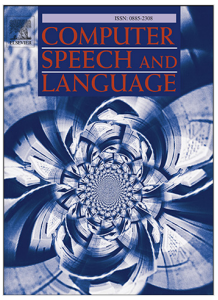
A Tag-based Methodology for the Detection of User Repair Strategies in Task-Oriented Conversational Agents

Journal Pre-proof



A tag-based methodology for the detection of user repair strategies in task-oriented conversational agents

Francesca Alloatti, Francesca Grasso, Roger Ferrod,
Giovanni Siragusa, Luigi Di Caro, Federica Cena

PII: S0885-2308(23)00122-5
DOI: https://doi.org/10.1016/j.csl.2023.101603
Reference: YCSLA 101603

To appear in: *Computer Speech & Language*

Received date : 29 November 2021
Revised date : 21 September 2023
Accepted date : 18 December 2023

Please cite this article as: F. Alloatti, F. Grasso, R. Ferrod et al., A tag-based methodology for the detection of user repair strategies in task-oriented conversational agents. *Computer Speech & Language* (2024), doi: https://doi.org/10.1016/j.csl.2023.101603.

A Tag-based Methodology for the Detection of User Repair Strategies in Task-Oriented Conversational Agents

Francesca Alloatti[a,b,*], Francesca Grasso, Roger Ferrod, Giovanni Siragusa, Luigi Di Caro, Federica Cena[b]

[a]*H-FARM Innovation. Via San Quintino 31, 10121 Turin, Italy*
[b]*Department of Computer Science, University of Turin. Corso Svizzera 185, 10149 Turin, Italy*

Abstract

Mutual comprehension is a crucial component that makes a conversation succeed. While it can be easily reached through the cooperation of the parties in human-human dialogues, such cooperation is often lacking in human-computer interaction due to technical problems, leading to broken conversations. Our goal is to work towards an effective detection of breakdowns in a conversation between humans and Conversational Agents (CA), as well as the different repair strategies users adopt when such communication problems occur. In this work, we propose a novel tag system designed to map and classify users' repair attempts while interacting with a CA. We subsequently present a set of Machine Learning models[1] trained to automatize the detection of such repair strategies. The tags are employed in a manual annotation exercise, performed on a publicly available dataset [2] of text-based task-oriented conversations. The batch of annotated data was then used to train the neural network-based classifiers. The analysis of the annotations provides interesting insights about users' behavior when dealing with breakdowns in a task-oriented dialogue system. The encouraging results obtained from neural models confirm the possibility of automatically recognizing occurrences of misunderstanding between users and CAs on the fly.

*Corresponding author
Email address: francesca.alloatti@h-farm.com (Francesca Alloatti)

[1] https://github.com/rogerferrod/boht

[2] The dataset is also available at https://github.com/rogerferrod/boht

Keywords: Conversational agents, repair strategies detection, human-centered artificial intelligence, human-computer interaction

1. Introduction

Conversational agents (CAs), such as chatbots and voicebots, represent a new frontier for human-computer interaction (HCI) since they allow people to interact with devices through natural language (Jurafsky and Martin, 2019). Instead of communicating with the machine via programming languages, or learning to use a graphic interface, users can speak or write to it in a free and more natural way (McTear, 2020). However, despite the latest advances in Artificial Intelligence, the conversational capabilities of machines are still unsatisfactory. CAs often fail at the *understanding* level, that is, when they are not capable of correctly parsing an input in natural language (Lee and Lee, 2021). This becomes particularly evident in chitchat settings, but it can be even more frustrating for users when interacting with task-oriented dialogue systems, whereas CAs should help humans attain a task (Seeger and Heinzl, 2021; Følstad and Brandtzaeg, 2020; Ashktorab et al., 2019).

While it is crucial to continue improving Natural Language Understanding capabilities of conversational systems, it is unrealistic to expect them to understand any human sentence at all times. In fact, humans themselves often have difficulties in understanding the language produced by other speakers. The mechanism that makes mutual comprehension possible is our capability to signal that error to the counterpart and to manifest the misunderstanding in order to initiate a repair of the conversation. In this sense, *repair strategies* are meant as sequences in a dialogue that aim at correcting some possible misunderstanding that may happen in that dialogue (Norman and Thomas, 1991; Sacks et al., 1974; Schegloff et al., 1977). In this contribution, we argue that one of the key factors that would make a dialogue system effective is its ability to detect and repair errors (Alloatti et al., 2021). Many existing works have already proved and highlighted the crucial role of repair expressions detection in improving human-agent interaction and, as a consequence, the overall users' experience (Li et al., 2020b; Schloss et al., 2022; Almansor et al., 2022; Følstad and Brandtzaeg, 2020) Our goal is therefore to transfer a common and spontaneous phenomenon of human communication to the field of human-computer interaction. We aim to instruct a CA to detect repair strategies put into place by users who interact with it. As

a consequence, the CA could react appropriately, improving the people's perception and assessment of its understanding capabilities(Cuadra et al., 2021).

In this paper we would like to answer the following research questions (RQ):

RQ1: Does the user show a certain behavior orientation when interacting with CAs and does he/she maintain it during the conversation?

RQ2: Facing a misunderstanding, how will users insist in the conversation in order to try and fix the problem?

RQ3: Is it possible to automatically identify a breakdown by using a state-of-the-art automatic classification system?

RQ4: Since repair strategies are not linguistically homogeneous and different subcategories may represent different communication intentions, is it possible to train a single classifier to distinguish among them?

We answered these questions by providing:

- an original tagset designed to mark different repair strategies employed by the user in the context of a task-oriented dialogue system. The tagset is used to manually annotate a dataset of conversations between users and a task-oriented CA in a real-life setting . The analysis of the annotated dataset allows us to answer RQ1 and RQ2, and thus (i) to map and classify all repair strategies put up by the users; (ii) to evaluate the performance of the dialogue system, since the presence of a repair strategy entails a communication problem; (iii) to map out the most problematic spots for the agent; and (iv) to gather insights about the users' reactions to those issues.

- several neural network-based classifiers that automatically recognize repair strategies in the dialogue; these classifiers could also be used to automatically tag and swift the annotation process up in future stages. We answered RQ3 and RQ4 by providing a comparison between a single classifier and multiple ones (one per repair strategy) on the above-mentioned dataset. Results obtained from the classification task suggest that one model per repair strategy performs better in recognizing the breakdowns, with outstanding results for the One-vs-Rest

classifier; for the single classifier, instead, we discovered that it is more suitable for subcategorizing the tags.

The paper is structured as follows. In the next section, we give a brief overview of existing works that deal with repair strategies studies and automatic dialogue classification. In section 3 we detail our classification methodology, based on the creation of a novel tagset and its application to a dataset, whereas in Section 4 we illustrate preliminary results from such methodology application, consisting of a manual annotation task. Section 5 describes the neural network classifiers and their parameters. We subsequently report their performance in recognizing the proposed tagset. Finally, in section 6 we discuss our main findings and future work.

2. Background and Related work

In this section, we provide an overview of the main related work in relation of the two main contributions of the paper: the repair strategies tagset and the automatic classification.

2.1. Repair strategies studies and tagsets

Repair strategies are not a fixed set of phenomena that can manifest themselves on any occasion, regardless of the speakers in the dialogue (Schegloff, 1992, 2007). Repair attempts are in fact contextual in nature. In the HCI field, the two parties in the dialogue are usually a machine and a human; the machine can take the shape of a robot (Lee et al., 2010), a multi-modal interface (Bourguet, 2006) or a back-and-forth dialogue system (Li et al., 2020a). Each of these embodiments of an agent implies the availability of certain strategies and the absence of others. For instance, some repair strategies are inherently tied to the vocal channel of communication (Beneteau et al., 2019; Myers et al., 2018; Porcheron et al., 2018): this is the case when the CA is not able to properly parse the user's speech. An error in the Automatic Speech Recognition module will then propagate, causing a breakdown in the conversation later on. Strategies can also be defined according to the main actor in the error-handling process (machine *versus* user), the purpose of the strategy (error prevention, discovery, or correction), and the use of different modalities of interaction (Bourguet, 2006). As regards these latter, the way people respond to problematic situations can outline different types of behavioral models (Ringberg et al., 2007). In this work, our focus

is on the responses adopted by human actors while they try to correct some miscommunication with a text-based task-oriented CA.

Several previous works attain the goal of identifying repair attempts in text-based CAs, although their focus differs from ours in various ways. For example, Moore and Arar list numerous repair strategies that can be found in HCI (Moore et al., 2018). Specifically, they analyze the different strategies according to the actor that signals the incomprehension. Even though the list provided is exhaustive, it is only partially applicable to our study, since our perspective focuses on the human party in the dialogue rather than on the machine's potential to activate a repair strategy. Moreover, the work does not offer any specific tagset to detect repair strategies implemented by the person.

Beneteau et al. identifies several strategies by analyzing interactions between families and the Amazon Alexa device (Beneteau et al., 2019). Some of those strategies can only be found when the CA is voice-based (e.g. *Prosodic changes* or *Increased volume*), while others may be applicable in text-based CAs too. For example, strategies such as *Repetition* were deemed to be also applicable to our case, and they were indeed found in our dataset. The *Repetition* strategy was also found in other work (Avdic and Vermeulen, 2020; Litman et al., 2006) as related to voice-based interaction such as with smart speakers. Avdic and Vermeulen also highlighted a "stop" strategy, equal to our *Closing* one(Avdic and Vermeulen, 2020).

Some interesting input can be found in Bourguet's taxonomy of errors (Bourguet, 2006). Our quadrant of interest is the *User correction* one, where we can find some strategies already highlighted by other articles: *repeat*, *rephrase*, and *spell out*, to name a few. A similar classification can also be found in Popescu-Belis' work (Popescu-Belis, 2008). We draw inspiration from his taxonomy to draw our own tagset, keeping existing tags as they were, where applicable. Allen and Core tackle the problem from another perspective and propose a manual for annotating dialogues (Allen and Core, 1997). Their tags mark important characteristics of utterances that indicate their role in the dialog and their relationship to each other. The authors themselves say they would expect that the annotation scheme would be refined for specific tasks, in order to provide further detail on phenomena of interest. We therefore took into account their work to model one of the levels of our tagset.

The work by Ashktorab and colleagues presents an orthogonal focus compared to ours: it outlines several repair strategies and tested various methods

to respond to them (Ashktorab et al., 2019). However, it focuses on preemptive repair attempts made by the CA when it notices a potential problem in its own way of presenting information, rather than reacting to a manifestation of incomprehension from the user. Even though their intuition is useful, we argue that repair strategies are by design something that can only be noticed by the counterpart in the dialogue. An utterance can be marked as a breakdown in the conversations not because it pertains to a specific class of erratic statements, or because it is ambiguous per se, but because it is interpreted as such by the other speaker of the conversation (Norman and Thomas, 1991).

Whilst the concept of repair strategy is not completely absent from the literature in HCI, none of the previous work aimed at classifying all the ways through which a *user* can try to repair an interaction with a written, task-oriented chatbot. Most of them provide a qualitative description of repair strategies that can be found in a dataset (Li et al., 2020a; Benotti and Blackburn, 2021), but they do not organize them in a complete and exhaustive tagset that can be used to automatically classify those strategies. For instance, several works (Popescu-Belis, 2008; Bourguet, 2006; Moore et al., 2018; Beneteau et al., 2019) state that *Repetition* is a strategy, but they do not provide unambiguous examples of what a repetition is, nor do they differentiate it from other repair strategies in terms of lexical features, behavioral implications, etc. Since our ultimate goal is to develop a system that will recognize repair attempts on its own, we created a novel tag system where each repair strategy is represented. Table 1 reports the aforementioned state-of-art approaches and their differences with respect to ours.

2.2. Automatic dialogues classification

Most Machine Learning methods are generally created for supervised automatic classification, thus separating data into sets (generally called classes, tags, or labels) after being fed with an already-labeled set of data (training set). Specifically, supervised classifiers use and continuously learn a set of weights that, combined with the current input, generates a candidate label. In case of error, i.e. when the model predicts the wrong class, the weights are changed in order to reconcile the input with the output label.

Machine Learning models are able to work with different kinds of data, such as images, audio, and texts. In our case, we focus on text-based state-of-the-art classification models, which usually leverage text-to-vector transformations. Generally speaking, words in texts are transformed into vec-

State-of-art work	Difference with respect to our approach
Moore et al. (Moore et al., 2018)	It analyses repair strategies according to the actor signaling them, but it does not provide a complete tag system. Moreover, our focus is solely on the repair attempts activated by the human actor.
Beneteau et al. (Beneteau et al., 2019)	It is mainly focused on voice-based repair strategies. However, the *Repetition* tag was deemed applicable also for our text-based CA.
Bourguet (Bourguet, 2006)	Some of its repair strategies (User correction) were applicable to our case, such as *Repetition*. They were integrated in our tagset via the top-down approach.
Popescu-Belis (Popescu-Belis, 2008)	Some of its repair strategies were applicable to our case, such as `RN` (i.e. *Negative answers*). They were integrated in our tagset via the top-down approach.
Ashktorab et al. (Ashktorab et al., 2019)	It focus on preemptive repair attempts made by the CA, while we argue that a repair strategy can only be brought to attention by the other speaker.
Allen and Core (Allen and Core, 1997)	Its tags mark important characteristics of utterances, but they are not tailored to the identification of repair strategies.
Avdic and Vermeulen (Avdic and Vermeulen, 2020)	It confirms the use of a "stop" strategy (i.e. *Closing*) in the interaction with smart speakers. We found that the same repair strategy can also be employed in text-based CAs.

Table 1: State-of-art approach on repair strategies and the difference with ours.

tors based on frequencies, co-occurrences, and latent/probabilistic semantic features extracted from the input document collection (Salton et al., 1975; Dumais et al., 1994; Hofmann, 1999; Blei et al., 2003), relying on the distributional hypothesis assumption (Harris, 1954). Modern approaches to vectorial transformation are instead currently based on neural networks, where input lexical features are encoded and passed through network layers to condense the original data while preserving the most relevant information (Mikolov et al., 2013a; Pennington et al., 2014; Bojanowski et al., 2017). Then, networks handling the sequential nature of texts have been proposed, capturing both order and syntactic structures behind natural language composition (Hochreiter and Schmidhuber, 1997; Gers et al., 2000; Cho et al., 2014). Finally, state-of-the-art network modeling of texts relies on transformers (Devlin et al., 2019), which generally adapt words vectorization based on their context of use or on specific parts of the input sequences through neural attention mechanisms (Bahdanau et al., 2015; Luong et al., 2015).

Since our data is composed of texts within dialogues, we also mention the literature on automatic detection or relations labeling among texts. For instance, Recognizing Textual Entailment (RTE) is the task of predicting the relation[3] of two sentences called hypothesis and premise. In the research field of RTE, the work of Bowman et al. (Bowman et al., 2015) paved the way for Neural Network models. In particular, the authors proposed both a dataset constructed through crowd-sourcing, and a Multi-Layer Perceptron (MLP) to classify the relation of the two proposed sentences. After this contribution, researchers started to experiment with Deep Learning models, also re-adapting ideas coming from different NLP fields such as Machine Translation. Rocktaschel et al. (Rocktäschel et al., 2015) proposed an encoder with attention for textual entailment, while (Tsuchiya, 2018; Gururangan et al., 2018; Poliak et al., 2018) found that, for some datasets, the hypothesis is all you need to identify the relation. According to them, it contains very salient information that can be used by a Neural Network to unravel the relation.

Focusing on dialogues, a very close task to ours is Dialogue Act Classification (DAC). The task requires that the chatbot identifies and classifies user utterances into different categories depending on the domain. This task uses the SWBD-DAMSEL tagset that contains 42 domains, e.g. Question, Command, Statement, Greeting, and so forth. Saha et al. (Saha et al., 2019)

[3]The labels are: *Entailment*, *Contradiction* and *Neutral*.

proposed a CNN-RNN model to extract the relevant features, followed by a CRF layer to recognize the label; they also tested the integration of other relevant features, such as bigrams, lemma form, PoS tags, question type words (What, When, Why, etc.), and greeting words (Thanks, Sorry, Forgive, etc.). Kato et al. (Kato et al., 2017), instead, proposed a recursive autoencoder (RAE) (Socher et al., 2011) to cope with the Japanese language. Finally, Saha et al. (Saha et al., 2020) merged the DAC task with the Emotion Recognition task in order to identify both the intent and the sentiment expressed by the user. In their work, they propose both a novel dataset called EMOTyDA, which contains texts, audios, and videos, and a Neural Network model with self, inter-modal, and inter-task attentions. This model takes in input an utterance (under the form of video, audio and text) and predicts both the DAC label and the emotion.

Another similar task is Dialogue State Tracking (DST). This task, also called Dialogue Belief Tracking, is crucial for dialogue systems. It infers the user's goals and intentions during the conversation with the agent. The extracted set of goal/intentions are then used to define the belief state, which the agent uses to select the next action according to its dialogue policy. The belief state is based on the ontology constructed in terms of slots and values (Ye et al., 2021; Dai et al., 2021; Chen et al., 2018; Ramadan et al., 2018), which represents the user goal and maintains a probability distribution over the possible user goals. Differently, our goal in this article is to identify the type of breakdown that occurs in the conversation and to automatically classify them according to the proposed tag structure. We thus decided to focus on the plain conversation, and not on the slot-value structure, in order to discover if the defined types of breakdowns are expressed in the text.

Our model aims to label utterances exploiting the linguistic features implicitly used by users. In such context, our task is similar to expertise modeling, which requires the adoption of statistical analysis in order to recognize knowledge units expressed in the conversation via bag of words, concepts, negation, and syntax features (Dascalu et al., 2008; McNamara et al., 2010).

In previous work, we proposed (Ferrod et al., 2021) a model for automatic detection of the user domain expertise from a conversation with a commercial chatbot. The method is based on a BiLSTM-CRF model (Huang et al., 2015) which processes each message in the conversation and identifies the expertise words, labeling them with a set of tags.

In this paper, we move away from short-text classification and expertise modeling in order to specifically recognize the repair strategies that emerge

within the interaction between users and machines. For this purpose, we rely on technologies similar to those described above, but we refer to a more complex task involving multiple messages within a conversation and the relationships between them. In this context, the label assignment could depend either on the single message or the entire conversation. We intended to pursue this goal by using only the linguistic features present in the texts. To the best of our knowledge, this is the first attempt towards an automatic recovery strategy detection in human-machine chatbot interactions.

3. Methodology for annotation

Given the absence of an available and complete framework in literature to specifically detect user repair strategies, we developed a new methodology that incorporates both a top-down approach and a bottom-up one. On the one hand, the top-down approach takes inspiration from the literature ((Moore et al., 2018; Popescu-Belis, 2008; Bourguet, 2006; Beneteau et al., 2019)), and aims at incorporating existing analyses of user behaviors in a comprehensive model. Through the bottom-up approach, on the other hand, we empirically analyzed actual dialogue sequences between a conversational agent and users in order to detect and map occurrences of user repair strategies. This ensures that each attempt made by a user is considered in our framework. Therefore, via the top-down approach, we took into account repair strategies that had already been outlined in previous work (Moore et al., 2018; Popescu-Belis, 2008; Bourguet, 2006; Beneteau et al., 2019), such as *Repetition* and *Rephrasing* (Beneteau et al., 2019), while through the bottom-up one, we were able to take into account also unaccounted repair attempts found in our dataset. The joining of the two approaches allowed us to elaborate a hierarchical framework that details the repair strategies a user may employ with a CA, while also grouping the strategies according to the behavioral model they entail. Specifically, the first and second levels of the taxonomy were mostly informed by the top-down approach, while the third one was nearly entirely created via the bottom-up methodology.

3.1. A new tagging system

Figure 1 illustrates the hierarchical structure of the tag system we developed.

The use of a taxonomy ensures that the tags with a specific entailment are grouped together under a master tag, since each tag entails different infer-

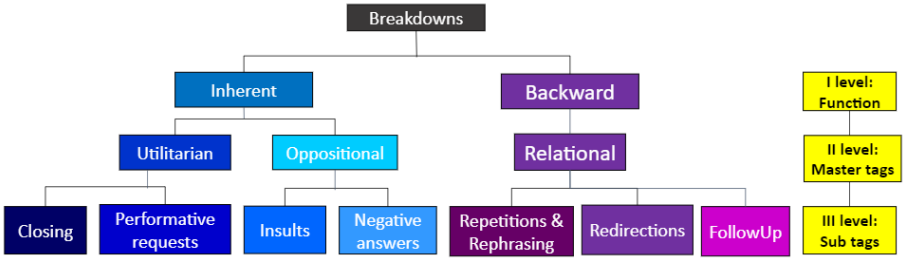


Figure 1: Taxonomy of the tagset for user repair strategies detection.

Backward Function	Utterances related to previous messages in the dialogue.
Inherent Function	Utterances related to the message itself, i.e. self-contained.

Table 2: First level of the tag taxonomy: utterance functions.

ences about the utterances it refers to - e.g. in terms of content, informative, and communicative function. A hierarchy of tags is also useful in the perspective of developing an automatic classifier: each level could be served by a different network architecture, according to the information it is encoding.

The first level of the taxonomy was drawn from the classification model contained in the work of Allen and Core (Allen and Core, 1997). They distinguish the function of an utterance, that is, the effect of a specific utterance on previous beliefs or future actions. We included the *Backward* function as part of our taxonomy, since a certain repair strategy may influence previous concepts stated in the dialogue, while the *Forward* one was not applicable: repair strategies shall always be triggered after a breakdown happens, therefore they will never be able to act upon actions in the future. Along with the Backward one, we elaborated the *Inherent* function as a result of the empirical data analysis we carried on the dataset. Inherent strategies do not refer to something previously said in the dialogue, as they do not depend on context. The identified functions are detailed in Table 2.

Our hierarchy of tags envisages a further sub-level: the *master tags*. In developing these labels, we looked at the user behavioral orientation classification of Ringberg et al. (Ringberg et al., 2007; Lee et al., 2010). The master tags thus address three peculiar *cultural models* to which users adhere via their behavioral orientation.

Utilitarian	The utilitarian model embraces a rational perspective. Users who employ this model do not perceive the chatbot failures as a personal attack nor view them as indicative of some kind of antagonism. Failures are indeed regarded as simple inconveniences; these users would make note of the failure and try to move on to some other system (e.g. a human operator) that could help them.
Oppositional	The oppositional cultural model evokes a consistently aggressive position whenever users experience a failure in the service they are using. CA users who employ this model tend to blame the system for the malfunction by expressing their discomfort via non-constructive criticism.
Relational	The relational cultural model is applied by users who express the desire to maintain emotional ties with the provider, even in the face of adverse events. In this context, people who employ relational strategies are those willing to change their own behavior to help the CA, rather than blaming the agent for its failure.

Table 3: Second level of the tag taxonomy: Master Tags.

We identified two master tags that pertain to the Inherent function: the *Utilitarian* and the *Oppositional* ones. These two do not require any specific context in order to occur. The *Relational* cultural model instead relates to the Backward function, since it may manifest itself within a dialog sequence. Table 3 describes the three master tags.

Finally, the sub-tags section was developed based on the empirical analysis and manual annotation of the dataset, via the bottom-up approach previously described. Each tag describes a repair strategy. While some of them can be found in previous works (e.g. *Repetition*), others represent an original contribution, as well as the labels themselves.

The Utilitarian model is expressed by the *Closing* and *Performative Requests* tags: these strategies entail a resolute and rational behavior. Next, the *Insults* and *Negative Answers* tags fall under the Oppositional master tag, since they mirror an adverse attitude. Finally, three tags - i.e. *Repe-*

titions&Rephrasings, *Redirections* and *FollowUp* - belong to the Relational cultural model, as they reflect a will to keep the conversation going. Table 4 illustrates the details of the tags and the types of repair expressions we identified through them.

3.2. The Dataset

In order to attain our goal, we needed a dataset of conversations where users would have some sort of problem with a CA, but also where they would have an interest in obtaining the right answer rather than give up. Such a situation would entail the presence of many repair strategies, because users would try to straighten the conversation whenever a breakdown occurs. We therefore resorted to a proprietary dataset of conversations between users and a CA, obtained from an Italian ICT company that deployed the CA on its platform[4]. The CA provides information on electronic invoicing to small businesses and freelancers. The conversations were gathered in a real-world setting, so that people who interacted with the system were actual users that had a real interest in obtaining a precise answer to their doubts. The interaction is stateless[5] and as such, it can be assimilated to a question-answering system (Alloatti et al., 2019; Bianchini et al., 2017). When the user asks a question, the CA computes the most appropriate answer. It can also express uncertainty if it is not confident enough, or provide the user with a set of options in case the uncertainty is between different answers. The CA is freely available to all users of the invoicing platform: for instance, during the year 2020, the CA delivered 817,000 single messages to 117,000 different users. Each user and their data is anonymized to preserve privacy.

Such a large public ensures a wide variety of demographics, linguistic competence, and technical expertise on the subject that reflects on the way users talk to the agent. Instead of conducting a restricted experiment, we preferred to waive the control over a specific group of users and, instead, to obtain as many real conversations as possible. In this way, we ascertain the veracity of the dialogues, and therefore the reliability of our evaluation of the users' behavior.

The raw dataset has 142,607 rows, grouped into 15,585 conversation sessions. A conversation session is an exchange between the CA and a specific

[4]The CA is deployed by the company for its authenticated users.

[5]The statelessness of the dataset does not affect the tagset and model employability in other contexts such as dialogue datasets and overall stateful systems.

Tag Name	Reference utterance	Examples
Closing	The user's intention is to close the conversation in a neutral way.	*Close the chat; Exit*
Performative Requests	Requests to talk to a human operator.	*I want to talk with a human; Is there a technician I could talk to*
Insults	Plain insults, including ironic or cruel comments.	*You're stupid; Yeah this was super useful...*
Negative Answers	The user is expressing a straightforward negation, without providing further detail.	*No; That's wrong*
Repetition & Rephrasings	Users repeat or rephrase their request to the CA, in the attempt of being understood.	First request: *I need to cancel an invoice*; second request: *How to erase invoices*
Redirections	The user changes topic abruptly when faced with a mistake.	First request: *I need to cancel an invoice*; second request: *Nevermind, tell me where I can find my password*
FollowUp	It contradicts or refers to a specific portion of what the CA said just previously.	The CA says: *You can find your new invoices at this link*; the user says: *Ok but what about the old ones?*

Table 4: Third level of the tag taxonomy: Sub-tags. All the examples of utterances has been translated for reader convenience, since the conversations are in Italian.

user over a certain number of rows. Each session contains at least 6 messages from the user; this quantity was deemed to be the minimum necessary to include a meaningful exchange between the two parties. Note that each row includes the timestamp of the corresponding single message. The rows in the dataset are sorted in chronological order. Filtering out the system APIs and other messages that do not belong to the users - and therefore were deemed not to be of interest for our purposes - we obtained 15,571 conversation sessions. Since each session contains multiple messages, the total number of potentially useful messages is 46,916.

3.3. The annotation procedure

The annotation was conducted by two expert annotators with formal training in linguistics and they are familiar with the internal functioning of CAs. They were also informed about the context in which the conversations were gathered, in order to have a better understanding of the real users' point of view. The procedure was carried out through two steps: first, every utterance entailing a user's repair intention was identified and thus marked as a repair attempt. An utterance coincides with a user's message, a single raw of the dataset. Each raw can thus either be marked as a repair strategy, or not. Secondly, each identified repair strategy was tagged according to the content carried by the statement. In other terms, the tag provides an abstract characterization of the content of the whole utterance, that is, what the user intended to express. Each repair strategy can only be associated with a single tag; subsequently, a tag can only be associated with an utterance marked as a repair strategy: those that are not, do not have a tag. For clarity purposes, Figure 2 reports examples of tagged exchanges.

The annotation was carried out in parallel by the two annotators. In order to proceed in a quicker manner, each annotator received half a portion of the dataset to analyze. This methodology allowed to obtain more annotated data in less time[6]. Besides, in order to assure accordance between the two experts, the team organized discussion sessions for problematic cases.

4. Analysis of the annotation

In this section, we describe the preliminary results obtained by analyzing an annotated sample of our dataset.

4.1. Results of the analysis

We started our analysis by computing the inter-rater agreement on a sample of 226 conversations (944 messages): the result is pretty high with Cohen's Kappa coefficient values equal to 0.80, or 0.84 generalizing to the first level tags.

As noted above, the rows in the dataset are listed in chronological order, consequently so is each subset. There have been no selection criteria for the

[6]Each annotator was able to tag a subset of the dataset during a time span of one month.

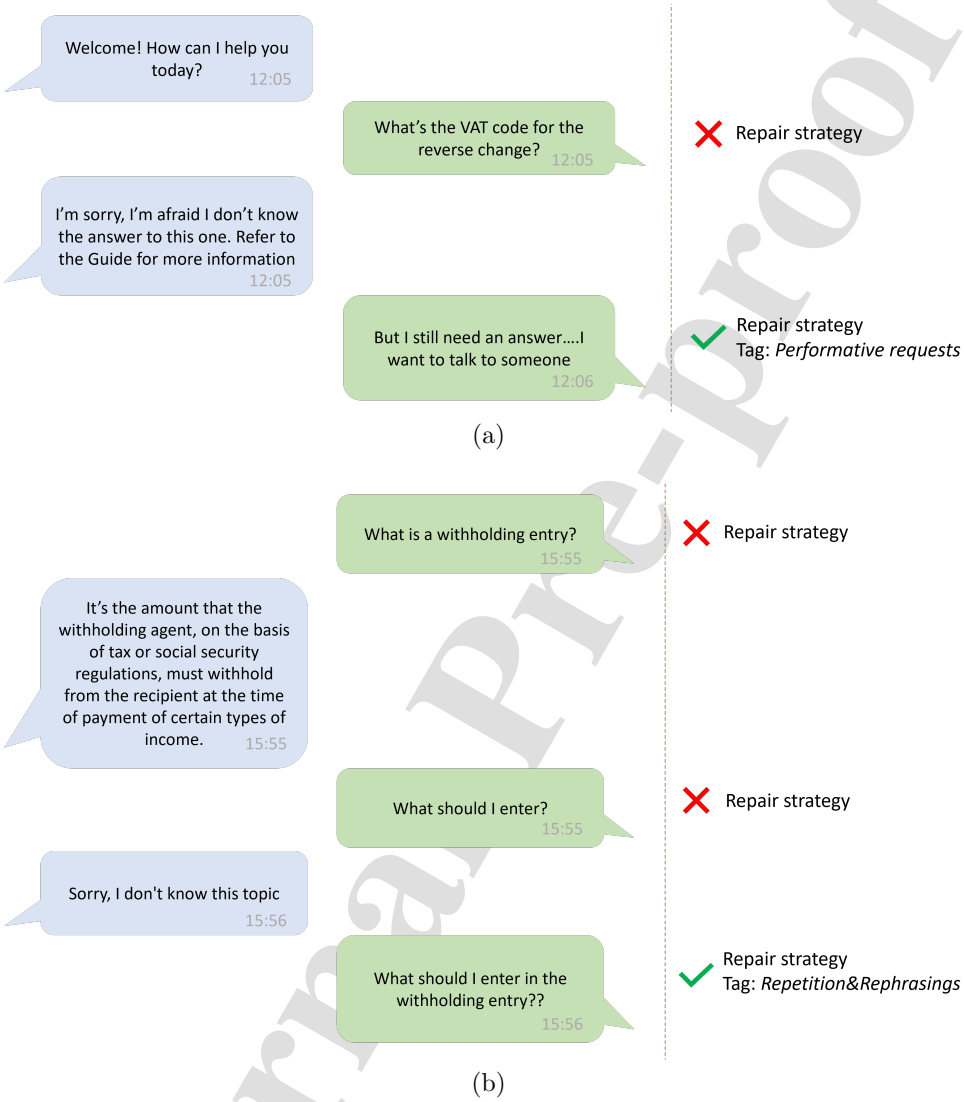


Figure 2: Examples of conversation sessions with repair strategies classifiable as (a) Performative Requests and (b) Repetition&Rephrasing.

subsets of conversation sessions: the annotation occurred from the beginning of each subset in sequential order. Considering that the sessions in the dataset are qualitatively randomized (given the uncontrolled environment of their occurrence, the anonymous character of the dialogues, and the heterogeneous nature of the thousands of users), there was no bias risk on the annotation

Function	Master Tags	Sub-Tags	Amount
Inherent	Utilitarian	Closing	324 (8.22%)
		Performative requests	114 (2.89%)
	Oppositional	Insults	106 (2.69%)
		Negative answers	133 (3.37%)
Backward	Relational	Repetitions & Rephrasings	2681 (67.99%)
		Redirections	144 (3.65%)
		FollowUp	441 (11.18%)
Total number of breakdowns: **3943**			

Table 5: Distribution of tags. The percentage refers to the total number of utterances marked as a repair strategy.

exercise, meaning that the eventually annotated sample can be qualified as representative.

Preliminary results concern 2175 annotated conversation sessions. All these sessions amount to 7585 messages. Of these, 3943 (52%) contain misunderstandings. The presence or absence of misunderstanding is almost balanced (48% absence, 52% presence). However, the categorization of misunderstandings is quite varied. The Backward function covers 43.06% of the breakdowns (equal to 3266 turns) while the Inherent function only 8.92% (equal to 677 turns).

Table 5 reports the distribution of the tags on the total number of utterances that were tagged as repairs.

4.2. Discussion

The most used tag is Repetitions&Rephrasings, which is part of the Relational cultural model. It means that most users prefer to try and repair the conversation with the CA by assuming responsibility for the mistake, rather than blaming the agent. The second and fourth most frequent tags are also part of the same master tag, thus confirming the hypothesis that in the context of task-oriented conversational agents, users have a paramount interest in solving their doubts. They are therefore particularly inclined to attempt to repair the exchange by changing their own behaviour: this first finding answers the second research question (RQ2).

The Closing tag, however, is also quite frequent but pertains to a different master. This tag is used by users who want to close the conversation, either because they already obtained the answer they were seeking, or because they

no longer want to chat. Its ubiquity therefore partially explains its frequency; in order to fully understand its role, it is also useful to look at the possible correlation between different tags.

Figure 2a reports the normalized co-occurrence matrix of tags. Two tags co-occur if they appear at least once in the same conversation. We normalized it via cosine distance: first, we calculated the cosine similarity between each pair of tags; then, we took the inverse in order to obtain the distance. The resulting matrix is a dissimilarity matrix, where higher values correspond to distant (i.e., potentially decorrelated) tags. The symmetric distance matrix can be displayed graphically by applying Multidimensional Scaling (MSD) algorithm, as shown in Figure 2b. We used the non-metric MDS since the matrix does not report real distances, but rather a similarity measure. In this way, we preserved the ordering imposed by the cosine distance, obtaining good results with relatively low distortion (0.0215 stress value).

The intersection point of the NEG (Negative answers) and INS (Insults) tags show a low dissimilarity value. This means that, despite being less frequent overall, the two tags are often found together in the same conversation sessions. This data confirms the profile of Oppositional users, who will employ those two tags jointly in their dialogues while generally avoiding a rational or empathetic cultural model. Similarly, the R&R (Repetitions&Rephrasings) tag especially correlates with the FWU (FollowUp) one as well as with the RED (Redirections) tag. The former combination describes the situation where users are not convinced by the answer the chatbot is giving and will therefore try to clarify their requests. They do this both by rephrasing or repeating their requests, and by focusing their rebuttal on a specific element of the chatbot's answer, in the hope of eliciting a more specific explanation. The R&R and RED combination depicts a similar scenario, where users opt to change the topic of their request directly once the repetition attempt proved to be ineffective. All R&R, FWU, and RED pertain to the Relational master tag, once again confirming the thesis that users usually belong to one of these three cultural models and will often employ consistent tags, rather than shifting from one model to another. In relation to the first research question (RQ1), it is thus safe to assume that the use of certain repair strategies highlights user behavioral orientation towards a CA.

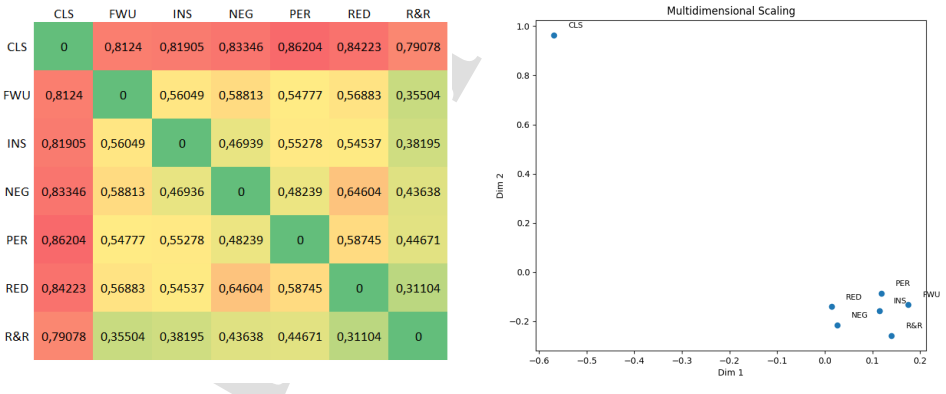
As before, the Closing tag appears to have a peculiar distribution compared to the other tags. It correlates very little with the other tags and, compared to other repair strategies, it has fewer conversation sessions in common with R&R. This apparent discrepancy can be explained by analysing how

CLS	FWU	INS	NEG	PER	RED	R&R
166 (70%)	111 (28%)	14 (13%)	28 (19%)	18 (19%)	14 (10%)	1085 (68%)

Table 6: Absolute number of conversation sessions in which only one type of tag occurs. The percentage (in parentheses) refers to the total amount of conversation sessions in which that tag appears by itself. Note that the CLS (Closing) tag is quite frequent, but correlates very little with other repair strategies: in 70% of the cases, it is indeed found alone in the conversations.

many conversation sessions only contain one type of tag (Table 6). Specifically, the Closing tag appears by itself in a high number of sessions. This is consistent with the fact that this tag does not correlate much with other repair strategies. Users who employ this strategy are thus primarily interested in closing the conversation and are not willing to try any other strategy to repair the conversation: once again, this description fits the master tag of the Closing strategy, i.e. Utilitarian behavior.

Finally, each tag features a high correlation with the R&R tag. This occurs basically because, as we could notice, this specific tag is the most used one overall.



(a) (b)

Figure 3: (a) The matrix shows the correlation between two different tags. The numbers indicate the level of dissimilarity (i.e. scarce co-occurrence) between the two tags. (b) The figure shows the graphic display of the symmetric distance matrix.

5. Automatic classification of Repair Strategies

As described in the Introduction, our goal is to create a classifier that is able to detect the breakdowns in a conversation. The manual annotation performed in Section 3 provided the data to train our model. For the scope of our analysis, we focused on the first level of the tagset, which comprises the macro categories *Inherent* (i.e., breakdowns, such as Insults, that require a message-oriented analysis) and *Backward* (i.e., those messages that need to be contextualized within the conversation)[7].

Since the repair strategies are not linguistically homogeneous and different subcategories represent different communication intentions, we needed to use two different computational approaches:

- for the Inherent classes, it was necessary to focus on the vocabulary and relations between words, i.e. the classifier has to predict a tag for each sentence in the conversation;
- for the Backward classes, it needed to search for recurring patterns within a larger context, involving either the previous sentence or the entire conversation. In this latter case, the classifier has to predict a tag for the entire conversation (or a set of sentences).

We thus defined three main classifier sets:

Inherent Repair Strategy VS Rest: the classifiers in this set recognize whether the analyzed sentence contains an Inherent breakdown. We will refer to it as *INH VS Rest* from now on;

Backward Repair Strategy VS Rest: the models recognize whether the conversation (or a group of sentences) contains a Backward breakdown. *BCK VS Rest* from now on;

Absence of Misunderstandings: we are not only interested to find whether a sentence (or conversation) contains an inherent or backward breakdown, but also its absence. In this way, we can first analyze if the conversation contains a breakdown and then its type, speeding the tagging process up. We will refer to this set of classifiers as *POS VS Rest* from now on.

[7]We excluded the FWU tag from the Backward category as it is the only one semantically linked to the CA's responses and it is not limited to the user's interaction.

In order to be able to determine the presence of a breakdown, we have to process each sentence of the conversation. We decided to create a single classifier (under the name of *Single Model*) that merges all three previous sets at the cost of sacrificing its performance due to the heterogeneity of the repair strategies. It is worth remembering that our final goal would be to embed this repair strategies detection capability in the CA itself, in order for it to immediately spot problems in the dialogue and act upon them, no matter the kind of strategy the user has employed.

Each classifier is defined as a neural network model that uses a word-embedding (Mikolov et al., 2013b; Pennington et al., 2014) defined via pre-trained Italian BERT model[8] (Devlin et al., 2019) which has 768 units.

INH VS Rest. In the INH vs Rest classification, the core of the classifier transforms the input sentences into a unique vector that semantically represents the entire message; the resulting vector is then fed in input to a MultiLayer-Perceptron - also known as MLP - (2 layers with ReLU activation function) in order to obtain the tag distribution. For the core, we experimented with both BERT's CLS tag vector, which represents the input sentence, and the simple word embedding of the sentences; over this latter layer, we tested a Convolutional Neural Network[9] (CNN), a Bidirectional Long Short Term Memory (BiLSTM) network[10] (Hochreiter and Schmidhuber, 1997) and a multi-head attention model (ATT) (Vaswani et al., 2017).

BCK VS Rest. Differently from the INH vs Rest classification, the BCK vs Rest classifier requires a memory of the previously encountered messages, since the breakdown could occur from the interaction between the current sentence and a previous one. Therefore, we used a hierarchical neural network to solve this task, i.e. a neural network composed of two layers: the first layer transforms the input sentence into a single vector representation, while the second layer relates the current sentence with its previous context. For simplicity, we will call the first layer *sentence-level* and the second layer *conversation-level* from now on. In this task, we used the same models of the INH vs Rest classification for the sentence-level; the conversation-level,

[8]https://huggingface.co/dbmdz/bert-base-italian-cased

[9]Following the model proposed by Kim et al. (Kim, 2014).

[10]The BiLSTM reads the input sentences from left-to-right and from right-to-left, obtaining a more accurate representation compared to a unidirectional one.

instead, is based on an LSTM network. In this latter case, we did not use a bidirectional one since future messages are not relevant for the identification and classification of breakdowns.

POS VS Rest. The task of POS vs Rest classification is similar to BCK vs Rest because we have to analyze both the single sentences and their context (previous sentences) to determine if there is an absence of misunderstanding or not. For this task, we tested the BERT's CLS tag, the BiLSTM model, and the multi-head attention model for the sentence-level.

Single Model. The single model is an end-to-end model that identifies the breakdowns in the conversation and classifies them. Our goal is to determine if it is possible to discriminate the 3 classes in a single lecture of the conversation. The model is a hierarchical neural network composed of a sentence-level layer and a conversation layer. For the sentence-level layer, we tested the same neural network models of BCK vs Rest. The main difference, with respect to this latter one, is that it can predict each of the three tags POS, BCK, or INH.

5.1. Implementation

We trained the classifiers using the weighted log negative loss given the imbalance of the dataset:

$$loss(x, y) = -w_y \ln P(y|x; \theta) \tag{1}$$

where the input x and y represent, respectively, the network results and the true class prediction, while w_y is the weight associated with y class; the probability of obtaining the correct label given the output of the network is parameterized by θ (the parameters of the model). The training process takes place by minimizing the weighted average of the loss on the number of messages of the conversations and on all the mini-batches. Weights are calculated on the basis of label frequencies. We used Adam (Kingma and Ba, 2014) as optimizer.

In order to regularize the network, and to increase the generalization strength, we applied dropout layers (Srivastava et al., 2014) and weight decay (Loshchilov and Hutter, 2018). The probability of dropout is fixed at 0.2 (keeping the probability of 0.8), while weight decay can assume values in

the range [0, 1e-5] according to the model (more details in Section 5.2). For multi-head attention models, we set the number of heads to 2 and we used positional encoding. Finally, we set the gradient clipping to 3 in order to avoid the vanishing/exploding gradient issue.

5.2. Evaluation

For the evaluation, we split the dataset into training, evaluation, and test sets. We used the evaluation set for the early stopping and hyper-parameters tuning. Table 7 reports the class distribution in the three sets.

Name	POS	INH	BCK
Entire dataset	53.83	8.93	37.24
Training set	54.21	8.68	37.14
Evaluation set	53.87	7.75	38.38
Test set	52.01	11.38	36.6

Table 7: The table reports the tag distributions (percentage) for the training, evaluation, and test sets.

We trained the models on an Nvidia RTX2080Ti until the evaluation loss did not improve substantially for 5 consecutive epochs. The source code is available at `https://github.com/rogerferrod/boht`.

5.2.1. Quantitative Results

In this section, we present the best results obtained during the experiments. In particular, Table 8 shows the performances of the different architectures developed for the one-vs-rest classifier.

Looking more closely at the best models, it can be noticed that different tasks correspond to very different architectures. For example, to identify *Insults*, *Closing*, or other repair strategies that are strongly oriented to the vocabulary (INH class), the CNN architecture provides the best performance; at the same time, however, the convolution applied to the classification of the other two classes does not allow the network to converge. Vice versa, the recurring model, which performs well for the detection of BCK and POS, does not seem to be suitable for the detection of INH. These results, therefore, confirm the assumptions made about the different semantic nature of tags.

We also experimented with a CNN + LSTM model for both the BCK and POS tasks, but it did not converge; we believe that CNN module is not

	Model	F1 Score	Precision	Recall
BCK	CLS + LSTM	52.66	53.52	51.82
	BiLSTM + LSTM	**65.47**	**77.64**	**56.59**
	ATT + LSTM	62.59	75.88	53.26
INH	CNN + MLP	**58.99**	**68.91**	**51.57**
	CLS + MLP	57.14	48.74	69.05
	BiLSTM + MLP	24.67	15.97	54.28
POS	CLS + LSTM	56.51	54.54	58.63
	BiLSTM + LSTM	**71.27**	**78.92**	**64.97**
	ATT + LSTM	63.35	68.08	64.70

Table 8: The table reports the Precision, Recall, and F-measure of the proposed classifiers. The best results are in bold.

able to create a vector representation of the sentence. For the INH task, we tested the ATT + MLP; as for the CNN + LSTM model, it did not converge.

By combining the best models together we can build a multiclass classifier, following the one-vs-rest approach, whose results are shown in Table 9. These results validate the third research question (RQ3), since we demonstrated it is possible to train either several classifiers or a multiclass one to recognize the repair strategy in a conversation (or sentence).

Model		POS	INH	BCK	AVG
One vs Rest	F1	69.58	56.52	67.70	**64.62**
	P	77.21	58.03	60.43	**65.23**
	R	63.32	55.08	77.13	**65.18**

Table 9: The table reports the results of the multiclass classifier. The best results are in bold.

Finally, we compared the obtained results with those of the *Single Model*, developed with the aim of distinguishing the three classes in a single step. By comparing the scores reported in Table 10 with those previously shown, it is possible to notice a decrease in the performance. This phenomenon is largely due to the different accuracy on the INH class; indeed, the scores of the other two classes do not differ much from the corresponding values in Table 9. This last experiment demonstrates how difficult is to recognize INH, BCK and POS with a single model, proving the diverse nature of the tags also from a computational point of view. We thus positively answered

the fourth research question (RQ4), showing that it is possible to create a single neural network model to recognize the three classes at the cost of sacrificing the performances. Table 11 shows the optimized hyperparameters of all previously presented best models.

Model		POS	INH	BCK	AVG
BiLSTM + LSTM	F1	69.79	36.36	64.95	**57.03**
	P	63.04	36.65	72.86	**58.19**
	R	78.16	34.33	58.59	**57.02**
CLS + LSTM	F1	59.04	54.69	46.11	53.28
	P	59.24	56.30	45.48	53.67
	R	58.83	53.17	46.77	52.93
ATT + ATT	F1	71.96	23.45	69.41	54.94
	P	56.30	14.28	99.50	56.69
	R	99.69	65.38	53.30	72.79
ATT + LSTM	F1	66.55	13.51	60.94	47.00
	P	63.90	8.40	71.36	47.89
	R	69.42	34.48	53.18	52.36

Table 10: The table reports the results of the tested architectures for the Single Model. The best results are in bold.

Model	Parameters
BCK vs Rest	1° layer 192 (x2)
	2°layer 150
INH vs Rest	1° layer 150
	2°layer 512
	3° layer 300
POS vs Rest	1° layer 192 (x2)
	2°layer 300
	Weight decay 1e-5
Single model	1° layer 384 (x2)
	2°layer 192

Table 11: Hyperparameters for chosen models; where not indicated, the following default values are considered applied: learning rate 0.001, dropout 0.2, weight decay 0.

To further facilitate the comparison between the two proposed models, we report in Table 12 the results obtained by both the multiclass classifier

and the Single Model classifier on the binary case (i.e. identify the presence of any kind of misunderstanding). Although both classifiers are very close on the F1 score, their Precision and Recall ones diverge. In particular, the One-vs-Rest approach shows a very high precision (79.07) in identifying the presence of a misunderstanding in the conversation, reducing the number of false positives (only 108 cases); on the other hand, the Single Model classifier has a lower Precision (about 15 points lower) that leads to 215 false positives, but it is able to recognize a large variety of misunderstandings, compared to the One-vs-Rest, thanks to the high Recall (80.27).

We believe that the One-vs-Rest classifier is better at the task of recognizing the breakdowns expressed by the user during the conversation, allowing the CA to create an ad-hoc reply; the Single Model, instead, could be used to subcategorize the tags given its high Recall.

Model	F1	Precision	Recall
One vs Rest	72.83	79.07	65.81
Single Model	72.43	65.98	80.27

Table 12: Comparison of the One-vs-Rest model and the Single Model on the binary task (i.e., identifying the presence of any kind of misunderstanding).

5.2.2. Qualitative Analysis

In order to better interpret the performance results of the model, and to eventually figure a way to enhance it in future work, we took a closer look at the instances of disagreement between the manual annotation and the one performed by the model. The aim was to possibly find some reasonable explanation for the failed agreement, or to check whether they occur in recognizable patterns. We thus conducted a manual check and analysis on 111 disagreements between the manual tagging exercise and the automatic one[11]. The analysis led us to hypothesize a reasonable explanation for 39 cases of disagreement. These could be occurrences of understandable misinterpretations of the given message by the classifier (semantic level), a lack of understanding of the statement on the contextual (i.e. pragmatic) level, or trouble with specific meaningful linguistic units. More in detail:

[11]Over a random sample of 347 messages.

- 20 instances of understandable disagreements concern cases of "false BCK". We notice that the repetition of lexical units (mostly a single word or a syntagm) between adjacent messages will lead the classifier to identify the given message as an instance of Repetition&Rephrasing, and thus to mark it as BCK.

- Similarly, 5 occurrences of "false INH" involve messages containing words such as "close" or "exit", thus leading the classifier to eventually recognize them as *Closing* attempts, although the communicative context shows that they are actually not.

- There are 3 more instances of disagreement identifiable as being semantic in nature. In these cases, the classifier shows inherent limits in comprehension at the lexical-semantic (i.e. word) level. This is the case, for example, of technical terminology, such as "401 (k) Plan", or of greeting formulas used as a farewell (and therefore to be interpreted as *Closing* expressions).

- In 8 cases the classifier presents problems in correctly detecting and classifying statements that begin with a negation (*neg*), showing a lack of competence in understanding, e.g., a construct of the type *neg + verb*.

- Finally, we identify 3 occurrences of "missing INH" (incorrectly classified as POS) which address the scarce pragmatic competence of the model. This is the case, for example, of sarcastic or idiomatic expressions equivalent to *Insults*. In both cases, the literal meaning has no hostile sense and could be overlooked by the classifier.

Even if the adaptive response generation is beyond the scope of the current work, we list below some conversations, taken from the dataset, to better visualize the utility of predicting the correct repair strategy and answering accordingly. For each breakdown detected in the interaction between the conversational agent (CA) and the user, we report the corresponding repair strategy and a hypothetical response of the agent.

CA Welcome! How can I help you today?

User What's the VAT code for the reverse change?

CA I'm sorry, I'm afraid I don't know the answer to this one. Refer to the Guide for more information

User But I still need an answer... I want to talk to someone

Repair strategy (Performative requests)

CA I apologize for the inconvenience. I'll connect you to a representative for assistance.

CA Welcome! How can I help you today?

User Change the bank

CA Choose one of these options: i) How do I create a bank on the Invoice platform? ii) How do i create a bank in the CashIn platform?

User Replace bank

Repair strategy (Repetitions & Rephrasing)

CA I understand you want to replace a bank. Please provide me with more details or choose: i) Replace bank on the Invoice platform ii) Replace bank on the CashIn platform.

CA Welcome! How can I help you today?

User Who are you?

CA I'm sorry, I'm afraid I did not understand.

User Can I assign the deduction?

Repair strategy (Redirections)

CA Yes, you can assign deductions. Would you like guidance on enabling deduction tax management for issuing an invoice to a flat complex?

User Invoices with fiscal deduction

Repair strategy (Repetitions & Rephrasing)

CA To clarify, are you inquiring about how to set up invoices with fiscal deductions or seeking more details on the process?

6. Conclusion and Future Work

In this paper, we presented a new methodology to comprehensively map and classify repair strategies users adopt whenever they run into some fallacy within a conversation with a CA. The methodology has been applied to a publicly available dataset of dialogues from a task-oriented CA. We developed an original tagset designed to mark each repair strategy we identified and used it to manually tag a portion of the dataset. We then trained different supervised neural network models using the annotated portion of the dataset. They show good results in recognizing the expressed recovery strategy.

Through our work, we were able to answer four different research questions. The analysis of the manual annotation highlighted consistent correlations between the tags that pertain to the same master level, thus confirming that users have a certain behavioral orientation when facing a mistake with a CA (RQ1). Specifically, the most frequent tag is Repetitions&Rephrasings, which is part of the Relational cultural model. Its abundance entails the fact that most users prefer to try to fix the problem by changing their own phrasing, rather than simply giving up. Since the second and fourth most frequent tags are also part of the same master tag, we can now affirm that facing a misunderstanding with a task-oriented CA, most users will employ a Relational behavior (RQ2).

Our final goal would be to endow a CA with the ability to spot and categorize repair strategies automatically. We thus developed several neural network classifiers in order to recognize the Inherent classes (INH), Backward classes (BCK) and absence of misunderstanding (POS). From the evaluation, we discovered that BiLSTM + LSTM is the best model to recognize BCK and POS, while the CNN + MLP has the best Precision (and F1) score in recognizing the INH tag; these results validate the third research question (RQ3). We also confirmed RQ4 by showing that it is possible to train a single classifier to recognize the three tags. However, the resulting classifier performs slightly worse compared to the previous ones, especially in recognizing the INH tag, since it requires different features with respect to BCK and POS.

Finally, the analysis of the disagreements between the manual annotation and the automatic one has allowed us to get some precious hints on the nature of the model's performance and, as a consequence, on where to possibly direct future work. The analysis revealed that the model could be enhanced with more complex semantic, pragmatic, and morphosyntactic competence,

for example by drawing on common knowledge resources or domain-specific vocabulary.

Our study can be seen as an example of Human-Centered Artificial Intelligence (HCAI) approach (Shneiderman, 2021), which combine Artificial Intelligence (AI) algorithms with human-centered thinking. HCAI combines research on AI algorithms with user experience design methods.

The contributions of our work can be summarized as follows:

1. We created a novel tagset that can be applied to many other human-machine dialogue contexts. It can also constitute a starting point for further fine-grained analyses.
2. Through the detection of user repair strategies, we are able to identify and classify the most common mistakes the CA makes, thus suggesting a new way through which a CA can be evaluated. An abundance of repair strategies entails low performance.
3. Results show that through our tagging exercise, different types of users can be distinguished. Thus, a range of appropriate responses can be implemented depending on the type of user detected.
4. The findings can lead to a greater and deeper understanding of the user's perceptions of the agent in terms of expectations, reactions, and eventually their mental model. This can be revealing from a user studies perspective.
5. An evaluation of several classifiers to automatically recognize the repair strategies, from which we discovered that each I level (function) tag requires its own classifier.

In future work, we plan to expand the batch of manually annotated data, in order to feed our models with more examples and thus improve their performance. Moreover, we plan to use the information obtained by the given repair strategy to produce different answers: users that consistently employ Repetitions across all their sessions might benefit from a more detailed explanation by the CA on why it is not able to answer appropriately. On the other hand, users who insult the CA would probably find a detailed explanation annoying and useless. Our final aim is to equip a CA with all the tools to detect and repair any possible breakdown that may happen in a dialogue, as a human being would do.

References

Allen, J., Core, M., 1997. Draft of DAMSL: Dialog act markup in several layers. Unpublished manuscript.

Alloatti, F., Di Caro, L., Bosca, A., 2021. Conversation analysis, repair sequences and human computer interaction, in: Proceedings of the AAAI Conference on Artificial Intelligence, Fourth Workshop on Reasoning and Learning for Human-Machine Dialogues, pp. 1–4.

Alloatti, F., Di Caro, L., Sportelli, G., 2019. Real life application of a question answering system using BERT language model, in: Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, Association for Computational Linguistics, Stockholm, Sweden. pp. 250–253.

Almansor, E.H., Hussain, F.K., Hussain, O.K., 2022. Measuring chatbot quality of service to predict human-machine hand-over using a character deep learning model. International Journal of Web and Grid Services 18, 479–495.

Ashktorab, Z., Jain, M., Liao, Q.V., Weisz, J.D., 2019. Resilient Chatbots: Repair Strategy Preferences for Conversational Breakdowns. Association for Computing Machinery, New York, NY, USA. p. 1–12.

Avdic, M., Vermeulen, J., 2020. Intelligibility issues faced by smart speaker enthusiasts in understanding what their devices do and why, in: 32nd Australian Conference on Human-Computer Interaction, Association for Computing Machinery, New York, NY, USA. p. 314–328.

Bahdanau, D., Cho, K.H., Bengio, Y., 2015. Neural machine translation by jointly learning to align and translate, in: 3rd International Conference on Learning Representations, ICLR 2015.

Beneteau, E., Richards, O.K., Zhang, M., Kientz, J.A., Yip, J., Hiniker, A., 2019. Communication breakdowns between families and alexa, in: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA. p. 1–13.

Benotti, L., Blackburn, P., 2021. A recipe for annotating grounded clarifications, in: Proceedings of the 2021 Conference of the North American

Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4065–4077.

Bianchini, A., Tarasconi, F., Ventaglio, R., Guadalupi, M., 2017. "gimme the usual" - how handling of pragmatics improves chatbots. CLiC-it , 30.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. Journal of machine Learning research 3, 993–1022.

Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics 5, 135–146.

Bourguet, M.L., 2006. Towards a taxonomy of error-handling strategies in recognition-based multi-modal human–computer interfaces. Signal Processing 86, 3625–3643. Special Section: Multimodal Human-Computer Interfaces.

Bowman, S.R., Angeli, G., Potts, C., Manning, C.D., 2015. A large annotated corpus for learning natural language inference, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Chen, W., Chen, J., Su, Y., Wang, X., Yu, D., Yan, X., Wang, W.Y., 2018. Xl-nbt: A cross-lingual neural belief tracking framework, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 414–424.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734.

Cuadra, A., Li, S., Lee, H., Cho, J., Ju, W., 2021. My bad! repairing intelligent voice assistant errors improves interaction, in: Proceedings of the ACM on Human-Computer Interaction, ACM New York, NY, USA. pp. 1–24.

Dai, Y., Li, H., Li, Y., Sun, J., Huang, F., Si, L., Zhu, X., 2021. Preview, attend and review: Schema-aware curriculum learning for multi-domain

dialogue state tracking, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pp. 879–885.

Dascalu, M., Chioasca, E.V., Trausan-Matu, S., 2008. Asap-an advanced system for assessing chat participants, in: International Conference on Artificial Intelligence: Methodology, Systems, and Applications, Springer. pp. 58–68.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2019. Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186.

Dumais, S., et al., 1994. Latent semantic indexing (lsi) and trec-2. Nist Special Publication Sp , 105–105.

Ferrod, R., Cena, F., Di Caro, L., Mana, D., Simeoni, R.G., 2021. Identifying users' domain expertise from dialogues, in: Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, pp. 29–34.

Følstad, A., Brandtzaeg, P.B., 2020. Users' experiences with chatbots: findings from a questionnaire study. Quality and User Experience 5, 1–14.

Gers, F.A., Schmidhuber, J., Cummins, F., 2000. Learning to forget: Continual prediction with lstm. Neural computation 12, 2451–2471.

Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S.R., Smith, N.A., 2018. Annotation artifacts in natural language inference data, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 107–112.

Harris, Z.S., 1954. Distributional structure. Word 10, 146–162.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9, 1735–1780.

Hofmann, T., 1999. Probabilistic latent semantic indexing, in: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 50–57.

Huang, Z., Xu, W., Yu, K., 2015. Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 .

Jurafsky, D., Martin, J.H., 2019. Speech and language processing - an introduction to natural language processing,computational linguistics, and speech recognition (third draft).

Kato, T., Nagai, A., Noda, N., Sumitomo, R., Wu, J., Yamamoto, S., 2017. Utterance intent classification of a spoken dialogue system with efficiently untied recursive autoencoders, in: Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, pp. 60–64.

Kim, Y., 2014. Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar. pp. 1746–1751.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Lee, M., Lee, S., 2021. "i don't know exactly but i know a little": Exploring better responses of conversational agents with insufficient information, in: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA.

Lee, M.K., Kiesler, S., Forlizzi, J., Srinivasa, S., Rybski, P., 2010. Gracefully mitigating breakdowns in robotic services, in: 2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp. 203–210.

Li, C.H., Yeh, S.F., Chang, T.J., Tsai, M.H., Chen, K., Chang, Y.J., 2020a. A conversation analysis of non-progress and coping strategies with a banking task-oriented chatbot, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA. p. 1–12.

Li, T.J.J., Chen, J., Xia, H., Mitchell, T.M., Myers, B.A., 2020b. Multimodal repairs of conversational breakdowns in task-oriented dialogs, in: Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, Association for Computing Machinery, New York, NY, USA. p. 1094–1107.

Litman, D., Swerts, M., Hirschberg, J., 2006. Characterizing and predicting corrections in spoken dialogue systems. Computational Linguistics 32, 417–438.

Loshchilov, I., Hutter, F., 2018. Decoupled weight decay regularization, in: International Conference on Learning Representations.

Luong, M.T., Pham, H., Manning, C.D., 2015. Effective approaches to attention-based neural machine translation, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1412–1421.

McNamara, D.S., Crossley, S.A., McCarthy, P.M., 2010. Linguistic features of writing quality. Written communication 27, 57–86.

McTear, M., 2020. Conversational ai: Dialogue systems, conversational agents, and chatbots. Synthesis Lectures on Human Language Technologies 13, 1–251.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013a. Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, pp. 3111–3119.

Mikolov, T., Yih, W.t., Zweig, G., 2013b. Linguistic regularities in continuous space word representations, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746–751.

Moore, R., Szymanski, M., Arar, R., Ren, G., 2018. Studies in Conversational UX Design. Human–Computer Interaction Series, Springer International Publishing.

Myers, C., Furqan, A., Nebolsky, J., Caro, K., Zhu, J., 2018. Patterns for how users overcome obstacles in voice user interfaces, Association for Computing Machinery, New York, NY, USA. p. 1–7.

Norman, M., Thomas, P., 1991. Informing HCI design through conversation analysis. International Journal of Man-Machine Studies 35, 235–250.

Pennington, J., Socher, R., Manning, C., 2014. Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543.

Poliak, A., Naradowsky, J., Haldar, A., Rudinger, R., Van Durme, B., 2018. Hypothesis only baselines in natural language inference, in: Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics, pp. 180–191.

Popescu-Belis, A., 2008. Dimensionality of dialogue act tagsets. Language Resources and Evaluation 42, 99–107.

Porcheron, M., Fischer, J.E., Reeves, S., Sharples, S., 2018. Voice Interfaces in Everyday Life. Association for Computing Machinery, New York, NY, USA. p. 1–12.

Ramadan, O., Budzianowski, P., Gasic, M., 2018. Large-scale multi-domain belief tracking with knowledge sharing, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp. 432–437.

Ringberg, T., Odekerken-Schröder, G., Christensen, G.L., 2007. A cultural models approach to service recovery. Journal of Marketing 71, 194–214.

Rocktäschel, T., Grefenstette, E., Hermann, K.M., Kočiskỳ, T., Blunsom, P., 2015. Reasoning about entailment with neural attention, in: Proceedings of the 2015 International Conference on Learning Representations.

Sacks, H., Schegloff, E., Jefferson, G., 1974. A simple systematic for the organisation of turn taking in conversation. Language 50, 696–735.

Saha, T., Patra, A., Saha, S., Bhattacharyya, P., 2020. Towards emotion-aided multi-modal dialogue act classification, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 4361–4372.

Saha, T., Srivastava, S., Firdaus, M., Saha, S., Ekbal, A., Bhattacharyya, P., 2019. Exploring machine learning and deep learning frameworks for task-oriented dialogue act classification, in: 2019 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–8.

Salton, G., Wong, A., Yang, C.S., 1975. A vector space model for automatic indexing. Communications of the ACM 18, 613–620.

Schegloff, E., Jefferson, G., Sacks, H., 1977. The preference for self-correction in the organization of repair in conversation. Language 53, 361–382.

Schegloff, E.A., 1992. Repair after next turn: The last structurally provided defense of intersubjectivity in conversation. American Journal of Sociology 97, 1295–1345.

Schegloff, E.A., 2007. Sequence Organization in Interaction: A Primer in Conversation Analysis. volume 1. Cambridge University Press.

Schloss, D., Gnewuch, U., Maedche, A., 2022. Towards designing a conversation mining system for customer service chatbots .

Seeger, A.M., Heinzl, A., 2021. Chatbots often fail! can anthropomorphic design mitigate trust loss in conversational agents for customer service? ECIS 2021 .

Shneiderman, B., 2021. Human-centered ai: A new synthesis. Ardito C. et al. (eds) Human-Computer Interaction – INTERACT 2021. INTERACT 2021. Lecture Notes in Computer Science, vol 12932. Springer, Cham. 12932, 3–8.

Socher, R., Pennington, J., Huang, E.H., Ng, A.Y., Manning, C.D., 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions, in: Proceedings of the 2011 conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 151–161.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research 15, 1929–1958.

Tsuchiya, M., 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need, in: Advances in neural information processing systems, pp. 5998–6008.

Ye, F., Manotumruksa, J., Zhang, Q., Li, S., Yilmaz, E., 2021. Slot self-attentive dialogue state tracking, in: Proceedings of the Web Conference 2021, pp. 1598–1608.

Highlights - A Tag-based Methodology for the Detection of User Repair Strategies in Task-Oriented Conversational Agents

- In order to improve the understanding capabilities of an agent, it must be able to spot repair strategies employed by the user
- A complete framework to detect these strategies does not exist
- A new set is elaborated and applied to a dataset of conversations
- Results prove the applicability of the tagset and are used to develop a classifier

Vitae

Francesca Alloatti

Francesca Alloatti obtained her Master degree in Linguistics in 2018 and she obtained her PhD in Computer Science at the University of Turin in 2022. She is currently a Computational linguist for H-FARM Innovation. Her double affiliation in university and industry brought her to investigate the interaction between real users (e.g. in a real life setting) with conversational agents. Her main interest is to close the gap in Human-Computer Interaction in terms of mutual understanding between the human part and the machine one.

Giovanni Siragusa

Giovanni Siragusa has obtained his PhD in Computer Science by Università degli Studi di Torino with a thesis on a Content Aware Attention Method for Neural Abstractive Summarization.

His research interests lie in the field of Automatic Text Summarization and Chatbots. For the former one, he is studying methods to exploit the document(s) via pointer network and attention in order to improve the generated summary; for Chatbots, he is interested in Open Domain Chatbot and the generation of more sentiment and human-like utterances.

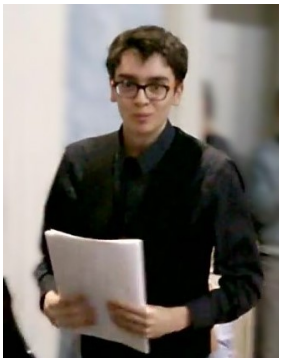
Luigi Di Caro

Luigi Di Caro is Associate Professor at the Department of Computer Science of the University of Torino. He has a Ph.D. in Computer Science, and his main interests include Artificial Intelligence (AI), Natural Language Processing (NLP), Data Mining (DM), Machine Learning (ML), Legal Informatics (LI) and related interdisciplinary interactions with Cognitive Sciences (CS) and social-impact applications. Luigi Di Caro has active international collaborations with more than 50 people in different countries, and leads the NLP activities within the “Social Computing” research group of the Department of Computer Science.

Francesca Grasso

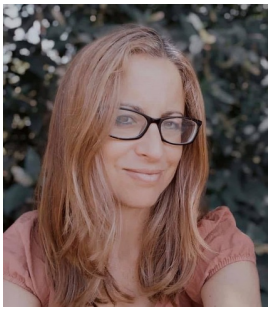
Francesca Grasso obtained her Master Degree in Linguistics in 2020 with a thesis in Germanic studies and Theoretical Linguistics and she is currently a postgraduate researcher in the Department of Computer Science at the University of Turin. Her main research areas include Knowledge Modelling – with a particular focus on Lexical Semantics – and Human-Computer Interaction (HCI),

with special regard to Chatbots. She is primarily interested in merging actual linguistic knowledge and computer science research in order to develop innovative and linguistically (i.e. empirically) motivated solutions for the design of both Conversational Agents (CA) and semantic networks.



Roger Ferrod

Roger Ferrod received his M.Sc. degree in Computer Science from University of Turin (Italy), in 2020, with a thesis on the automatic detection of users' expertise from dialogues. He is currently a PhD candidate at the same department working on text classification, relation extraction and spelling correction over biomedical texts. His research interests include Ontology Learning, Graph Neural Networks and citation networks.



Federica Cena

Federica Cena is an associate professor at the Computer Science Department of the University of Turin. She works on the intersection on Artificial Intelligence and Human Computer Interaction. In the last years, she is mainly devoted in studying the implications of Internet of Things for user modeling and personalisation, with a special focus on assistive applications for cognitive disabilites and frailty. She is the author of more than 100 scientific publications at conferences and in international journals. Home page: http://www.di.unito.it/~cena.

Author contribution statement

Francesca Alloatti: Conceptualization, Data curation, Formal analysis, Resources, Methodology, Writing – original draft.

Francesca Grasso: Conceptualization, Data curation, Formal analysis, Methodology, Writing – original draft.

Roger Ferrod: Methodology, Software, Visualization, Formal analysis, Writing – original draft.

Giovanni Siragusa: Methodology, Software, Visualization, Formal analysis, Writing – original draft.

Luigi Di Caro: Supervision, Validation, Writing – review & editing.

Federica Cena: Supervision, Validation, Writing – review & editing.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: