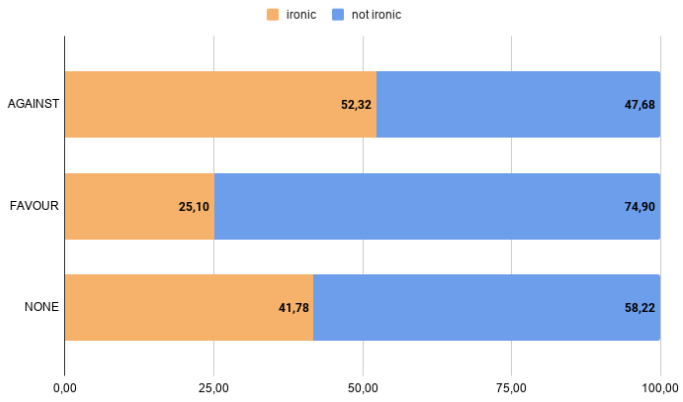


Figure 4.1: Distribution of irony accordingly to stance labels.

As irony, sometimes declined into sarcasm, can be used to ridicule and diminish a target, it is not surprising to see that ironic instances are 52% (926 tweets) in the class that

presents an AGAINST stance towards the Sardines Movement. On the other hand, among the tweets that FAVOUR, irony is present only 25% of the times (197 tweets), whereas the other 75% of favourable tweets is not ironic. For what concerns the NONE class, irony is present in 41% of the cases (287 tweets), still marking a significant presence, but not as significant as in the AGAINST class.

From these simple observations it is already more than clear that irony is a phenomenon that is vastly present in social media messages, and furthermore widely used to express one's stance towards a determined target. These findings highlight once again, the importance of studying and treating irony as a preprocessing step before tackling the task of irony detection.

4.2 Analyzing morphology and syntax in ironic tweets

In this step of the work, I carried out a quantitative analysis aimed at identifying specific occurrences of PoS tags and UD relations, whose different distribution might hinder or help automatic irony detection.

In order to obtain a representation of the *SardiStance* dataset in *Universal Dependencies* format, I applied the full pipeline of tokenization, lemmatization, PoS-tagging and dependency parsing provided by *UDPipe* [Straka and Straková, 2017]. For this purpose, I trained a new UDPipe model on three different gold benchmarks, namely PoSTWITA-UD (6,712 tweets) [Sanguinetti et al., 2018], UD-Italian (14,167 sentences) [Simi et al., 2014] and TWITTIRÒ-UD (1,424 tweets) [Cignarella et al., 2019b].

PoS tags	**IRONIC**	**frequency %**	**NOT IRONIC**	**frequency %**
ADJ	1,886	4.62	3,269	5.09
ADP	5,239	12.83	8,150	12.69
ADV	2,221	5.44	3,780	5.89
AUX	1,541	3.77	2,774	4.32
CCONJ	1,429	3.50	2,308	3.59
DET	5,353	13.11	8,495	13.23
INTJ	197	0.48	185	0.29
NOUN	6,482	15.87	10,211	15.90
NUM	323	0.79	532	0,83
PRON	2,372	5.81	3,945	6.14
PROPN	2,065	5.06	2,844	4.43
PUNCT	4,594	11.25	6,614	10.30
SCONJ	644	1.58	1,048	1.63
SYM	1,905	4.66	2,929	4.56
VERB	4,384	10.74	6,939	10.81
X	203	0.50	191	0.30
total	**40,838**		**64,214**	

Table 4.4: Distribution of PoS tags in the *SardiStance* corpus divided according to irony.

I did not perform any manual correction of the dataset obtained automatically through the UDPipe pipeline, but for the sole purpose of a first exploration of PoS tags and

dependency relations, the quality of the data obtained automatically can be considered sufficiently good.

Table 4.4 shows the distribution of PoS tags in the corpus, differentiating between ironic tweets and not ironic tweets, while Table 4.5 shows their distribution but accordingly stance labels (against, favour, none).

PoS tags	AGAINST	frequency %	FAVOUR	frequency %	NONE	frequency %
ADJ	2,864	5.04	1,250	4.94	1,041	4.54
ADP	7,278	12.82	3,207	12.66	2,904	12.66
ADV	3,258	5.74	1,429	5.64	1,314	5.73
AUX	2,239	3.94	1,096	4.33	980	4.27
CCONJ	2,052	3.61	872	3.44	813	3.54
DET	7,506	13.22	3,296	13.02	3,046	13.28
INTJ	216	0.38	101	0.40	65	0.28
NOUN	9,252	16.29	3,905	15.42	3,536	15.41
NUM	422	0.74	242	0.96	191	0.83
PRON	3,460	6.09	1,455	5.75	1,402	6.11
PROPN	2,610	4.60	1,183	4.67	1,116	4.86
PUNCT	5,968	10.51	2,708	10.69	2,532	11.04
SCONJ	899	1.58	393	1.55	400	1.74
SYM	2,364	4.16	1,448	5.72	1,022	4.45
VERB	6,175	10.87	2,649	10.46	2,499	10.89
X	225	0.40	88	0.35	81	0.35
total	**56,788**		**25,322**		**22,942**	

Table 4.5: Distribution of PoS tags in the *SardiStance* corpus divided according to stance.

As it can be seen from the values in both tables and the relative frequency (expressed in percentage) it seems that there is whatsoever no kind of imbalance of PoS tags among categories, neither for what concerns *ironic* vs. *not ironic* nor among the three stance's classes *against*, *favour* and *none*. The PoS tags that were extracted from the tweets, with respect to the different labels have the same distribution over classes.

On the other hand, the distributions of the dependency relations as per the UD format can be seen in the two following tables. In Table 4.6 the distribution of deprels is reported differentiating between *ironic* tweets and *not ironic* tweets, while in Table 4.7 their distribution is provided according to stance labels (*against*, *favour* and *none*).

The observation of the dependency relations' distribution in the two different tables, does not show particular variance nor imbalance with statistical meaning. That is, the dependency relations are equally distributed in tweets that are classified ironic and not ironic. And the dependency relations contained in the texts of all the three different classes of stance are also very similar and present no imbalance.

These results may suggest that we can not use the morphosyntactic knowledge for modeling the phenomena we observed in the previous chapters, i.e., irony and stance. Nevertheless, in Chapter 2, we have seen that morphosyntactic features provide a contribution in the detection of irony, giving an answer to the first of my research questions [RQ-1] (*Could features derived from morphology and syntax help to address the task of*

dependency relations	IRONIC	frequency %	NOT IRONIC	frequency %
acl	274	0.67	480	0.75
acl:relcl	485	1.19	915	1.42
advcl	550	1.35	922	1.44
advmod	2,143	5.25	3,645	5.68
amod	1,457	3.57	2,504	3.90
appos	61	0.15	83	0.13
aux	795	1.95	1,401	2.18
aux:pass	88	0.22	134	0.21
case	4,664	11.42	7,218	11.24
cc	1,417	3.47	2,294	3.57
ccomp	364	0.89	611	0.95
compound	86	0.21	122	0.19
conj	1,781	4.36	2,966	4.62
cop	658	1.61	1,238	1.93
csubj	60	0.15	81	0.13
dep	311	0.76	357	0.56
det	5,128	12.56	8,035	12.51
det:poss	173	0.42	352	0.55
det:predet	58	0.14	104	0.16
discourse	200	0.49	197	0.31
discourse:emo	67	0.16	105	0.16
dislocated	5	0.01	8	0.01
expl	506	1.24	762	1.19
expl:impers	62	0.15	92	0.14
expl:pass	12	0.03	7	0.01
fixed	99	0.24	162	0.25
flat	25	0.06	57	0.09
flat:foreign	41	0.10	32	0.05
flat:name	340	0.83	504	0.78
iobj	289	0.71	413	0.64
list	0	0.00	2	0.00
mark	1,179	2.89	1,941	3.02
nmod	2,248	5.50	3,518	5.48
nsubj	2,065	5.06	3,338	5.20
nsubj:pass	75	0.18	96	0.15
nummod	258	0.63	433	0.67
obj	1,995	4.89	3,305	5.15
obl	2,354	5.76	3,467	5.40
obl:agent	58	0.14	133	0.21
parataxis	1,244	3.05	1,957	3.05
parataxis:hashtag	308	0.75	422	0.66
punct	4,592	11.24	6,613	10.30
root	1,410	3.45	1,832	2.85
vocative	52	0.13	42	0.07
vocative:mention	333	0.82	584	0.91
xcomp	468	1.15	730	1.14
total	**40,838**		**64,214**	

Table 4.6: Distribution of dependency relations divided according to irony.

dependency relations	AGAINST	frequency %	FAVOUR	frequency %	NONE	frequency %
acl	435	0.77	158	0.62	161	0.70
acl:relcl	764	1.35	353	1.39	283	1.23
advcl	804	1.42	340	1.34	328	1.43
advmod	3,137	5.52	1,377	5.44	1,274	5.55
amod	2,209	3.89	936	3.70	816	3.56
appos	82	0.14	36	0.14	26	0.11
aux	1,088	1.92	573	2.26	535	2.33
aux:pass	134	0.24	37	0.15	51	0.22
case	6,442	11.34	2,868	11.33	2,572	11.21
cc	2,037	3.59	867	3.42	807	3.52
ccomp	528	0.93	217	0.86	230	1.00
compound	129	0.23	37	0.15	42	0.18
conj	2,609	4.59	1,152	4.55	986	4.30
cop	1,016	1.79	486	1.92	394	1.72
csubj	84	0.15	29	0.11	28	0.12
dep	329	0.58	198	0.78	141	0.61
det	7,130	12.56	3,116	12.31	2,917	12.71
det:poss	290	0.51	143	0.56	92	0.40
det:predet	86	0.15	40	0.16	36	0.16
discourse	224	0.39	97	0.38	76	0.33
discourse:emo	83	0.15	57	0.23	32	0.14
dislocated	9	0.02	3	0.01	1	0.00
expl	696	1.23	292	1.15	280	1.22
expl:impers	75	0.13	39	0.15	40	0.17
expl:pass	13	0.02	3	0.01	3	0.01
fixed	141	0.25	63	0.25	57	0.25
flat	37	0.07	30	0.12	15	0.07
flat:foreign	35	0.06	17	0.07	21	0.09
flat:name	440	0.77	212	0.84	192	0.84
iobj	390	0.69	157	0.62	155	0.68
list	0	0.00	0	0.00	2	0.01
mark	1,693	2.98	720	2.84	707	3.08
nmod	3,090	5.44	1,407	5.56	1,269	5.53
nsubj	2,932	5.16	1,283	5.07	1,188	5.18
nsubj:pass	104	0.18	27	0.11	40	0.17
nummod	336	0.59	203	0.80	152	0.66
obj	2,885	5.08	1,244	4.91	1,171	5.10
obl	3,223	5.68	1,357	5.36	1,241	5.41
obl:agent	116	0.20	35	0.14	40	0.17
parataxis	1,683	2.96	784	3.10	734	3.20
parataxis:hashtag	343	0.60	244	0.96	143	0.62
punct	5,966	10.51	2,708	10.69	2,531	11.03
root	1,770	3.12	785	3.10	687	2.99
vocative	51	0.09	29	0.11	14	0.06
vocative:mention	474	0.83	270	1.07	173	0.75
xcomp	646	1.14	293	1.16	259	1.13
total	**56,788**		**25,322**		**22,942**	

Table 4.7: Distribution of dependency relations divided according to stance.

irony detection?). According to the recent trend of explainability in AI[2], I think that,

[2]See e.g., XAI at https://en.wikipedia.org/wiki/Explainable_artificial_intelligence and

these results give some hint about the fact that, for actually explaining how the linguistic and pragmatic phenomena are acting, we can not stop our investigation on quantitative aspects, such as numerical counts referring to the presence/absence of a certain PoS tag or dependency relation. Following this perspective, we should integrate in future approaches the ability to better find patterns and structures that could capture any meaningful feature also at a qualitative level.

4.3 A tentative error analysis comparing irony and stance

At this point of the thesis it would be ideal to perform some final experiments where irony is detected by integrating morphosyntactic information, and then stance is detected by incorporating the irony prediction. Unfortunately, in order to write the present dissertation, I needed to focus on describing at my best the work done on irony (Chapter 2) as well as the work done on stance (Chapter 3) and leave further experiments in this regard for future work and future publications. Additionally, one of the biggest obstacles to carry out such kind of experiments is the lack of available annotated resources. In fact, in order to perform such experiments, datasets that encode annotations for both phenomena would be fundamental (irony *and* stance at the same time). Indeed, developing linguistic resources such as annotated datasets for NLP purposes is a complex and time-consuming activity that needs to be performed by several skilled annotators. For the moment being, and to the best of my knowledge, the only dataset to comply these requirements is the *SardiStance* dataset (as described in Section 4.1).

Therefore, exploiting the dataset that I annotated together with my colleagues, I was able to carry out an error analysis in which I take into account the experiments done on stance detection in Chapter 3 and I correlate the predictions obtained with Multilingual BERT on SD in Italian with the presence/absence of irony in the gold test set of the *Sardistance* dataset. Although it is not a complete experimental setting, it allows me to compare the performances on stance detection and its correlation with irony on at least one dataset among the ones exploited in the previous chapters of this thesis.

In Table 4.8 a numerical evidence obtained by comparing the output prediction for stance of M-BERT and the gold test set of *SardiStance* is presented. After comparing the gold test set with the predicted label concerning SD (columns 1-3), it is also interesting to observe the distribution and the percentage of ironic and not ironic tweets (columns 4-7) for each predicted outcome.

The main idea spread in literature is that irony is a *polarity reverser* [Bosco et al., 2013] and that its presence in a text could hinder the correct comprehension of a certain utterance for humans as well as hindering the performances of NLP systems. Following this concept, tweets that contain irony could be more difficult to be understood from machines as well. Therefore, we would expect a lower accuracy if many of them are present, or we would expect a wider presence of ironic tweets in tweets that have been misclassified, assuming that the presence of irony is one of the main issues with text classification.

the variety of recent workshops on this topic among which *IJCAI-PRICAI 2020 Workshop on Explainable Artificial Intelligence* (XAI) https://sites.google.com/view/xai2020/home.

n#	gold stance		predicted stance	n#	IRONIC	freq. %	NOT IRONIC	freq. %
742	AGAINST	→	AGAINST	491	269	54.79	222	45.21
		→	FAVOUR	177	78	44.07	99	55.93
		→	NONE	74	31	41.89	43	58.11
196	FAVOUR	→	AGAINST	54	15	27.78	39	72.22
		→	FAVOUR	118	22	**18.64**	96	**81.36**
		→	NONE	24	9	37.50	15	62.50
172	NONE	→	AGAINST	59	26	44.07	33	55.93
		→	FAVOUR	57	21	36.84	36	63.16
		→	NONE	56	22	39.29	34	60.71

Table 4.8: Frequency of ironic and not ironic tweets accordingly to the different stances predicted by M-BERT. Comparison of corrected predictions and wrong predictions.

For example, in Table 4.8 we would expect a higher percentage of ironic tweets in the lines that highlight misclassifications (e.g., FAVOUR → AGAINST), and a lower presence of irony in classifications that are conducted correctly (e.g., AGAINST → AGAINST). The presentation of such correlations is shown in Table 4.8. However, the numbers reported there seem to show a neutral distribution. By observing the frequency of ironic and not ironic tweets in each setting of correctly classified (and wrongly classified) stances, there seems to be no outstanding relevant skewness of data.

The only unbalanced percentages that can be observed in the table are those of the distribution of ironic vs. not ironic tweets in the following setting: "FAVOUR → FAVOUR" (in bold). In that particular setting the M-BERT prediction shows an accuracy of 60.2% (118 tweets out of 196 are correctly labeled), and inside that percentage of correctly classified instances there are 81.36% of not ironic tweets. One bold assumption, by reading this information, could be that the prediction of the class FAVOUR, in this particular dataset, has been detected with a moderate accuracy because there is a lower presence of ironic texts. Although nothing similar could be deducted on the other prediction settings, and no other percentage distribution seem to have any kind of relevance.

It is important to stress that the entire test set taken into account here, consists of only 1,110 tweets and thus, it is very hazardous to make assumptions on such a small dataset. Furthermore, on such a small scale the elements that might concur in performing a good (or bad) classification of tweets are uncountable. The presence/absence of irony could not definitely be the only factor that is taken into account. However, it has been challenging to investigate more closely whether there might be a significant connection between irony and stance. This first step, conducted as a manual error analysis, could just be considered a preliminary step for investigating the relationship between irony and stance on a computational level, which is surely going to be one of the next steps in my research.

Chapter 5

Conclusions and future work

The present thesis is part of the broad panorama of studies of Computational Linguistics. In particular, it is collocated inside the sub-area of Natural Language Processing. As we have seen in the previous chapters, my main goal has been that of studying, in depth, the contribution of syntax in the field of sentiment analysis and, therefore, to study its behaviour in texts extracted from social media or, more generally, online contents.

Furthermore, given the recent interest of the scientific community in the Universal Dependencies project, which proposes a morphosyntactic annotation format aimed at creating a "universal" representation of the phenomena of morphology and syntax in a variety of languages, in this work I made a wide use of this format, aiming at proposing a study in a multilingual perspective, considering the languages I know (Italian, English, French and Spanish). From another perspective, my investigation also collocates within the growing trend of studies devoted to make AI's results more explainable, going beyond the achievement of scores in performing tasks and making their motivations readable and comprehensible for experts in such domain.

For all the above-mentioned reasons, in Chapter 1 I introduce the reader to all the background aspects that must be taken into account in order to understand the various aspects discussed in the present thesis. Later, on the one hand, I focused on the problem of irony detection, that I took into account as first case study (Chapter 2) and on the other hand I focused on the problem of stance detection (Chapter 3). As I briefly mentioned earlier, my central scope has been that of investigating the impact of morphosyntactic information in such tasks.

I wrote both Chapter 2 and Chapter 3 by trying to maintain a symmetric structure between them, where possible. In both chapters, firstly, I surveyed the related work on the main topic discussed, devoting particular attention to the description of evaluation campaigns and shared tasks that have been organized in the last few years, also stressing the importance of freely available annotated datasets and benchmarks (Sections 2.1 and 3.1). Later on, I described, for both irony and stance a broad panorama of approaches and machine learning techniques that are typically used in those fields, naming the innovative approaches exploited by participants of the above mentioned shared tasks and outlining the state-of-the-art models (Sections 2.2 and 3.2).

After having stressed the significant impact of evaluation campaigns on the devel-

opment of resources and the creation of evaluation benchmarks in different languages, I presented a summarized overview of the IronITA 2018 shared task (*Irony Detection in Italian tweets*) for irony detection (Section2.1.1), and a summarized overview of the *SardiStance 2020* shared task, for stance detection (Section 3.1.1). My direct involvement in the organization of both shared tasks allowed a meaningful improvement of my awareness of the importance of the formalization of these problems, paving the way for a deeper understanding of the main approaches applied in these fields.

Thirdly, only in the chapter on irony, I focused on the description of the TWITTIRÒ corpus, that I developed within my PhD studies, and that in part has also been distributed for training participating systems in the above-mentioned shared task on *Irony Detection in Italian tweets* (Section 2.1.2). This section has been dedicated in order to highlight the issues that may arise while dealing with a complicated pragmatic phenomenon such as that of irony, especially within the delicate process of corpus creation which entails both annotation and negotiation among human annotators. Starting from a nucleus of some hundreds of tweets collected during the development of my master thesis, I have later on enhanced the annotations of this corpus within the duration of my PhD up to now, i.e., the TWITTIRÒ-UD treebank, which is my contribution to the Universal Dependencies project.

Following, in more technical sections, I described my participation in several shared tasks. For what concerns irony detection, I have participated in the IroSVA 2019 irony detection shared task (*Irony Detection in Spanish Variants*) and, therefore, I presented the first introductory work that moves in the direction of a syntax-based approach for detecting irony (Section 2.3.1). On the other hand, for what regards stance detection I provided a more detailed description of my participation in the *StanceCat* shared task at *IberEval 2017*, in which I had the first opportunity to get in touch with a formal modeling of stance detection as a multi-class classification problem (Section 3.2.1). For stance detection, I had the opportunity of participating in a second shared task, i.e. *RumorEval* shared task at *SemEval 2019* in which I proposed a preliminary study of features based on dependency syntax (see Section 3.3.1).

Finally, in Section 2.3.2 for irony detection and Section 3.3.2 for stance detection, I described the most complete researches completed during my PhD, in which all the research efforts are finally combined together. I provided more accurate descriptions of feature engineering and system development in a multilingual scenario, together with a wider panorama of results and discussion points regarding primarily the impact of syntax on the tasks of irony and stance detection both. Specifically, in Section 3.3.2, I mirrored for stance detection the multilingual experiments previously done for irony detection (see Chapter 2, Section 2.3.2), and applied the same framework to the problem of stance detection. Therefore, precisely as it has been done in the chapter on irony detection, I presented the results obtained with regards to experiments performed in four different languages: English, Spanish, French and Italian. In the chapter on stance (Chapter 3) I experimented also with a fifth language, i.e., Catalan, due to the availability of a benchmark dataset also in this language. Furthermore, I studied the phenomenon of stance with respect to six different targets – one per language, and two different for

Italian (*Constitutional Referendum* and *Sardines Movement*) – with a variety of complex architectures that primarily exploit morphosyntactic knowledge represented throughout the format of Universal Dependencies.

After having explored the field of irony detection and of stance detection, especially with the focus of exploring the contribution given by morphology and syntax in those fields, finally, in Chapter 4 I provided an analysis that functions as a bridge between the two fields. The lesson learned form the two previous chapters, in fact, suggested that morphosyntactic cues might have proven useful in the automatic detection of irony and that they combine well as features in classical machine learning algorithms, as well as in neural architectures. The same could not be said for the second case study, that of stance detection. In fact, as explained before, the expression of one's stance is frequently a shift that seems to depend more often on semantics rather than on syntactic patterns or constructions.

In the present thesis, which is the product of my three-year long PhD studies, I have tried to establish a balance between the development of resources and computational experiments. Doing this, of course, has entailed several limitations, as the amount of time dedicated to the creation of resources is very big. As these datasets have been made available for the research community under the form of shared tasks, hopefully they will also have a large impact on a broader scientific progress in the respective research domains. Furthermore, the computational experiments have been performed for multiple languages, which makes it more interesting with regards to the portability of the obtained results versus other languages.

5.1 Conclusions

Finally, the results described in this thesis have attempted to answer the research questions I introduced in Section 1.3:

RQ-1 *Could features derived from morphology and syntax help to address the task of irony detection?*

Only in part, and with low (to none) statistical significance. In the work described in Section 2.3.2, I created linguistic resources syntactically annotated according to the well-known dependency-based scheme of *Universal Dependencies*. I took advantage of datasets used in shared tasks for irony detection and I enriched them with morphosyntactic annotation. The versatility of the UD format allowed me to apply a dependency-based approach for the detection of irony independently of a target language. Additionally, I experimented with UD-based word embeddings. From the experiments and the error analysis, described in Section 2.3.2, I could highlight how adding features based on morphosyntactic information could lead to a better performance in the task of irony detection in different settings and in different combinations of groups.

RQ-2 *To what extent does using resources such as treebanks for training NLP models improve the performance in irony detection?*

Referring to the datasets encoded in UD format, I proposed three distinct experimental settings. Firstly, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored, with the aim of finding the most informative set of features for detecting the presence of irony. In the second scenario two well-known word-embedding models are tested against gold standard datasets. Finally, in the third setting, dependency-based syntactic features are combined into the Multilingual BERT architecture. Furthermore, I experimented with datasets made available from previous shared tasks on irony detection in four languages: French (DEFT 2017), English (SemEval-2018 Task 3), Spanish (IroSvA) and Italian (IronITA). From these experiments I was able to prove that for those languages in which the resources in UD format were created *ad hoc*, results were also higher (see Section 2.3.2) However, results showed low (almost to none) statistical significance, meaning that different combinations of feature groups could be explored.

RQ-3 *Could features derived from morphology and syntax help to address the task of stance detection?*

Not completely. In Section 3.3.2, I described how I created linguistic resources syntactically annotated according to the well-known dependency-based scheme of *Universal Dependencies*. I exploited six different datasets used in previous shared tasks (or made available by independent researchers) for stance detection and I enriched them with morphosyntactic annotation. Once again, the versatility of the UD format allowed me to apply a dependency-based approach for the detection of stance independently of a target language. In Section 3.3.2 I mirrored the experiments done for irony and I have seen how the results obtained in this second case study are not completely satisfactory. As expected, the results point to the fact that the one's stance could be more often depending on semantic clues rather than on syntactic ones.

RQ-4 *To what extent does using resources such as treebanks for training NLP models improve the performance in stance detection?*

Partially. In Section 3.3.2, I proposed two distinct experimental settings. Firstly, a variety of syntactic dependency-based features combined with classical machine learning classifiers are explored, with the aim of finding the most informative set of features for detecting the presence of stance. In the second scenario those dependency-based syntactic features are combined into the M-BERT architecture. Once again, I experimented with datasets made available by previous shared tasks

or by independent researchers on stance detection in five different languages: English (*Hillary Clinton*, SemEval-2016 Task 6), Spanish and Catalan (*Catalan Independence*, StanceCat @ IberEval 2017), French (*Macron*), and Italian (*Constitutional Referendum*, and *Sardines Movement*). By means of these experiments, I was able to see how it seems that morphosyntactic cues are not directly helping in the detection of stance. And the same assumption is valid across languages.

It has been duly noted that syntax does not seem to be particularly informative nor helpful regarding directly the task of stance detection (second case study, in particular see Section 3.3.2). On the other hand, also supported by some previous linguistic studies, syntax seems to play an important role in the detection of irony.

A new speculation that comes to mind, is that it could be more useful to perform a *"cascade task"*. Meaning that, firstly it might be useful to predict irony, with the help of morphosyntactic cues (step 1), and only then (as step 2), proceeding in the detection of stance. In general, my assumption, is that predicting irony could be the first step in numerous other tasks, even shallow sentiment analysis, or the identification of fake news.

This outcome is something that should not be ignored, but obviously carrying out supervised studies in this sense would also mean dedicating a great effort and consuming much time in the creation of annotated datasets (that ought to be annotated on various layers, for different dimensions and phenomena). In fact, to further study this line of investigation, in Chapter 4 I proposed a shallow analysis of the Italian dataset regarding the Sardines Movement, which is only a small and limited beginning, but it is also certainly opening a new research perspective.

My work has certainly many limitations. Firstly, I needed to deal with the scarcity of data annotated in some adequate way and with the reduced size of the few datasets that are indeed available or those I helped to develop. Furthermore, this kind of investigation, mostly based on morphosyntactic cues that are applied to SA-related tasks, is a rather new one. In fact, there are very few studies going towards this direction up to these days. After having described my work in its entirety, it is once more fundamental to stress that this investigation collocates within the growing trend of studies devoted to make Artificial Intelligence results more explainable, going beyond the achievement of scores in performing tasks and making their motivations understandable for experts in the domain.

By having looked at some of the not completely positive results obtained in the wide variety of experiments performed in this thesis, it is fundamental to stress that we did not solely want to appreciate the outcomes in terms of numerical performances, but rather being more focused in the more profound linguistic reasons behind them. And the same is valid also for why sometimes results are poorer and why features do not make improvements on a certain task.

I am positive that if we manage to understand what is the linguistic knowledge a certain approach, or a group of features, leverages when it produces good (or poor) results, among many possible approaches, it could allow us to make more mature choices for following work. Indeed, the future of NLP research needs to go towards approaches that better integrate different types of knowledge (such as syntactic knowledge, for once)

and that manage to be more versatile for certain types of data and in different application contexts.

5.2 Future work

For what concerns the line of investigation on irony, in the future I would like to propose a wider experimental setting than the one proposed in this thesis (see especially Section 2.3.2), taking into account also other language settings. For instance, testing the approach on other languages for which both irony-annotated datasets and UD resources are available. A starting point towards such an expansion could be Arabic, for which a dataset annotated for irony is already available [Ghanem et al., 2020] and also a treebank of user-generated content exists [Seddah et al., 2020]. Furthermore it would be interesting to deal with a non-European language in order to assess how irony is uttered in other cultures, as well. Another option, regarding irony detection, could be to expand this line of investigation to German, since at least one corpus[1] exists and a treebank of German tweets [Rehbein et al., 2019] is also available and could, thus, be used for training a model based on dependency syntax.

For what concerns stance detection, we have seen how relying only on morphosyntactic cues is not sufficiently strong as approach and does not lead to a good performance overall. On the other hand, in Section 3.2.1 I described my participation in the *StanceCat* shared task, in which with my colleagues, we investigated the importance of contextual features. Those features seem to be highly informative for stance detection, which has been later on confirmed by the outcomes of the official rankings of the *SardiStance* shared task. In fact, participants combining NLP methods with contextual features extracted by the network structure could increase their performances up to 8.6% in terms of $F1_{avg}$.

Furthermore, within the experience matured in the *SardiStance* shared task, my attention was caught by one of the participating system proposed, whose method exploited the information regarding irony as an "auxiliary task" in order to later on detect the author's stance from a tweet, obtaining very good results [Giorgioni et al., 2020]. Subsequently, as I anticipated in Chapter 4 and in the previous section, I was then led to think that the most efficient way of dealing with the two tasks (irony detection on one side and stance detection on the other side), is that of treating irony detection as a sort of *preprocessing step* before detecting stance.

In fact, Stance detection is a relatively new computational linguistic task that is rapidly gaining exposure in different research communities. State-of-the-art approaches achieve results that are not far from those obtained by the baselines proposed in shared tasks, meaning that there is still plenty to do for improving performances in the stance detection task.

In the immediate future, I would like to test an approach based on morphosyntactic and contextual features both inside the upcoming shared task *VaxxStance*[2] on Basque and Spanish tweets. In participating in the shared task, I will aim at exploiting the

[1] http://kti.tugraz.at/staff/rkern/courses/kddm2/2018/reports/team-27.pdf.

[2] The task will take place between March and September 2021: https://vaxxstance.github.io/.

useful lessons learned in the last three years. Furthermore, I would like to follow the main intuition that led to the creation of my PhD path and the creation of the present thesis, and also to follow in the footsteps of what anticipated by one of the participating teams in *SardiStance*. In fact, they speculated that irony – among other characteristics – might be a useful feature in detecting stance, therefore, I plan to create a machine learning framework in which irony is used as an "auxiliary task", i.e., as a preprocessing step before detecting stance.

Bibliography

A. Aker, L. Derczynski, and K. Bontcheva. Simple Open Stance Classification for Rumour Analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2017)*. INCOMA Ltd., 2017.

F. Albogamy and A. Ramsay. Universal Dependencies for Arabic tweets. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, (RANLP 2017)*. INCOMA Ltd., 2017.

A. AlDayel and W. Magdy. Stance Detection on Social Media: State of the Art and Trends. *Information Processing & Management*, 58(4), 2021.

R. Alkhalifa and A. Zubiaga. QMUL-SDS @ SardiStance: Leveraging Network Interactions to Boost Performance on Stance Detection using Knowledge Graphs. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

R. Artstein and M. Poesio. Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

S. Attardo. Irony markers and functions: Towards a goal-oriented theory of irony and its processing. *Rask: International journal of language and communication*, 12(1):3–20, 2000.

I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*. ACL, 2016.

H. Bahuleyan and O. Vechtomova. UWaterloo at SemEval-2017 Task 8: Detecting Stance Towards Rumours with Topic Independent Features. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017)*. ACL, 2017.

D. Bamman and N. A. Smith. Contextualized sarcasm detection on Twitter. In *Proceedings of the 9th International Conference on Web and Social Media, (ICWSM 2015)*. AAAI, 2015.

F. Barbieri and H. Saggion. Modelling irony in Twitter: Feature analysis and evaluation. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*. ELRA, 2014.

F. Barbieri, H. Saggion, and F. Ronzano. Modelling sarcasm in Twitter, a novel approach. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. ACL, 2014.

F. Barbieri, V. Basile, D. Croce, M. Nissim, N. Novielli, and V. Patti. Overview of the Evalita 2016 SENTIment POLarity Classification Task. In *Proceedings of 3rd Italian Conference on Computational Linguistics (CLiC-it 2016) & 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*. CEUR-WS.org, 2016.

A. Baruah, K. Das, F. Barbhuiya, and K. Dey. Context-aware sarcasm detection using bert. In *Proceedings of the 2nd Workshop on Figurative Language Processing*. ACL, 2020.

P. Basile and G. Semeraro. UNIBA - Integrating distributional semantics features in a supervised approach for detecting irony in Italian tweets. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

V. Basile, A. Bolioli, V. Patti, P. Rosso, and M. Nissim. Overview of the EVALITA 2014 SENTIment POLarity Classification task. In *Proceedings of the 4th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2014)*. Pisa University Press, 2014.

V. Basile, M. Lai, and M. Sanguinetti. Long-term Social Media Data Collection at the University of Turin. In *Proceedings of the 5th Italian Conference on Computational Linguistics (CLiC-it 2018)*. CEUR-WS.org, 2018.

E. Bassignana, V. Basile, and V. Patti. Hurtlex: A Multilingual Lexicon of Words to Hurt. In *Proceedings of the 5th Italian Conference on Computational Linguistics (CLiC-it 2018)*. CEUR-WS.org, 2018.

C. Bazzanella. Oscillazioni di informalità e formalità: scritto, parlato e rete. *Formale e informale. La variazione di registro nella comunicazione elettronica*, 2011.

S. Behzad and A. Zeldes. A Cross-Genre Ensemble Approach to Robust Reddit Part of Speech Tagging. In *Proceedings of the 12th Web as Corpus Workshop (WAC-XII)*. ELRA, 2020.

F. Benamara, C. Grouin, J. Karoui, V. Moriceau, and I. Robba. Analyse d'opinion et langage figuratif dans des tweets: présentation et résultats du Défi Fouille de Textes DEFT2017. In *Actes de l'atelier DEFT 2017 associé à la conférence TALN*. ATALA, 2017.

M. Bennici. ghostwriter19 @ SardiStance: Generating new tweets to classify SardiStance EVALITA 2020 political tweets. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

B. Bharathi, J. Bhuvana, and N. N. Appiah Balaji. SardiStance@EVALITA2020: Textual and Contextual stance detection from Tweets using machine learning approach. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

I. Bhat, R. A. Bhat, M. Shrivastava, and D. Sharma. Universal Dependency Parsing for Hindi-English Code-Switching. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (NAACL-HLT 2018)*. ACL, 2018.

S. L. Blodgett, J. Wei, and B. O'Connor. Twitter universal dependency parsing for African-American and mainstream American English. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL 2018)*, 2018.

C. Bosco, V. Patti, and A. Bolioli. Developing Corpora for Sentiment Analysis: The Case of Irony and Senti-TUT. *IEEE Intelligent Systems*, 28(2):55–63, 2013.

C. Bosco, T. Fabio, B. Andrea, and A. Mazzei. Overview of the EVALITA 2016 Part Of Speech on TWitter for ITAlian Task. In *Proceedings of 3rd Italian Conference on Computational Linguistics (CLiC-it 2016) & 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2016)*. CEUR-WS.org, 2016a.

C. Bosco, M. Lai, V. Patti, F. M. Rangel Pardo, and P. Rosso. Tweeting in the Debate about Catalan Elections. In *Proceedings of the Workshop on Emotion and Sentiment Analysis (ESA 2016)*. ELRA, 2016b.

C. Bosco, F. Dell'Orletta, F. Poletto, M. Sanguinetti, and M. Tesconi. Overview of the EVALITA 2018 Hate Speech Detection Task. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

A. Bowes and A. Katz. When sarcasm stings. *Discourse Processes: A Multidisciplinary Journal*, 48(4):215–236, 2011.

M. Buytaert. *A Machine Learning Approach to Sentiment Analysis and Stance Detection for Political Tweets*. PhD thesis, Ghent University, 2018.

E. Cambria, D. Olsher, and D. Rajagopal. SenticNet 3: a Common and Common-sense Knowledge Base for Cognition-driven Sentiment Analysis. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*. ACM, 2014.

P. Carvalho, L. Sarmento, M. J. Silva, and E. de Oliveira. Clues for detecting irony in user-generated contents: Oh...!! it's "so easy" ;-). In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion (TSA 2009)*. ACM, 2009.

G. Castellucci, D. Croce, and R. Basili. A language independent method for generating large scale polarity lexicons. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. ELRA, 2016.

Ö. Çetinoğlu and Ç. Çöltekin. Part of Speech Annotation of a Turkish-German Code-Switching Corpus. In *Proceedings of the 10th Linguistic Annotation Workshop (LAW-X)*. ACL, 2016.

R. Chakhachiro. *Translating Irony between English and Arabic*. Cambridge Scholars Publishing, 2019.

L. Chiruzzo, S. Castro, M. Etcheverry, D. Garat, J. J. Prada, and A. Rosá. Overview of HAHA at IberLEF 2019: Humor Analysis based on Human Annotation. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019)*. CEUR-WS.org, 2019.

Y. Choi and J. Wiebe. +/-effectwordnet: Sense-level Lexicon Acquisition for Opinion Inference. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*. ACL, 2014.

G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini. Computational Fact Checking from Knowledge Networks. *PloS one*, 10(6), 2015.

A. T. Cignarella. A Fine-grained Annotation of Irony in Italian Social Media Texts. Master's thesis, Università degli Studi di Torino, Dipartimento di Studi Umanistici, *Laurea Magistrale in Scienze Linguistiche (LM-39)*, 2017.

A. T. Cignarella and C. Bosco. ATC at IroSvA 2019: Shallow Syntactic Dependency-based Features for Irony Detection in Spanish Variants. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019)*. CEUR-WS.org, 2019.

A. T. Cignarella, C. Bosco, and V. Patti. TWITTIRÒ: a Social Media Corpus with a Multi-layered Annotation for Irony. In *Proceedings of the 4th Italian Conference on Computational Linguistics (CLiC-it 2017)*. CEUR-WS.org, 2017.

A. T. Cignarella, C. Bosco, V. Patti, and M. Lai. Application and Analysis of a Multi-layered Scheme for Irony on the Italian Twitter Corpus TWITTIRÒ. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC-2018)*. ELRA, 2018a.

A. T. Cignarella, S. Frenda, V. Basile, C. Bosco, V. Patti, and P. Rosso. Overview of the EVALITA 2018 Task on Irony Detection in Italian Tweets (IronITA). In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018b.

A. T. Cignarella, C. Bosco, V. Patti, and M. Lai. TWITTIRÒ: an Italian Twitter Corpus with a Multi-layered Annotation for Irony. *IJCoL - Italian Journal of Computational Linguistics*, 4(2):25–44, 2019a.

A. T. Cignarella, C. Bosco, and P. Rosso. Presenting TWITTIRO-UD: An Italian Twitter Treebank in Universal Dependencies. In *Proceedings of the 5th International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*. ACL, 2019b.

A. T. Cignarella, M. Sanguinetti, C. Bosco, and P. Rosso. Is This an Effective Way to Annotate Irony Activators? In *Proceedings of the 6th Italian Conference on Computational Linguistics (CLiC-it 2019)*. CEUR-WS.org, 2019c.

A. T. Cignarella, V. Basile, M. Sanguinetti, C. Bosco, F. Benamara, and P. Rosso. Multilingual Irony Detection with Dependency Syntax and Neural Models. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*. ACL, 2020a.

A. T. Cignarella, M. Lai, C. Bosco, V. Patti, and P. Rosso. SardiStance@EVALITA2020: Overview of the Stance Detection in Italian Tweets. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*, 2020b.

A. T. Cignarella, M. Sanguinetti, C. Bosco, and P. Rosso. Marking Irony Activators in a Universal Dependencies Treebank: The Case of an Italian Twitter Corpus. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC 2020)*. ELRA, 2020c.

A. Cimino, L. De Mattei, and F. Dell'Orletta. Multi-task learning in deep neural networks at EVALITA 2018. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

H. H. Clark and R. J. Gerrig. *On the Pretense Theory of Irony.* American Psychological Association, 1984.

K. Darwish, P. Stefanov, M. Aupetit, and P. Nakov. Unsupervised User Stance Detection on Twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 141–152, 2020.

D. Davidov, O. Tsur, and A. Rappoport. Semi-supervised Recognition of Sarcastic Sentences in Twitter and Amazon. In *Proceedings of the 14th Conference on Computational Natural Language Learning (CoNLL 2010)*. ACL, 2010.

J. C. De Albornoz, L. Plaza, and P. Gervás. SentiSense: An Easily Scalable Concept-based Affective Lexicon for Sentiment Analysis. In *Proocedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*. ELRA, 2012.

M.-C. De Marneffe and C. D. Manning. The Stanford typed dependencies representation. In *COLING 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation)*. ACM, 2008.

M.-C. De Marneffe, B. MacCartney, C. D. Manning, et al. Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*. ELRA, 2006.

M.-C. De Marneffe, T. Dozat, N. Silveira, K. Haverinen, F. Ginter, J. Nivre, and C. D. Manning. Universal Stanford dependencies: A cross-linguistic typology. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*. ELRA, 2014.

M.-C. De Marneffe, C. D. Manning, J. Nivre, and D. Zeman. Universal Dependencies. *Computational Linguistics*, 47(2):255–308, 2021.

L. De Mattei, A. Cimino, and F. Dell'Orletta. Multi-task learning in Deep Neural Networks for Irony Detection. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

D. DellaPosta, Y. Shi, and M. Macy. Why do liberals drink lattes? *American Journal of Sociology*, 120(5):1473–1511, 2015.

R. Delmonte. A linguistic rule-based system for pragmatic text processing. In *Proceedings of the 4th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2014)*. Pisa University Press, 2014.

R. Delmonte. Venses @ HaSpeeDe2 & SardiStance: Multilevel Deep Linguistically Based Supervised Approach to Classification. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga. SemEval-2017 Task 8: RumourEval: Determining Rumour Veracity and Support for Rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017)*. ACL, 2017.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

K. Dey, R. Shrivastava, and S. Kaushik. Topical stance detection for twitter: A two-phase lstm model using attention. In *European Conference on Information Retrieval (ECIR 2018)*. Springer, 2018.

E. Di Rosa and A. Durante. Irony detection in tweets: X2Check at Ironita 2018. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

C. C. Du Marsais, J. Paulhan, and C. Mouchard. *Traité des tropes*. Le Nouveau Commerce, 1981.

M. Dynel. Linguistic approaches to (non) humorous irony. *Humor - International Journal of Humor Research*, 27(6):537–550, 2014.

J. Eisenstein. What to Do about Bad Language on the Internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013)*. ACL, 2013.

M. S. Espinosa, R. Agerri, A. Rodrigo, and R. Centeno. DeepReading @ SardiStance: Combining Textual, Social and Emotional Features. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*. ELRA, 2006.

A. Esuli and F. Sebastiani. SentiWordNet: a High-coverage Lexical Resource for Opinion Mining. *Evaluation*, 17:1–26, 2007.

I. A. Farha and W. Magdy. From Arabic Sentiment Analysis to Sarcasm Detection: The ArSarcasm Dataset. In *Proceedings of the Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools co-located inside the 12th International Conference on Language Resources and Evaluation (LREC 2020)*. ELRA, 2020.

E. Fast, B. Chen, and M. S. Bernstein. Empath: Understanding Topic Signals in Large-scale Text. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 2016.

F. Ferraccioli, A. Sciandra, M. Da Pont, P. Girardi, D. Solari, and L. Finos. TextWiller @ SardiStance, HaSpeede2: Text or Con-text? A smart use of social network data in predicting polarization. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

J. R. Firth. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 1957.

J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 1971.

FNC. Fake News Challenge Stage-1 (FNC-1): Stance Detection. http://www.fakenewschallenge.org/, 2017.

J. Foster. "cba to check the spelling": Investigating Parser Performance on Discussion Forum Posts. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*. ACL, 2010.

S. Frenda, B. Ghanem, and M. Montes-y Gómez. Exploration of Misogyny in Spanish and English Tweets. In *Proceedings of the 3rd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018)*. CEUR-WS.org, 2018.

B. Ghanem, P. Rosso, and F. Rangel. Stance Detection in Fake News A Combined Feature Representation. In *Proceedings of the 1st Workshop on Fact Extraction and VERification (FEVER)*, pages 66–71, 2018.

B. Ghanem, A. T. Cignarella, C. Bosco, P. Rosso, and F. M. Rangel Pardo. UPV-28-UNITO at SemEval-2019 Task 7: Exploiting Post's Nesting and Syntax Information for Rumor Stance Classification. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval 2019)*. ACL, 2019a.

B. Ghanem, J. Karoui, F. Benamara, V. Moriceau, and P. Rosso. IDAT at FIRE2019: Overview of the Track on Irony Detection in Arabic Tweets. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*. ACM, 2019b.

B. Ghanem, J. Karoui, F. Benamara, P. Rosso, and V. Moriceau. Irony detection in a multilingual context. In *Proceedings of the European Conference on Information Retrieval (ECIR 2020)*. Springer, 2020.

A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, J. Barnden, and A. Reyes. Semeval-2015 Task 11: Sentiment Analysis of Figurative Language in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. ACL, 2015.

D. Ghosh, A. Vajpayee, and S. Muresan. A Report on the 2020 Sarcasm Detection Shared Task. In *Proceedings of the 2nd Workshop on Figurative Language Processing (FigLang 2020)*. ACL, 2020.

R. W. Gibbs. Irony in talk among friends. *Metaphor and symbol*, 15(1-2):5–27, 2000.

K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. ACL, 2011.

R. Giora. On irony and negation. *Discourse processes*, 19(2):239–264, 1995.

S. Giorgioni, M. Politi, S. Salman, D. Croce, and R. Basili. UNITOR@Sardistance2020: Combining Transformer-based architectures and Transfer Learning for robust Stance Detection. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

V. Giudice. Aspie96 at IronITA (EVALITA 2018): Irony Detection in Italian Tweets with Character-Level Convolutional RNN. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

J. González, L.-F. Hurtado, and F. Pla. ELiRF-UPV at IroSvA: Transformer Encoders for Spanish Irony Detection. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019)*. CEUR-WS.org, 2019.

J. Á. González, L.-F. Hurtado, and F. Pla. Transformer based contextualization of pre-trained word embeddings for irony detection in twitter. *Information Processing & Management*, 57(4):102262, 2020.

R. González-Ibáñez, S. Muresan, and N. Wacholder. Identifying sarcasm in Twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. ACL, 2011.

G. Gorrell, E. Kochkina, M. Liakata, A. Aker, A. Zubiaga, K. Bontcheva, and L. Derczynski. SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval 2019)*. ACL, 2019.

P. H. Grice. Logic and Conversation. *Syntax and Semantics 3: Speech Arts*, pages 41–58, 1975.

P. H. Grice. Further Notes on Logic and Conversation. *Pragmatics*, 1:13–128, 1978.

M. Hajjem, C. Latiri, and Y. Slimani. Twitter as a Multilingual Source of Comparable Corpora. In *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia*. ACM, 2014.

A. Hanselowski, P. Avinesh, B. Schiller, F. Caspelherr, D. Chaudhuri, C. M. Meyer, and I. Gurevych. A Retrospective Analysis of the Fake News Challenge Stance-Detection Task. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*. ACL, 2018.

D. Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea. CASCADE: Contextual Sarcasm Detection in Online Discussion Forums. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*. ACL, 2018.

D. I. Hernández Farías, J.-M. Benedí, and P. Rosso. Applying basic features from sentiment analysis for automatic irony detection. In *Proceedings of the 7th Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2015)*. Springer, 2015.

D. I. Hernańdez Farías, V. Patti, and P. Rosso. Irony detection in Twitter: The role of affective content. *ACM Transactions on Internet Technology (TOIT)*, 16(3):19, 2016.

D. I. Hernández Farías and P. Rosso. Irony, Sarcasm, and Sentiment Analysis. In *Sentiment Analysis in Social Networks*, pages 113–128. Elsevier Science and Technology, 2016.

M. Hu and B. Liu. Mining and Summarizing Customer Reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2004.

Y.-H. Huang, H.-H. Huang, and H.-H. Chen. Irony detection with attentive Recurrent Neural Networks. In *European Conference on Information Retrieval (ECIR 2017)*. Springer, 2017.

A. Joshi, V. Sharma, and P. Bhattacharyya. Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. ACL, 2015.

A. Joshi, P. Bhattacharyya, and M. J. Carman. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):73, 2017.

A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

D. Jurafsky and J. H. Martin. Speech and language processing: An introduction to speech recognition, computational linguistics and natural language processing. *Upper Saddle River, NJ: Prentice Hall*, 2008.

H. Kanayama and R. Iwamoto. How Universal are Universal Dependencies? Exploiting Syntax for Multilingual Clause-level Sentiment Detection. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*. ELRA, 2020.

A. Karima and K. Smaili. Measuring the comparability of multilingual corpora extracted from Twitter and others. In *HrTAL2016-Tenth International Conference on Natural Language Processing*, 2016.

J. Karoui. *Détection automatique de l'ironie dans les contenus générés par les utilisateurs*. PhD thesis, Université Toulouse 3 Paul Sabatier; Faculté des Sciences Economiques et de Gestion, Université de Sfax (Tunisie), 2017.

J. Karoui, F. Benamara, V. Moriceau, N. Aussenac-Gilles, and L. H. Belguith. Towards a contextual pragmatic model to detect irony in tweets. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*, 2015.

J. Karoui, F. Benamara, V. Moriceau, V. Patti, and C. Bosco. Exploring the Impact of Pragmatic Phenomena on Irony Detection in Tweets: A Multilingual Corpus Study. In *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL 2017)*. ACL, 2017.

S. Kayalvizhi, D. Thenmozhi, and C. Aravindan. SSN_NLP@SardiStance : Stance Detection from Italian Tweets using RNN and Transformers. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

M. Khodak, N. Saunshi, and K. Vodrahalli. A Large Self-Annotated Corpus for Sarcasm. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*. ELRA, 2018.

J. Kim, J.-J. Li, and J.-H. Lee. Evaluating multilanguage-comparability of subjectivity analysis systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. ACL, 2010.

L. Kong, N. Schneider, S. Swayamdipta, A. Bhatia, C. Dyer, and N. A. Smith. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*. ACL, 2014.

E. Kouloumpis, T. Wilson, and J. D. Moore. Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the ICWSM: International AAAI Conference on Web and Social Media*. AAAI, 2011.

P. Kralj Novak, J. Smailović, B. Sluban, and I. Mozetič. Sentiment of emojis. *PLoS one*, 10(12), 2015.

D. Küçük and F. Can. Stance detection: A survey. *ACM Computing Surveys (CSUR)*, 53(1):1–37, 2020.

F. Kunneman, C. Liebrecht, M. van Mulken, and A. van den Bosch. Signaling sarcasm: From hyperbole to hashtag. *Information Processing & Management*, 51(4):500 – 509, 2015.

M. Lai. *Language and Structure in Polarized Communities*. PhD thesis, Università degli Studi di Torino and Universitat Politècnica de València (co-tutelle), 2019.

M. Lai, A. T. Cignarella, and D. I. Hernandez Fariás. iTACOS at IberEval2017: Detecting Stance in Catalan and Spanish Tweets. In *Proceedings of the 2nd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2017)*. CEUR-WS.org, 2017a.

M. Lai, D. I. Hernández Farías, V. Patti, and P. Rosso. Friends and Enemies of Clinton and Trump: Using Context for Detecting Stance in Political Tweets. In *Proceedings of the 15th Mexican International Conference on Artificial Intelligence (MICAI 2016)*. Springer, 2017b.

M. Lai, M. Tambuscio, V. Patti, G. Ruffo, and P. Rosso. Extracting Graph Topological Information and Users' Opinion. In *Proceedings of the 8th International Conference of the CLEF Association (CLEF 2017)*. Springer, 2017c.

M. Lai, V. Patti, G. Ruffo, and P. Rosso. Stance Evolution and Twitter Interactions in an Italian Political Debate. In *Natural Language Processing and Information Systems*. Springer, 2018.

M. Lai, M. Tambuscio, V. Patti, G. Ruffo, and P. Rosso. Stance polarity in political debates: A diachronic perspective of network homophily and conversations on twitter. *Data & Knowledge Engineering*, 124, 2019.

M. Lai, A. T. Cignarella, D. I. H. Farías, C. Bosco, V. Patti, and P. Rosso. Multilingual stance detection in social media political debates. *Computer Speech & Language*, 63(101075), 2020a.

M. Lai, V. Patti, G. Ruffo, and P. Rosso. #Brexit: Leave or remain? the role of user's community and diachronic evolution on stance detection. *Journal of Intelligent & Fuzzy Systems*, pages 1–12, 2020b.

M. Lai, A. T. Cignarella, L. Finos, and A. Sciandra. WordUp! at VaxxStance 2021: Combining Contextual Information with Textual and Dependency-Based Syntactic Features for Stance Detection. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021), in press*. CEUR-WS.org, 2021.

C. J. Lee and A. N. Katz. The differential role of ridicule in sarcasm and irony. *Metaphor and Symbol*, 13(1):1–15, 1998.

O. Levy and Y. Goldberg. Dependency-Based Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*. ACL, 2014.

M. Liakata, J.-H. Kim, S. Saha, J. Hastings, and D. Rebholz-Schuhmann. Three hybrid classifiers for the detection of emotions in suicide notes. *Biomedical informatics insights*, 5(Suppl. 1):175, 2012.

L. Liu, D. Zhang, and W. Song. Exploiting syntactic structures for humor recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics (ACL), 2018a.

Y. Liu, Y. Zhu, W. Che, B. Qin, N. Schneider, and N. A. Smith. Parsing Tweets into Universal Dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2018)*. ACL, 2018b.

V. Lyding, E. Stemle, C. Borghetti, M. Brunello, S. Castagnoli, F. Dell'Orletta, H. Dittmann, A. Lenci, and V. Pirrelli. The PAISA' Corpus of Italian Web Texts. In *Proceedings of the 9th World Archaeological Congress (WAC-9) co-located with the*

16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014). ACL, 2014.

T. Lynn, K. Scannell, and E. Maguire. Minority Language Twitter: Part-of-Speech Tagging and Analysis of Irish Tweets. In *Proceedings of the Workshop on Noisy User-generated Text*. ACL, 2015.

N. MacLeod and T. Grant. Whose Tweet? Authorship analysis of micro-blogs and other short-form messages. In *Proceedings of the International Association of Forensic Linguists' 10th Biennial Conference*. Aston University, 2012.

W. Magdy, K. Darwish, N. Abokhodair, A. Rahimi, and T. Baldwin. #isisisnotislam or #deportallmuslims?: Predicting unspoken views. In *Proceedings of the 8th ACM Conference on Web Science (WebSci 2016)*. ACM, 2016.

D. Maynard and A. Funk. Automatic Detection of Political Opinions in Tweets. In *Proceedings of the ESWC: Extended Semantic Web Conference (ESWC 2011)*. Springer, 2011.

R. McDonald, J. Nivre, Y. Quirmbach-Brundage, Y. Goldberg, D. Das, K. Ganchev, K. Hall, S. Petrov, H. Zhang, O. Täckström, et al. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*. ACL, 2013.

L. A. Michaelis and H. Feng. What is this, sarcastic syntax? *Constructions and Frames*, 7(2):148–180, 2015.

R. Mihalcea and S. Pulman. Characterizing Humour: An Exploration of Features in Humorous Texts. In *Proceedinggs of the 8th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2007)*. Springer, 2007.

R. Millar. *Language, Nation and Power: An Introduction*. Springer, 2005.

S. Mohammad and P. D. Turney. Crowdsourcing a Word-Emotion Association Lexicon. *Computational Intelligence*, 29(3):436–465, 2013.

S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry. SemEval-2016 Task 6: Detecting Stance in Tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. ACL, 2016.

S. M. Mohammad and P. D. Turney. Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*. ACL, 2010.

S. M. Mohammad, P. Sobhani, and S. Kiritchenko. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):1–23, 2017.

D. Mollá and A. Joshi. Overview of the 2019 ALTA Shared Task: Sarcasm Target Identification. In *Proceedings of the 17th Annual Workshop of the Australasian Language Technology Association (ALTA 2019)*. ALTA, 2019.

M. Moraca, G. Sabella, and S. Morra. UninaStudents @ SardiStance: Stance detection in Italian tweets - Task A. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*. CEUR-WS.org, 2020.

F. Å. Nielsen. A New ANEW: Evaluation of a Word List for Sentiment Analysis in Microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages*. CEUR-WS.org, 2011a.

F. Å. Nielsen. Afinn, 2011b. URL https://github.com/fnielsen/afinn.

J. Nivre, M.-C. De Marneffe, F. Ginter, Y. Goldberg, J. Hajic, C. D. Manning, R. T. McDonald, S. Petrov, S. Pyysalo, N. Silveira, et al. Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. ELRA, 2016.

J. Nivre, M.-C. De Marneffe, F. Ginter, J. Hajič, C. D. Manning, S. Pyysalo, S. Schuster, F. Tyers, and D. Zeman. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*. ELRA, 2020.

R. Ortega, F. Rangel, D. I. Hernández Farías, P. Rosso, M. Montes, and J. E. Medina. Overview of the Task on Irony Detection in Spanish Variants. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019)*. CEUR-WS.org, 2019.

R. Ortega-Bueno and J. E. Medina Pagola. UO_IRO: Linguistic informed deep-learning model for irony detection. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith. Improved Part-of-Speech Tagging for Online Conversational Text with Word Clusters. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2013)*, 2013.

A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedinggs of the 7th Conference on Language Resources and Evaluation (LREC 2010)*. ELRA, 2010.

A. Pak and P. Paroubek. Twitter for sentiment analysis: When language resources are not available. In *Proceedings of the 2011 22nd International Workshop on Database and Expert Systems Applications (DEXA 2011)*. IEEE, 2011.

B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2008.

J. W. Pennebaker, M. E. Francis, and R. J. Booth. Linguistic Inquiry and Word Count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, 71, 2001.

S. Petrov and R. McDonald. Overview of the 2012 Shared Task on Parsing the Web. In *Notes of the 1st Workshop on Syntactic Analysis of Non-Canonical Language (SANCL 2012)*, 2012.

S. Petrov, D. Das, and R. McDonald. A Universal Part-of-Speech Tagset. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*. ELRA, 2012.

F. Poletto, M. Stranisci, M. Sanguinetti, V. Patti, and C. Bosco. Hate speech annotation: Analysis of an Italian Twitter corpus. In *Proceedings of the 4th Italian Conference on Computational Linguistics (CLiC-it 2017)*. CEUR-WS.org, 2017.

M. Polignano, P. Basile, M. de Gemmis, G. Semeraro, and V. Basile. ALBERTO: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In *Proceedings of the 6th Italian Conference on Computational Linguistics (CLiC-it 2019)*. CEUR-WS.org, 2019.

S. Poria, A. Gelbukh, A. Hussain, N. Howard, D. Das, and S. Bandyopadhyay. Enhanced SenticNet with Affective Labels for Concept-based Opinion Mining. *IEEE Intelligent Systems*, 28(2):31–38, 2013.

J.-P. Posadas-Duran, G. Sidorov, and I. Batyrshin. Complete syntactic n-grams as style markers for authorship attribution. In *Proceedings of the Mexican International Conference on Artificial Intelligence (MICAI 2014)*. Springer, 2014.

R. Procter, F. Vis, and A. Voss. Reading the Riots on Twitter: Methodological Innovation for the Analysis of Big Data. *International Journal of Social Research Methodology*, 16(3):197–214, 2013.

T. Proisl. Someweta: A Part-of-Speech Tagger for German Social Media and Web Texts. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*. ELRA, 2018.

T. Ptáček, I. Habernal, and J. Hong. Sarcasm detection on Czech and English Twitter. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*. Dublin City University and ACL, 2014.

V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei. Rumor has it: Identifying Misinformation in Microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*. ACL, 2011.

U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

A. Rajadesingan and H. Liu. Identifying users with opposing opinions in Twitter debates. In *Proceedings of the 7th International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction (SBP 2014)*. Springer, 2014.

A. Rajadesingan, R. Zafarani, and H. Liu. Sarcasm detection on Twitter: A behavioral modeling approach. In *Proceedings of the 8th ACM International Conference on Web Search and Data Mining (WSDM 2015)*. ACM, 2015.

F. Rangel, P. Rosso, and M. Franco-Salvador. A low dimensionality representation for language variety identification. In *Proceedings of the 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2016)*. Springer, 2018.

A. Rashed, M. Kutlu, K. Darwish, T. Elsayed, and C. Bayrak. Embeddings-Based Clustering for Target Specific Stances: The Case of a Polarized Turkey. *arXiv preprint arXiv:2005.09649*, 2020.

K. Ravi and V. Ravi. A novel automatic satire and irony detection using ensembled feature selection and data mining. *Knowledge-Based Systems*, 120:15–33, 2017.

I. Rehbein, J. Ruppenhofer, and V. Zimmermann. A harmonised testsuite for POS tagging of German social media data. In *Proceedings of the 14th Conference on Natural Language Processing (KONVENS 2018)*. Austrian Academy of Sciences, 2018.

I. Rehbein, J. Ruppenhofer, and B.-N. Do. tweeDe – A Universal Dependencies treebank for German tweets. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*. ACL, 2019.

P. Resnick, S. Carton, S. Park, Y. Shen, and N. Zeffer. Rumorlens: A System for Analyzing the Impact of Rumors and Corrections in Social Media. In *Proceedings of the 2014 Computation+Journalism Symposium*, 2014.

A. Reyes and P. Rosso. On the difficulty of automatically detecting irony: beyond a simple case of negation. *Knowledge and Information Systems*, 40(3):595–614, 2014.

A. Reyes, P. Rosso, and D. Buscaldi. Finding humour in the blogosphere: the role of wordnet resources. In *Proceedings of the 5th Global WordNet Conference*. Narosa Publishing House, 2010.

A. Reyes, P. Rosso, and D. Buscaldi. From Humor Recognition to Irony Detection: The Figurative Language of Social Media. *Data & Knowledge Engineering*, 74:1–12, 2012.

A. Reyes, P. Rosso, and T. Veale. A Multidimensional Approach for Detecting Irony in Twitter. *Language Resources and Evaluation*, 47(1):239–268, 2013.

E. Riloff, A. Qadir, P. Surve, L. D. Silva, N. Gilbert, and R. Huang. Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, (EMNLP 2013)*. ACL, 2013.

M. Rouvier and P.-M. Bousquet. LIA@DEFT'2017: Multi-view Ensemble of Convolutional Neural Network. In *Actes de l'atelier «Défi Fouille de Textes» (DEFT 2017) dans la 16ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2017)*. ATALA, 2017.

I. Russo, F. Frontini, and V. Quochi. OpeNER Sentiment Lexicon Italian - LMF, 2016. URL http://hdl.handle.net/20.500.11752/ILC-73. ILC-CNR for CLARIN-IT repository hosted at the Institute for Computational Linguistics "A. Zampolli", National Research Council in Pisa.

M. Sahlgren. An introduction to random indexing. In *Proceedings of the Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering (TKE 2005)*, 2005.

H. Saif, Y. He, M. Fernandez, and H. Alani. Contextual Semantics for Sentiment Analysis of Twitter. *Information Processing & Management*, 52(1):5–19, 2016.

Y. Samih and K. Darwish. A Few Topical Tweets are Enough for Effective User Stance Detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. ACL, 2021.

M. Sanguinetti, C. Bosco, A. Lavelli, A. Mazzei, O. Antonelli, and F. Tamburini. PoSTWITA-UD: an Italian Twitter Treebank in Universal Dependencies. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC-2018)*. ELRA, 2018.

M. Sanguinetti, C. Bosco, L. Cassidy, Özlem. Çetinoğlu, A. T. Cignarella, T. Lynn, I. Rehbein, J. Ruppenhofer, D. Seddah, and A. Zeldes. Treebanking User-Generated Content: A Proposal for a Unified Representation in Universal Dependencies. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC 2020)*. ELRA, 2020.

M. Sanguinetti, C. Bosco, L. Cassidy, Özlem. Çetinoğlu, A. T. Cignarella, T. Lynn, I. Rehbein, J. Ruppenhofer, D. Seddah, and A. Zeldes. Treebanking User-Generated Content: A UD Based Overview of Guidelines, Corpora and Unified Recommendations. *Under review / pending decision: submitted to the Special Issue "Annotation of non-standard corpora" at Language Resources and Evaluation Journal (LREJ)*, 2021.

A. Santilli, D. Croce, and R. Basili. A Kernel-based Approach for Irony and Sarcasm Detection in Italian. In *Proceedings of the 6th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2018)*. CEUR-WS.org, 2018.

H. Schmid. Part-of-Speech Tagging with Neural Networks. In *Proceedings of the 15th Conference on Computational Linguistics (COLING 1994)*. ACL, 1994.

H. Schmid. Treetagger | A Language Independent Part-of-Speech Tagger. *Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart*, 43:28, 1995.

D. Seddah, B. Sagot, M. Candito, V. Mouilleron, and V. Combet. The French Social Media Bank: A Treebank of Noisy User Generated Content. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*. The COLING 2012 Organizing Committee, 2012.

D. Seddah, F. Essaidi, A. Fethi, M. Futeral, B. Muller, P. J. O. Suárez, B. Sagot, and A. Srivastava. Building a user-generated content North-African Arabizi Treebank: Tackling hell. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. ACL, 2020.

G. Sidorov. Should Syntactic N-grams Contain Names of Syntactic Relations? *International Journal of Computational Linguistics Applications*, 5(2):25–47, 2014.

G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández. Syntactic dependency-based n-grams as classification features. In *Proceedings of the Mexican International Conference on Artificial Intelligence (MICAI 2012)*. Springer, 2012.

G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández. Syntactic dependency-based n-grams: More evidence of usefulness in classification. In *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2013)*. Springer, 2013.

G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández. Syntactic n-grams as Machine Learning Features for Natural Language Processing. *Expert Systems with Applications*, 41(3):853–860, 2014.

N. Silveira, T. Dozat, M.-C. De Marneffe, S. R. Bowman, M. Connor, J. Bauer, and C. D. Manning. A gold standard dependency corpus for English. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*. ELRA, 2014.

M. Simi, C. Bosco, and S. Montemagni. Less is More? Towards a Reduced Inventory of Categories for Training a Parser for the Italian Stanford Dependencies. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*. ELRA, 2014.

R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*. ACL, 2013.

S. Somasundaran and J. Wiebe. Recognizing Stances in Online Debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. ACL, 2009.

D. Sperber and D. Wilson. Irony and the Use-Mention Distinction. *Philosophy*, 3:143–184, 1981.

H. Srivastava, V. Varshney, S. Kumari, and S. Srivastava. A novel hierarchical bert architecture for sarcasm detection. In *Proceedings of the 2nd Workshop on Figurative Language Processing (FigLang 2020)*, 2020.

M. Straka and J. Straková. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. ACL, 2017.

M. Stranisci, C. Bosco, V. Patti, and D. I. Hernández Farías. Analyzing and Annotating for Sentiment Analysis the Socio-political Debate on #labuonascuola. In *Proceedings of the 2nd Italian Conference on Computational Linguistics (CLiC-it 2015)*. CEUR-WS.org, 2015.

M. Stranisci, C. Bosco, D. I. Hernández Farías, and V. Patti. Annotating sentiment and irony in the online Italian political debate on #labuonascuola. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. ELRA, 2016.

Q. Sun, Z. Wang, Q. Zhu, and G. Zhou. Exploring various linguistic features for stance detection. In *Natural Language Understanding and Intelligent Applications*. Springer, 2016.

Y.-J. Tang and H.-H. Chen. Chinese Irony Corpus Construction and Ironic Structure Analysis. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*. Dublin City University and ACL, 2014.

M. Taulé, M. A. Martí, and M. Recasens. AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008)*. ELRA, 2008.

M. Taulé, M. A. Martí, F. M. R. Pardo, P. Rosso, C. Bosco, and V. Patti. Overview of the Task on Stance and Gender Detection in Tweets on Catalan Independence. In *Proceedings of the 2nd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2017) co-located with 33th Conference of the Spanish Society for Natural Language Processing (SEPLN 2017)*. CEUR-WS.org, 2017.

M. Taulé, F. M. R. Pardo, M. A. Martí, and P. Rosso. Overview of the Task on Multimodal Stance Detection in Tweets on Catalan #1Oct Referendum. In *Proceedings of the 3rd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018)*. CEUR-WS.org, 2018.

M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas. Sentiment Strength Detection in Short Informal Text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558, 2010.

D. Thenmozhi et al. Sarcasm identification and detection in conversion context using bert. In *Proceedings of the 2nd Workshop on Figurative Language Processing (FigLang 2020)*. ACL, 2020.

G. Ungeheuer and H. E. Wiegand. *Handbooks of Linguistics and Communication Science*. Walter de Gruyter GmbH & Co. KG, Berlin, Germany, 2008.

A. Utsumi. A unified theory of irony and its computational formalization. In *Proceedings of the 16th Conference on Computational Linguistics*. ACL, 1996.

C. Van Hee, E. Lefever, and V. Hoste. SemEval-2018 Task 3: Irony detection in English Tweets. In *In Proceedings of the International Workshop on Semantic Evaluation (SemEval 2018)*. ACL, 2018a.

C. Van Hee, E. Lefever, and V. Hoste. We usually don't like going to the dentist: Using common sense to detect irony on Twitter. *Computational Linguistics*, 44(4):793–832, 2018b.

B. C. Wallace, D. K. Choe, and E. Charniak. Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*. ACL, 2015.

A. P. Wang. #Irony or #Sarcasm — A Quantitative and Qualitative Study Based on Twitter. In *Proceedings of the PACLIC: the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 2013)*. ACL, 2013.

H. Webb, P. Burnap, R. Procter, O. Rana, B. C. Stahl, et al. Digital Wildfires: Propagation, Verification, Regulation, and Responsible Innovation. *ACM Transactions on Information Systems (TOIS)*, 34(3):15, 2016.

P. Wei, W. Mao, and D. Zeng. A target-guided neural memory model for stance detection in twitter. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2018.

W. Wei, X. Zhang, X. Liu, W. Chen, and T. Wang. pkudblab at SemEval-2016 Task 6: A Specific Convolutional Neural Network System for Effective Stance Detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. ACL, 2016.

C. Whissell. Using the Revised Dictionary of Affect in Language to Quantify the Emotional Undertones of Samples of Natural Languages. *Psychological Reports*, 2(105): 509–521, 2009.

A. Widlöcher and Y. Mathet. La plate-forme Glozz: environnement d'annotation et d'exploration de corpus. In *Actes de la 16ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2009)*. ATALA, 2009.

T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (EMNLP 2005)*. ACL, 2005.

C. Wu, F. Wu, S. Wu, J. Liu, Z. Yuan, and Y. Huang. THU_NGN at SemEval-2018 Task 3: Tweet Irony Detection with Densely Connected LSTM and Multi-task Learning. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval 2018)*, pages 51–56. Association for Computational Linguistics (ACL), 2018.

R. Xiang, X. Gao, Y. Long, A. Li, E. Chersoni, Q. Lu, and C.-R. Huang. Ciron: a New Benchmark Dataset for Chinese Irony Detection. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*. ELRA, 2020.

R. Xu, Y. Zhou, D. Wu, L. Gui, J. Du, and Y. Xue. Overview of NLPCC shared task 4: Stance detection in chinese microblogs. In *Natural Language Understanding and Intelligent Applications*. Springer, 2016.

G. Zarrella and A. Marsh. MITRE at SemEval-2016 Task 6: Transfer Learning for Stance Detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. ACL, 2016.

D. Zeman. Reusable Tagset Conversion Using Tagset Drivers. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008)*. ELRA, 2008.

B. Zhang, M. Yang, X. Li, Y. Ye, X. Xu, and K. Dai. Enhancing Cross-target Stance Detection with Transferable Semantic-Emotion Knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*. ACL, 2020.

C. Zhang and M. Abdul-Mageed. Multi-task bidirectional transformer representations for irony detection. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, 2019.

S. Zhang, X. Zhang, J. Chan, and P. Rosso. Irony detection via sentiment-based transfer learning. *Information Processing & Management*, 56(5):1633–1644, 2019.

S. Zhou, J. Lin, L. Tan, and X. Liu. Condensed Convolution Neural Network by Attention over Self-attention for Stance Detection in Twitter. In *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN 2018)*. IEEE, 2019.

E. Zotova, R. Agerri, and G. Rigau. Semi-automatic Generation of Multilingual Datasets for Stance Detection in Twitter. *Expert Systems with Applications*, 114547, 2021.

A. Zubiaga, M. Liakata, R. Procter, K. Bontcheva, and P. Tolmie. Towards Detecting Rumours in Social Media. In *AAAI Workshop: AI for Cities*, 2015.

A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter. Detection and Resolution of Rumours in Social Media: A Survey. *ACM Computing Surveys (CSUR)*, 51(2):32, 2018.