Université de Lorraine

L'Institut des Sciences du Digital Management Cognition

Supervised Project

What are you saying?
Dialogue act annotation

Authors:
Albert Millert,
Anar Yeginbergenova

Supervisors:
Chuyuan Li,
Maria Boritchev,
Maxime Amblard

A final report of the Supervised Project

Academic year 2020-2021

June 18, 2021

Contents

List of Figures

List of Tables

1 Introduction

The main objective of our work was to investigate the influence of different types of annotation - *RST* (Mann and Thompson, 1988a), *SDRT* (Asher and Lascarides, 2003); and datasets on the evaluation of the *Deep sequential model for discourse parsing on multi-party dialogues* (Shi and Huang, 2019). Instead of adjusting the model's architecture, we have decided to investigate the possibility of training a universal model yielding decent results for various (yet similar) datasets. We have used the following corpora in the work: *STAC* (Afantenos et al., 2015, Asher et al., 2016), *DAIC* (Gratch et al., 2014, DeVault et al., 2014), *Molweni* (Li et al., 2020), and *GUM* (Zeldes, 2017). We have analyzed each evaluation carefully to increase the interpretability of the model. Performing diverse experiments allowed us to obtain some insights into desirable characteristics of the datasets to retrieve the optimal results. We emphasize the discourse datasets since the original inspiration came from the task concerning the classification based on the patients' interviews for early diagnoses. While only a small part of the datasets are concerned with the topic directly, the others should, at least, provide linguistic features complementing the topic of the main idea. For example, interviews or, more generally - dialogues, are related to the original style of discourse provided in the *DAIC* dataset.

While working on the project, a few questions arise: 1) *"Is the model universal enough to be able to work with various datasets seamlessly?"* 2) *"Does the punctuation improve the model's performance?"* 3) if so - *"under what circumstances?"* In the following sections, we address these questions. On top of that, we describe the work we have put into the research. We also provide the reasoning behind the choices we have made along the way.

We begin by presenting various researches in the domain 2. Firstly, we focus on general discourse parsing techniques and approaches 2.1. Finally, we provide some similarities to our experiments 2.2.

The subsequent section 3 provides a comprehensive description of all datasets we have used to investigate the problem. After a brief introduction to each of them 3.1, we present both similarities and differences between them 3.2. Then, we discuss alternations introduced in the datasets and the reasoning behind doing so 3.3. We provide all the datasets' sizes in the respective tables for readability. A universal model must not rely on the size of the dataset since lots of available corpora are too small. The *Deep sequential model for discourse parsing on multi-party dialogues* is described in more detail in 4.

The following section - 5 regards all the experiments performed on the presented datasets. Additionally, we discuss the obtained results. Then, we provide a more detailed discussion and analyses in 6; there are also visualizations of the relations and their types which is the basis of the way our model functions. Not all the data contained the correct annotation, hence the need for alternative methods of model evaluation. The visualization is meant for this very reason, to acquire some interpretability of the learning process and obtained results.

In the last section 7 we gather the observations and provide the conclusions. We also include the propositions of how one could continue the research in this domain in (7.2). On top of that, we provide some resources which might become helpful in this regard. For various reasons, a part of all the tasks initially considered when planning the research has been postponed or rejected. We provide the reasoning and additional explanations in 7.1.

2 Related Work

Studies on dialogue parsing include building abstract representations by utilizing formal methods or machine learning models. Formal methods allow one to construct a comprehensive semantic representation of processed texts (both monologue and dialogue). Montague, 1970 stated that natural languages can be interpreted in terms of a language of logic and that they can, and should be, based on the same principles. However, Montague failed to provide a universal logical representation of discourse.

Soon, a theory facilitating a formal representation of discourse with the consideration of the dynamics of language was introduced (Kamp, Van Genabith, and Reyle, 2011) as *Discourse Representation Theory* (*DRT*). Instead of examining inputs consecutively sentence by sentence, this approach considers the sequence of sentences. It examines how the representation of new discourse affects the already processed data. DRT constructs a logical representation from which the original text could be derived easily. The paradigm is considered classical formal semantics by considering two assumptions: 1) the hearer building the mental representation of the sentences, 2) every following sentence is an addition to this representation. As further was concluded, the above assumptions cannot be valid simultaneously.

Following the motivation of *DRT*, Asher and Lascarides, 2003 introduced *Segmented Discourse Representation Theory* (*SDRT*) which adds discourse coherence theories alongside the *DRT*. *SDRT* proposes 16 discourse relation types with different utterances' pairs being assigned a relation type among: *Question-answer pair*, *Comment*, *Question Elaboration*, *Acknowledgement*, *Elaboration*, *Alternation*, *Explanation*, *Result*, *Continuation*, *Parallel*, *Correction*, *Conditional*, *Contrast*, *Clarification question*, *Narration*, *Background*. These are used in to connect sentences with each other resulting in a fully coherent structure.

We have taken the work of Afantenos et al., 2015 as the basis since it outperformed previous works (Li et al., 2014, Afantenos et al., 2015 with Muller et al., 2012) in the discourse representation domain. Previously, the state-of-the-art approaches for discourse parsing were either relying on hand-crafted features; the pipelines were not optimal to use on a bigger scale. For example, Shi and Huang, 2019 introduced the model computing the probability of the dependency relations between the combination of two utterances. The discourse structure was constructed based on the estimation of the probabilities. The major drawback of this approach is that the assignment of the dependency relations is limited to the local information that does not allow one to build an accurate discourse structure.

2.1 Discourse parsing research

There is plenty of research around discourse parsing. However, not many methods focus on the influence of different datasets on the same model, especially in the absence of proper relations; annotations between the utterances.

Ji and Eisenstein, 2014 presents research emphasizing representation learning by transforming surface features into a latent space utilizing the *Rhetorical Structure Theory* (denoted as *RST*) discourse parsing. By combining the large-margin transition structured prediction with representation learning, they have managed to parse discourse while jointly learning a projection of the surface features. The method improved the results obtained in the previous state-of-the-art in terms of relation prediction.

In Biran and McKeown, 2015, authors provide a tool for the full discourse parsing in the *Penn Discourse Treebank* (denoted as *PDTB*) framework. The *two taggers* method is based on two tagging tasks, namely - 1) identifying the relation per sentence and 2) identifying the relation between consecutive pairs of sentences. The authors prove that sequential information is crucial for cross-sentence discourse relations. They have facilitated a simple argument span identification to achieve state-of-the-art results. They have also published their parser publicly.

Authors of Perret et al., 2016 implemented a discourse parser which is responsible for predicting non-tree *DAGs* (directed acyclic graphs). They utilized *Integer Linear Programming* for encoding both the objective function and constraints as global decoding over local scores. A dataset used in their work came from multi-party chat dialogues. Their work is based on the distribution of relations coming from the *SDRT* annotations.

Braud, Coavoux, and Søgaard, 2017 introduced a discourse parser that exceeded the baseline *System MFS* which labels all nodes with the most frequent relations in the training and development sets for English in the majority of metrics. Authors claimed that their experiment was the first experiment on cross-lingual discourse parsing (English, Brazilian, Spanish, German, Dutch, Basque, and others).

In Jia et al., 2018b authors propose a transition-based discourse parser facilitating memory networks taking discourse cohesion into account. This technique significantly improves discourse parsing, especially for long-span scenarios. The work proved to outperform both the traditional feature-based methods. Later, the authors provide another model (Jia et al., 2018a). They claim that most of the research focuses on analyzing a whole discourse at once. Such a method fails at finding longer-span relations and at representing them properly as discourse units. Their *long short-term memory* (*LSTM*) model works in two stages: one to parse intra-sentence and the other one for the inter-sentence discourse structures. Their research has shown to improve the parsing works conducted in English and Chinese.

In Joty et al., 2019 authors explained that discourse parsing is a broad set of NLP-related tasks which consist of discovering the topic structure, coherence, coreference resolution, and conversation structure. Part of the paper compares the discourse analysis between monologue and conversation, synchronous vs. asynchronous conversations. They also discussed the crucial linguistic features in the discourse analyses.

Other works focused on the role underlying the information flow and argumentative structure in natural languages. In Liu, Shi, and Chen, 2020 authors claim that the parsing task for languages (different than English) is partially skipped in the research. This happens due to the lack of the annotated data. They propose a neural, cross-lingual discourse parser that facilitates multilingual vector representations with the segment-level translation of the source data. The training data they have used was small, yet it performed comparably to the state-of-the-art methods on cross-lingual document-level discourse parsing. The datasets (as can be observed in more detail in 3.2) we have chosen to use are limited in size as well.

2.2 Chosen methods

Up until this point, we have presented the work that has influenced the research in the general domain of discourse parsing. In this view, these researches underlie any research which even indirectly relates to the topic. However, in this subsection, we would like to introduce the works which directly influenced and inspired our work.

Shi and Huang, 2019 introduced a sequential model for dialogue parsing that allows building discourse structures by taking into account not only the local information but also the context within the dialogue. In this research, the authors used the *STAC* dataset (Asher et al., 2016, Afantenos et al., 2015) to perform their experiments, the corpus of multi-party dialogues annotated for discourse structure in the style of *SDRT* (Asher and Lascarides, 2003). We utilized their findings alongside the model and data to test against a broader set of applications investigating its universality.

There are several projects on discourse parsing and annotation. The first one considered in our work has been published in Li et al., 2020. The authors have publicly published a dataset containing several multi-party dialogues from the *Ubuntu Chat Corpus* annotated in the *SDRT* style. Another similar project is the *GUM* corpus (Zeldes, 2017). It is an open-source multilayer corpus of richly annotated texts in the style of *RST* (Mann and Thompson, 1988b). The *Distress Analysis Interview Corpus* (*DAIC*) (Gratch et al., 2014) is the set of transcripts from clinical interviews. We have primarily conducted our research around this dataset. Compared to other considered datasets, this one does not contain the relations' annotations; hence, it has been used for testing and extrinsic evaluation of the model.

3 Details and Descriptions of the Datasets

This section regards more detailed descriptions and their comparison. To better understand the task and to be able to analyze the results, we shall investigate the nature of the data. One cannot draw meaningful conclusions regarding why a model trained on some dataset performs better in the classification problem than the one evaluated on some other data without knowing how diverse the considered datasets are.

3.1 Brief overview

The *Distress Analysis Interview Corpus - DAIC* dataset has been introduced in (Gratch et al., 2014) and (DeVault et al., 2014). It comes from English transcripts of the interviews between a virtual assistant called *Ellie* and patients. Originally, the *DAIC* dataset has been used in the classification problem to help detect depression early among the patients. Each video transcription has been hand-annotated by a professional.

The *Strategic Conversation* - (*STAC* - (Asher et al., 2016)) corpus consists of transcribed and manually annotated chat conversations of the players exchanging resources and negotiating goods during the gameplay.

The *Molweni* - a machine reading comprehension dataset (Li et al., 2020) has been developed over multiparty dialogue. It contains samples from the *Ubuntu Chat Corpus* (Uthus and Aha, 2013).

We have additionally merged the *Molweni* and *STAC* datasets. It is denoted as *STAC x Molweni* or *S x M* for brevity. There are two main reasons behind why we decided to merge datasets:

1. the two test datasets are too small compared to *DAIC*,
2. we wanted to combine two datasets which domains of applications were different and investigate whether merged information would improve the model or help predict the relations.

We were ensured in our decision by the fact that both datasets were annotated using the same technique - *SDRT*.

All the mentioned scenarios have utilized *Segmented Discourse Representation Theory* - (*SDRT* Lascarides and Asher, 2008) framework to annotate the data (see 2).

To include a more various dataset and to compare the influence of the types of annotation on the model's performance, we have used another type of annotation, namely - *GUM* - an open-source multilayer corpus containing twelve types of annotated texts. Each year the dataset (Zeldes, 2017) is expanded by the students of the Georgetown Univesity as part of their course on annotation.

3.2 Comparison

When discussing the data sizes, we will refer to both sizes of the training and testing data. In either case, we have utilized the whole dataset without restricting ourselves to a portion of it; to not introduce unwanted bias. We strived to make the most out of the data without limiting the domain. We will compare the sizes of the datasets annotated using the same technique - *SDRT* since the relations are of the same type. Hence, it makes sense to compare them.

The sizes have been juxtaposed as tables 1, 2. As one can see, the datasets vary greatly but when considered in terms of training and test sets, they preserve the order of magnitude of sizes. *STAC* dataset denotes regular *STAC* data, whilst *STAC NP* denotes *STAC* data with all the punctuation removed. Similarly, *Molweni NP* denotes *Molweni* dataset without punctuation, and *STAC x Molweni NP* - *STAC x Molweni* without punctuation respectively.

The *DAIC* dataset originally lacks both punctuation and annotation; in consequence: 1) there is no need to add punctuation, 2) this dataset ought not to be used to train the model. 1) is the exact reason why we have decided to remove punctuation in the other datasets - we wanted to get them closer to *DAIC* which we consider the most important for the task. *DAIC cont full* is the entire *DAIC* dataset with naively annotated speakerships' *Continuations* in the interviews. The *Continuation* relation type occurs when two subsequent utterances belong to the same speaker. *DAIC cont short* is the same data, but it was randomly subsampled to make the data smaller. Intuitively, the smaller dataset should be used for testing, whilst the bigger one - for training. That is not the case in our experiment. As mentioned before, *DAIC* dataset doesn not contain proper *SDRT* structure annotation; therefore, it cannot serve as the training set. We have decided to use the two since we wanted to investigate whether the smaller data would be informative enough to yield the same results as the bigger one.

TABLE 1: Sizes of the training datasets

Dataset Sizes	Dialogues	Utterances	Relations	Punctuation
STAC (NP)	1026	11432	11109	YES (NO)
Molweni (NP)	9000	79487	70452	YES (NO)
STAC x Molweni (NP)	1026	90919	81561	YES (NO)
DAIC cont full	188	47153	25780	NO

TABLE 2: Sizes of the test datasets

Dataset Sizes	Dialogues	Utterances	Relations	Punctuation
STAC (NP)	111	1156	1126	YES (NO)
Molweni (NP)	500	4430	3911	YES (NO)
STAC x Molweni (NP)	611	5586	5037	YES (NO)
DAIC cont short	10	2563	1467	NO

The *STAC* training data contains 1026 dialogues with 11432 utterances and 11109 relations among them. Test data constitutes 111 dialogues with 1156 utterances and 1126 relations. We have obtained the corpus by merging multiple game transcription logs.

The *Molweni* training data contains 9000 dialogues with 79487 utterances and 70452 relations. Test data constitutes 500 dialogues with 4430 utterances and 3911 relations.

The dataset was of the correct format, so we could use it easily.

The *DAIC* dataset contains 188 dialogues with 47153 utterances and 25780 relations. We do not distinguish between the training and test data. The shorter version is simply a subset of all interviews. It contains 10 dialogues, 2563 utterances and 1467 relations.

3.3 Altered datasets

Additionally, we have processed all the datasets (naturally, excluding *DAIC*) so that they do not contain punctuation. The obtained datasets are supposed to be more similar to the *DAIC*. We wanted to investigate how such adjustments influence the prediction. The *DAIC* dataset is the only one originally not containing the punctuation. Our experiment utilizes the *Deep sequential model* originally trained on the *STAC* dataset. But we were primarily concerned with the *DAIC* data. When considering both, we came up with the idea that we needed the other datasets to become as close to the *DAIC* as possible while bearing in mind that the model was implemented specifically for the *STAC*.

The corpus obtained from the merge of *STAC* and *Molweni*, namely - *STAC x Molwni*, training data contains 10026 dialogues with 90919 utterances and 81561 relations. Test data constitutes 611 dialogues with 5586 utterances and 5037 relations. The numbers are the sum of the counts of the other datasets, as expected. Removal of the punctuation has been applied to this hybrid dataset equally.

The most common discourse relation types in the *STAC* dataset are *Question-answer pair*, *Comment* and *Acknowledgment* both in test and train data. Whereas when considering the *Molweni* corpus, we have found that the most common relations between the utterances are *Comment*, *Clarification question* and *Question-answer pair*. A slight difference between the relation frequencies may indicate a different nature of the data and the domain of application. However, having different datasets of various domains helps evaluate the model and investigate its quality.

3.4 Speakership statistics in DAIC

We have calculated the statistics for the *DAIC* dataset on its subset (roughly $\frac{1}{3}$) of the entire data. The main focus of the computations was to explore the turns in the conversations and compare the speakership between patients and an online avatar. The statistics would be way more insightful if there were specialist diagnoses' annotations provided or - any additional information about patients.

Table 7 provides simple statistics regarding patients' speakerships in the interviews. This information gives a general insight into the size of the dataset in terms of speakership turns.

The shortest (turn-wise) interview of a total of 83 turns consists of 42 patient's turns contributing to over 50% of patient's speakership share in the interview. However,

TABLE 3: Juxtaposition of the simple statistics between patients' and overall turns in the interviews

Turns	Min	Quartile I	Quartile II	Quartile III	Mean	Std	Max
Patients	42	97.5	121.5	157.75	139.5323	74.1953	386
Overall	83	178	212.5	249	226.7581	82.2847	473

TABLE 4: Patient's speakership share in total length of the interview

Turns	Min	Quartile I	Quartile II	Quartile III	Mean	Std	Max
Patients	38.93%	53.37%	57.55%	64.44%	59.12%	9.36%	81.61%

we have calculated the minimal patient's speakership share of 38.93% for the interview with 175 turns in total. It means that the shortest interview does not relate to the lowest patient's speakership share. The interview with a maximum length of the total amount of turns 473 is also the interview with the highest share of patient's speakership - 81.6%. That indicates that usually, for the interviews with patients' speakership over 75%, the overall amount of turns does not drop below 400 turns level.

In general, no interview with a patient's share under 50% exceeded the length of 200 turns in total. That indicates (more or less) that shorter interviews have a higher chance of having been done with a bit less talkative patient (turn-wise). Without deep analyses, one can naïvely draw a pair of hypotheses that often: 1) shorter interviews (turn-wise) correspond to a lower share of patient's speakership in the whole interview; 2) longer interviews - the patient's speakership share tends to be higher.

On average, the share of the patient's speakership in the interview is close to 60% ($\sim$140 turns), when the average interview consists of roughly 230 turns. Given values of the 3^{rd} quartile, one can observe that the interviews' lengths in $\frac{3}{4}$ of all documents and the number of patients' speakerships are close to the mean value. That hints that the values closer to maximum (with very talkative patients) are a bit more rare cases than the others.

It is crucial to point out that the values in table 7 are independent of each other, i.e. minimum values in the *Patients* row and the *Overall* row do not necessarily refer to the same interviews (however, in this case, they do). The table should not be interpreted pairwise (single columns *min*, *max*) are not obtained for the same interview - these are two independent values, even though they may turn out to refer to the same interview. To retrieve the information about the ratio of patients' share to the overall length of the interview (turn-wise), one should refer to the table 8 in which values correspond to the patient's speakership within the interviews.

The average token's length observed in the patients' turns is 3.6 long. Words of lengths 4, 2, 3, 5 have the biggest share among other word lengths. 4-character words make up 23.78%, 2-character - 23.26%, 3-character - 19.84%, 5-character - 4.66%. This group of the most common words' lengths altogether makes up roughly 72% of all the tokens. The average amount of tokens within a single patients' turn is 9.56, with minimal value - 1, and maximum - 125. It appears that the shorter turns are a lot more probable to occur in patients' utterances. Single token utterances make up to 19.99%, 2-token - 9.19%, 3-token - 7.21%, 4-token - 6.05%. It is important to note that many of the turns consisting of single tokens appear to be responses to commonly known *yes/no-questions*. We have observed 1729 of such single token utterances in the *DAIC* dataset sample. Table 5 constitutes the most common tokens found in this category of single token patients' turns.

Table 5 provides the insights to the most common tokens establishing single-token turns in the discourse found in the analyzed subset.

TABLE 5: The most common tokens and their share among the category of single-token patients' turns

Token	Tokens share in the category %
um	25.56
yeah	8.16
no	8.1
uh	7.35
yes	6.83
<laughter>	4.45
mhm	3.53
so	2.78
mm	2.55
okay	1.91

4 Deep Sequential Model

Authors of Shi and Huang, 2019 have proposed the *Deep Sequential Model model for discourse parsing on multi-party dialogues*. We have chosen it for the research due to its free availability online and its undeniably related use-case to the one of our interest. On top of that, the model is a current state-of-the-art yielding the best results for the multi-party discourse parsing task. The datasets used for training the model are primarily the discourse transcriptions annotated using the *SDRT* technique. Instead of focusing on tweaking and adjusting the model's architecture.

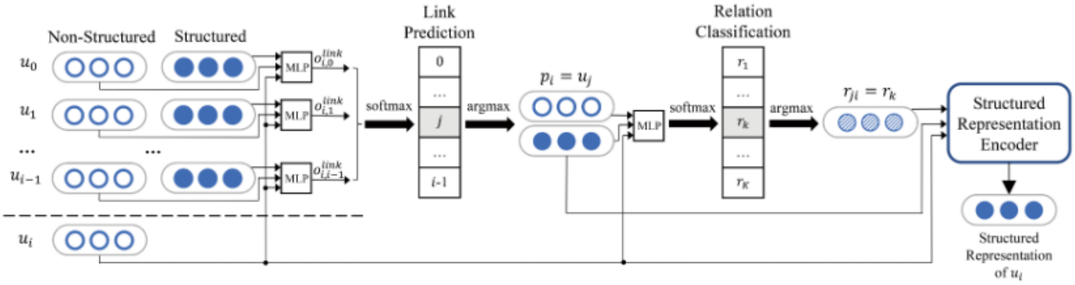

FIGURE 1: Visualization of the *Deep Sequential Model for Discourse Parsing on Multi-Party Dialogues* Afantenos et al., 2015

The Deep sequential model constructs the discourse structure incrementally by predicting dependency relations and building structure jointly and alternatively. The model sequentially scans the utterances, in the dialogue, the so-called *Elementary Discourse Units* (*EDUs*). The model decides for each *EDU* to which previous *EDU* the current one should link and what is the relation type. This approach does not rely on the local information about the utterances but - on the already constructed discourse structure.

The model first computes the non-structured representations of the *EDUs* with hierarchical *Gated Recurrent Unit* (*GRU*) encoders (Cho et al., 2014). Such representations are then used to predict the dependency relations and to encode those structured representations. Next, the model makes sequential scan through the *EDUs* and performs the following three steps to handle a respective *EDU* - u_i:

1. **Link prediction** facilitating the prediction of the parent p_i of the u_i *EDU*, followed by the computation of the scores between u_i and the candidates $u_j(j < i)$ with the multilayer perceptron model (*MLP*) (Gardner and Dorling, 1998). The softmax function is then used to perform the normalization and choose the linked *EDU* with the largest probability. We refer to the evaluation metric denoting this task as *Bi* score.

2. **Relation classification** predicting the relation between p_i and u_i; both of which are fed into the *MLP* to obtain the distribution over the relation types. The assignment of the relation type r_{ij} consists of choosing the one with the highest probability attached to it. We refer to the evaluation metric denoting this task as *Multi* score.

3. **Structured representation encoding** is based on the computation of a structured representation of u_i with a structured representation encoder responsible for encoding the predicted discourse structure. Relation embedding r_{ji}, the non-structured representation of u_i, and the structured representation of $p_i = u_i$ are fed into the encoder to derive a structured representation of u_i.

4.1 Model setup

Shi and Huang, 2019 work does not contain the exact information about the hyperparameters used to train the best-performing model. We have initially tried various settings. The goal we wanted to achieve was to get as close as possible to the authors' results. We have listed our results in 5. The results obtained by the model with the setup presented in this subsection (which we have used in the experiments later) have outperformed the results obtained using the default setup (provided in the source code). Both of them were slightly lower than what the authors have presented in their work. Perhaps, other setups could have yielded better results, but we wanted to investigate the model which was the closest to the one presented in the original paper; hence, we have stuck to the one we are going to describe.

TABLE 6: A set of hyperparameters of the *Deep sequential model* used in all the experiments; a version altered from the default setup provided by the authors since it yields results closer to the ones presented by them in the paper

Hyperparameter	Value
Vocabulary size	2000
Maximum distance between two related EDUs	20
Dimension of the word embedding	100
Dimension of the relation embedding	100
Dimension of the binary features (Bi)	4
Use structured encoder	True
Use speaker highlighting mechanism	True
Use shared encoders	False
Use random structured representation	False
Number of epochs	30
Number of hidden units	128
Number of RNN layers in encoders	1
Number of relation types	16 (related to SDRT)
Mini-batch size	16
Probability to keep units in dropout	0.5
Learning rate	0.8
Learning rate decay factor	0.989

5 Experiments and Results

Now that we have introduced different types of datasets, the model's architecture, and the evaluation metrics, we are ready to present and discuss the results obtained from the experiments. We introduce different runs with short analyses of the obtained scores in sequential order, the way we have conducted them.

According to the authors, the original F1 score obtained by the *STAC*-trained model, tested against the *STAC* data, equaled 73.2% for the link classification and - 55.7% for both the link and relation classification. When recreating their work, we have obtained results, on average, equal to 71.5% and 47.7% for F1 *Bi* and F1 *Multi* respectively. These results indicate that the model handles the binary classification task well, but it performs slightly worse when one adds relations' prediction to the link classification. A link classification is a link between speakers - the addressee of the speaker's utterance. In *SDRT*, a relation is classified as one of sixteen relation types (see 2).

As mentioned in 3, there has been another dataset at our disposal for the experiments - *Molweni*. We have tested previously described model, trained on *STAC* dataset, on the *Molweni* data. Obtained results reached 53.9% and 24.5% F1 scores for the link and link + relation accordingly. Even though part of the words from the test set is not present in the training set, all the words are treated equally by the mode. The reason for the model's low performance is that *STAC* corpus has a limited vocabulary. We do not expose the model to some vocabulary; therefore, it cannot learn the meaning of some utterances resulting in a decreased performance.

After this experiment, we have trained and tested the model on the *Molweni* data. Such a setting yielded F1 *Bi* score of 86.6% and F1 *Multi* score equal to 55.2%. The next experiment consisted of testing this model against the *STAC* corpus. The results have decreased substantially to F1 *Bi* equal 43.5% and 20.0% of F1 *Multi*. The difference between the two corpora is limited to the domain, the structure, and the environment. For instance, in the *Molweni* dataset, the utterances are lengthy and full, unlike the *STAC* data consisting of short utterances. Utterances in the *STAC* are terse because the gaming environment in which the speakers tend to communicate requires short and quick statements focusing on the informativity. Therefore, we conclude that the model trained on the *Molweni* dataset recognizes the patterns which result in the correct relation and link classification.

After obtaining these results, our next experiment was to merge both *STAC* and *Molweni* corpora to train the model. The F1 scores obtained for the *STAC x Molweni* data were 84.3% for the link classification, and - 51.9% for both link and relation classification. We then tested the model against both testing datasets individually. When tested on the *Molweni* dataset, it yielded 77.7% and 20.97% F1 scores. The same model has been tested against the *STAC* test data yielding 71.1 and 31.5 for F1 Bi and *Multi* respectively. It seems that the *STAC* dataset had way more influence on the combined data.

The main experiments have been juxtaposed in the two following tables 7 and 8. The notation remains unchanged from the brief description provided in section 3.2; suffix *NP* denotes no punctuation, and *S x M* - merge of two datasets *STAC* and *Molweni*. More details about different datasets can be found in 3 section. Training datasets are listed along the *Y*-axis while test datasets are listed along the *X*-axis.

A simple observation leads to a conclusion that the removal of punctuation, in most cases, improves the performance of the model. However, when applied to a model trained on standard data (containing the punctuation), it drastically decreases the F1 scores. It is especially prominent in the results obtained from testing a *Molweni*-trained model when applied on the *Molweni* test data itself vs. *Molweni NP* - without the punctuation. There is an additional bias introduced, in this case, since the *Molweni*-trained model tested against the *Molweni* test dataset uses the same type of data, hence, the nature of the dispute is more or less the same. However, considering the same model, but taking into consideration *STAC x Molweni* data with and without punctuation, one can observe a significant performance drop (but slightly smaller than in the previously described case). The differences seem to be less visible when analyzing the F1 *Bi* instead of the F1 *Multi* score.

The bias is the most obvious when considering both *STAC* and *Molweni* datasets. They are of a different nature (different sources and domains of discourse). When the original data does not contain punctuation, and we consider datasets of the same domain of discourse, one should remove the punctuation.

1. When the original data does not contain punctuation, and we consider datasets of the same domain of discourse, we should remove the punctuation to improve the performance not to mislead the model.
2. When the original data does not contain punctuation, and we consider a dataset

TABLE 7: F1-Multi scores juxtaposed for all the experiments

Train \Test	STAC	STAC NP	Molweni	Molweni NP	S x M	S x M NP	DAIC full	DAIC short
STAC	**47.733**	43.962	24.470	18.736	25.984	21.150	17.831	17.142
STAC NP	12.954	**45.700**	16.298	16.411	18.839	19.035	3.077	2.770
Molweni	19.975	15.545	**55.184**	24.695	42.460	33.858	9.198	10.769
Molweni NP	17.635	17.300	37.494	**45.676**	33.467	35.493	10.990	11.471
STAC x Molweni	31.509	26.828	20.880	21.061	**51.910**	35.386	25.468	27.117
STAC x Molweni NP	31.676	34.099	19.413	19.458	18.733	**44.633**	12.862	13.422

TABLE 8: F1-Bi scores juxtaposed for all the experiments

Train \Test	STAC	STAC NP	Molweni	Molweni NP	S x M	S x M NP	DAIC full	DAIC short
STAC	**71.515**	68.199	53.860	51.716	57.283	55.221	45.025	42.731
STAC NP	**71.291**	71.017	63.138	62.619	64.872	64.552	46.669	48.537
Molweni	43.544	44.964	**86.612**	75.643	68.657	69.119	36.691	38.978
Molweni NP	43.711	42.791	75.395	**86.080**	68.657	69.173	32.322	34.452
STAC x Molweni	71.041	69.118	77.652	77.111	**84.254**	75.411	45.991	48.030
STAC x Molweni NP	69.536	70.455	74.199	75.102	73.207	**83.932**	45.675	48.381

of a different domain of discourse, removal of the punctuation usually improves the model's performance (slightly).

3. When the original data contains punctuation, one should try to either provide it or keep it if it is already present.

6 Analyses

After running all the experiments and obtaining the results, now, we can dive into the analysis part of our experiments. In this section, we will evaluate the model performance, provide a thorough investigation of the results, as well as illustrate different points in figures.

As mentioned previously, we trained our model on different datasets and measured the performance of the model by testing on other datasets at our disposal. The final results allowed us to come to some conclusions on corpus and model itself. Since the datasets were of a different nature, in the process of experiments, we decided to merge two datasets hoping that it will help us increase the accuracy of the output. However, the results were somewhat different from what we have expected.

All three datasets we worked on have a similar structure: the utterance, the speaker, a link between utterances, i.e. to which speaker y the utterance of speaker x was addressed to, and the relation type. For the *STAC* dataset, the length of the utterances was short (on average), compared to the *Molweni*. The average length of the utterance in *STAC* data is 3.3, whereas in *Molweni* this number equals to 10.8. Hence, the STAC model performed worse when tested on Molweni because the model never learned to classify long sentences. On the other hand, *Molweni*-trained model worked relatively good when tested against the long data and slightly worse on the short ones. After realizing such a pattern, we merged these two datasets and trained the model using them to improve the overall performance and make the model adapt to different types of inputs. Such a decision allowed us to increase the vocabulary length, variability and afterwards feed the model with the sentences of different lengths. Therefore, the prediction accuracy increased when tested individually on *STAC* and *Molweni* datasets. According to the results, the model that was trained only on *Molweni* dealt outstandingly well on the classification of the *Question-answer pair* relation and it outperformed other models on this relation type.

However, when observing our data, we mentioned that if the model misclassified the question-answer type, those were the question-answer pairs without any obvious question words or question marks. Question mark and other punctuation removal have worsened the results; therefore, we can conclude that the model was biased towards punctuation. Nevertheless, the model seems to learn the question words such as "*wh*"-words, "*how*", etc. and could predict the relation types correctly when the utterances contained those words. In general, the combination of the datasets helped to increase the F1 scores for both datasets, but test results for *Molweni* improved to ∼2% and for the *STAC* data to ∼1%.

Overall, the model performance on the link prediction was very high for all datasets (with ∼70% accuracy). Turning into other relation types, for the *Molweni* among all

16 relation types the most frequent ones are *Comment*, *Clarification question*, *Question-answer pair* and *Continuation*. For the *STAC* data, they are *Question-answer pairs (QAP)*, *Comment*, *Acknowledgment*, *Continuation*. Common misclassification for the model was to classify an *EDU* with *Continuation* when the true label of it was *Elaboration* and vice-versa, same behaviour have been indicated between *Clarification question* and *Question-answer pair*.

In figure 2 the distribution of F1 scores is illustrated for the merged dataset. It shows that the predictions are very diverse and sometimes the model had highly accurate predictions, and sometimes not so much.

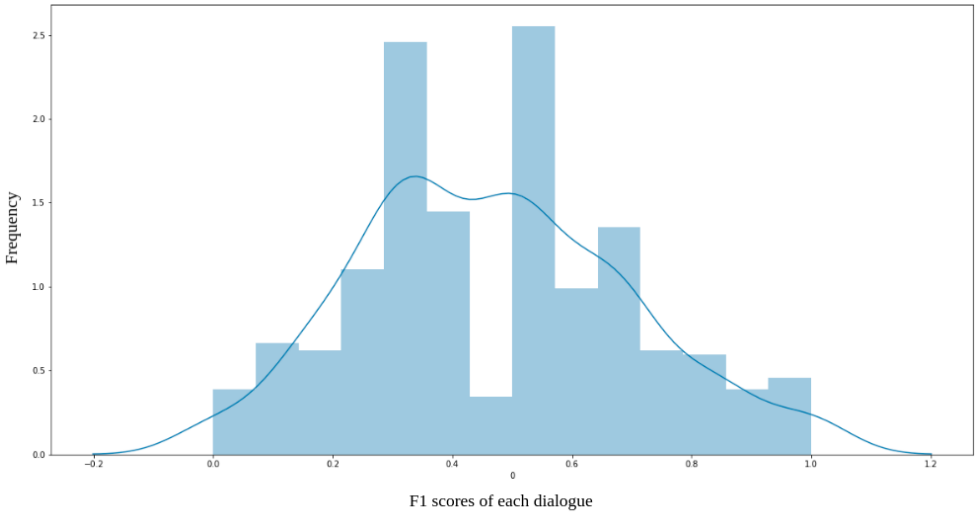

FIGURE 2: Visualization of the distribution of F1 scores of the 611 dialogues of the test data

6.1 Illustration of the results

Next, there is an illustration of predicted dialogues along with the representation of true labels, both for the worst and best predictions. In Figure 3, the top graph holds all true relations and links, and the bottom graph was predicted by the model; the nodes correspond to the utterances, the arrows show links and on top of them the relation types are written, the text corresponding to each node is located in the middle. The data is written in the way how the model treats it, therefore, the tokens such as *UNK* (short for unknown) mean that the word has never been seen in the training set and is not in the vector representation so the default UNK-token is assigned for such cases; numbers are replaced with the token *num* to treat all the numbers equally, because knowing each value of the numbers does not carry important information. The F1 score of this graph is equal to 0, i.e. nearly none of the predictions were classified correctly. We took this example to investigate the problem deeper and find why the model fails to predict correctly on simple dialogues.

The model assumes that the relation between first and second is *Acknowledgment* because it begins with the word ’yes’ and followed by some UNK tokens, in fact, it is one of the usual structure of the *Acknowledgment* in the corpus when the utterance starts from 'yes', 'right', 'thanks', etc. However, the model could not recognize the meaning of other parts of the sentence, and additionally, it fails to find the Question-answer pair. It sees that the 'wh'-words are question words, i.e. if the sentence consists of such word it is most probable to be a question, but the prediction failed with finding the correct location of the answer in the dialogue.

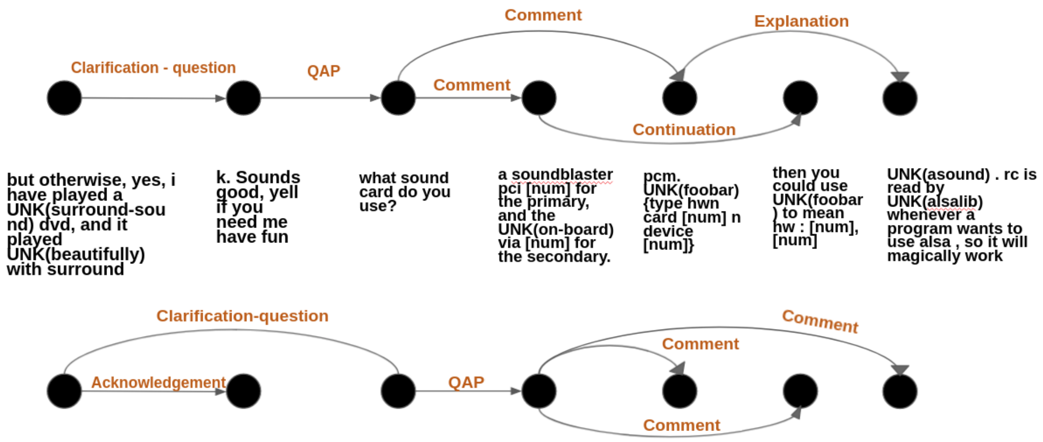
Comment
Explanation
Clarification - question
QAP
Comment
Continuation
but otherwise, yes, i have played a UNK(surround-sound) dvd, and it played UNK(beautifully) with surround
k. Sounds good, yell if you need me have fun
what sound card do you use?
a soundblaster pci [num] for the primary, and the UNK(on-board) via [num] for the secondary.
pcm. UNK(foobar) {type hwn card [num] n device [num]}
then you could use UNK(foobar) to mean hw : [num], [num]
UNK(asound) . rc is read by UNK(alsalib) whenever a program wants to use alsa , so it will magically work
Clarification-question
Comment
Comment
Acknowledgement
QAP
Comment

FIGURE 3: Graph representation of predictions. F1 score = 0 (top graph - true labels, middle - utterance of each node(speaker), bottom graph - results predicted by model, arrows - relation types)

The example above helps us to conclude that the model does not work perfectly. However, by looking at Figure 4, the opposite can be said. The model perfectly placed its predictions of links and relations, even with the sentences majorly consisting of *UNK* tokens. We discussed previously how the model classifies questions and in this example, the answer is located right after the question itself. In the example above the situation was identical, but the classification wrong. Knowing that, we may assume that the model locates all the answers right after the question, sometimes it works, but sometimes it does not. Nevertheless, other relation types were classified correctly. We mentioned before, that *Comment* is one of the most frequent relation types, by the fact that it assigned correct links and relations to the data we can presume that the model is not biased towards the most seen labels.

In these two examples, we showed the dialogues with F1 scores equalling 0 and 1 (the worst and the best). Such F1 scores are not frequent, and most of the predictions are between 0 and 1 as shown in figure 2. As the most parts of the classifications are explainable we may also conclude that the model learned the patterns of the 16 relation types.

6.2 Running model on the DAIC dataset

After performing all the mentioned experiments, the next step was to to try it on another corpus - *DAIC*, the corpus that has no gold annotation. The data differs from both *STAC* and *Molweni* by both the structure and the domain (interviews between patient and interviewer). The utterances are divided into several small chunks resulting in consecutive utterances produced by one person. In addition, this corpus had no punctuation as it was spoken data converted into a written one. Once we

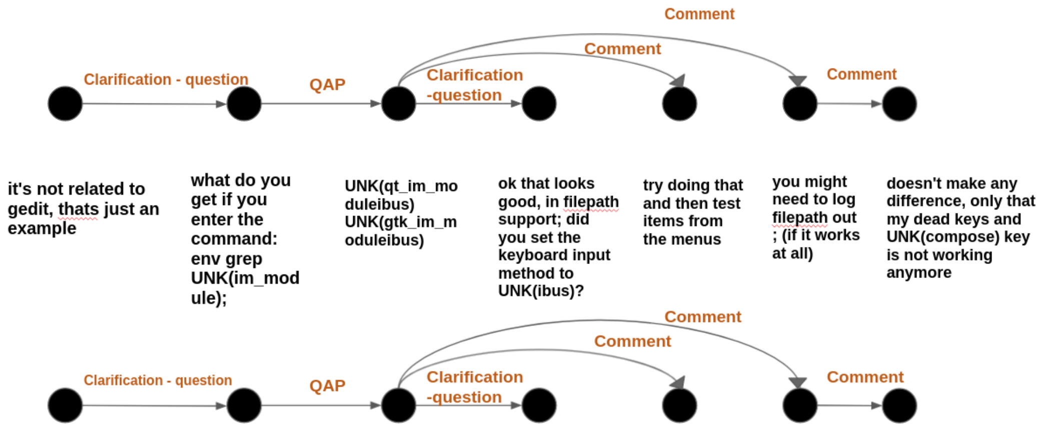
Comment
Comment
Clarification - question
QAP
Clarification
-question
Comment
it's not related to gedit, thats just an example
what do you get if you enter the command: env grep UNK(im_module);
UNK(qt_im_moduleibus) UNK(gtk_im_moduleibus)
ok that looks good, in filepath support; did you set the keyboard input method to UNK(ibus)?
try doing that and then test items from the menus
you might need to log filepath out ; (if it works at all)
doesn't make any difference, only that my dead keys and UNK(compose) key is not working anymore
Comment
Comment
Clarification - question
QAP
Clarification
-question
Comment

FIGURE 4: Graph representation of predictions F1 score = 1 (top graph - true labels, middle - utterance of each node(speaker), bottom graph - results predicted by model, arrows - relation types)

ran our trained models on *DAIC*, it was immediately discovered that the model fails to classify any *Question-answer pairs* and due to the nature of the dataset, most of the relations were predicted as *Continuation*. Soon enough we mentioned that other datasets had punctuation, especially question marks, which played crucial role in predicting *QAP*, and re-trained our model with removed punctuation from all corpus. Punctuation played a determinative role for the model since the performance dropped as well as the accuracy score.

The noticeable decrease was detected for *STAC* because, as it turned out, the questions were strongly relying on the question marks. For example, in "- *I can echange ore for wood though*", "*-give or want ore?*". It may even be hard for human to properly determine the relation type after removing the question marks (alongside the other punctuation). The *DAIC* corpus was ran on models trained on *STAC* and *Molweni*. Doing so, allowed us to better understand the behaviour of the model.

In the figure 5 one can observe that most of the time it is *Ellie* (the interviewer) who speaks and that for model, majority of the words are labeled as *unknown*, i.e. all the words with token *UNK* were treated equally by the model. This happened because of the limited vocabulary of the corpus it was trained on.

In Figure 6 the results of the same part of the dialogue but predicted by model trained on *Molweni* is illustrated. The model recognizes almost all the words in the utterance compared to *STAC* and recognizes *Question-answer pairs* even without any punctuation, but possibly because of the presence of the question words, i.e. *where, how*.

7 Conclusions

The main objective of the conducted research was to investigate the influence of the different criteria on the overall performance of the *Deep Sequential Model* (Afantenos

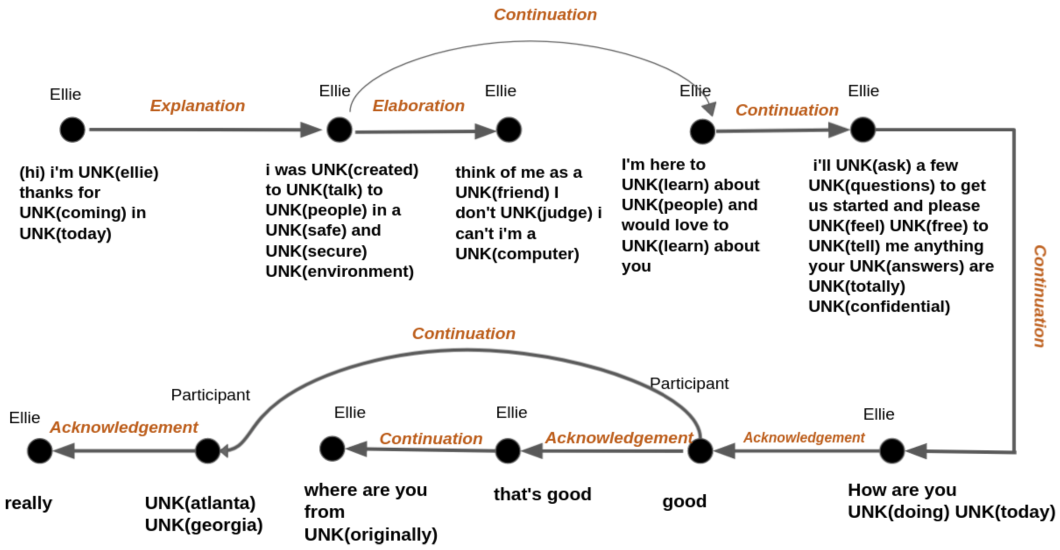
Continuation
Ellie
Explanation
Ellie
Elaboration
Ellie
Ellie
Continuation
Ellie
(hi) i'm UNK(ellie) thanks for UNK(coming) in UNK(today)
i was UNK(created) to UNK(talk) to UNK(people) in a UNK(safe) and UNK(secure) UNK(environment)
think of me as a UNK(friend) I don't UNK(judge) i can't i'm a UNK(computer)
I'm here to UNK(learn) about UNK(people) and would love to UNK(learn) about you
i'll UNK(ask) a few UNK(questions) to get us started and please UNK(feel) UNK(free) to UNK(tell) me anything your UNK(answers) are UNK(totally) UNK(confidential)
Continuation
Continuation
Participant
Participant
Ellie
Acknowledgement
Ellie
Continuation
Ellie
Acknowledgement
Acknowledgement
Ellie
really
UNK(atlanta) UNK(georgia)
where are you from UNK(originally)
that's good
good
How are you UNK(doing) UNK(today)

FIGURE 5: Graph representation of predictions for DAIC on model trained on STAC

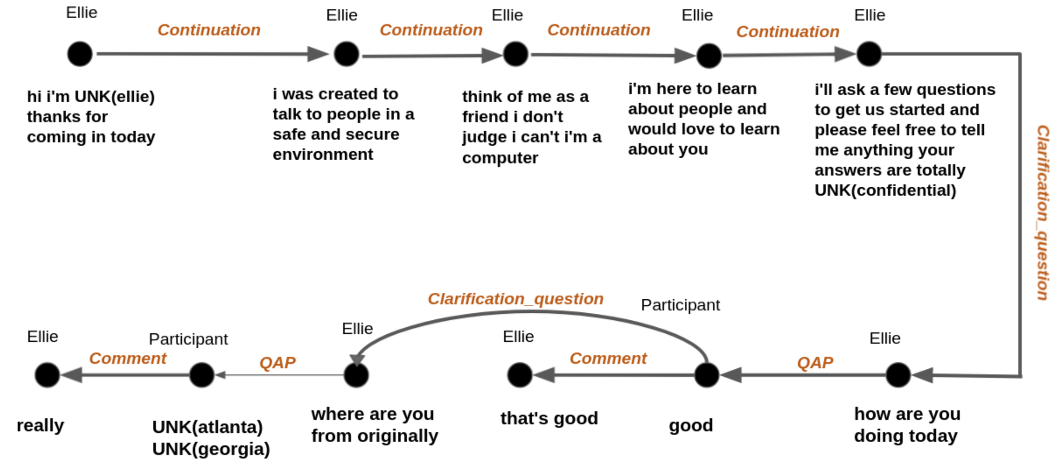
Ellie
Continuation
Ellie
Continuation
Ellie
Continuation
Ellie
Continuation
Ellie
hi i'm UNK(ellie) thanks for coming in today
i was created to talk to people in a safe and secure environment
think of me as a friend i don't judge i can't i'm a computer
i'm here to learn about people and would love to learn about you
i'll ask a few questions to get us started and please feel free to tell me anything your answers are totally UNK(confidential)
Clarification_question
Clarification_question
Participant
Ellie
Participant
Ellie
Comment
QAP
Ellie
Comment
QAP
Ellie
really
UNK(atlanta) UNK(georgia)
where are you from originally
that's good
good
how are you doing today

FIGURE 6: Graph representation of predictions for *DAIC* on the model trained on *Molweni*

et al., 2015). As the dataset representative of the primary domain of discourse, we have used the *DAIC* (Gratch et al., 2014) dataset. This dataset does not contain punctuation and is an interview between two participants exchanging the speakership in the act of dialogue discourse. Since we have used the model, specifically developed for the *STAC* research (Asher et al., 2016), we approached the problem of investigating whether the model is capable of representing knowledge in a slightly naive but universal manner. We have begun by trying to achieve a set of hyperparameters best representing the original model by reverse engineering it since we were not provided the exact configuration by the authors. They have only provided the final results. We tried to recreate their work and then use the obtained model's config for training and testing models on different datasets.

The obtained results have been merged into the tables in 3.2. We have shown more detailed analyses in 6. Based on the obtained results, we have proposed a set of hypotheses. Briefly, the removal of punctuation does improve the models' performance only if the original dataset did not contain this type of information either. It is especially prominent when testing the model on the dataset from the same domain of discourse. We conclude from this observation that punctuation in this task is just another meaningful type of information required by the model to draw relevant predictions regarding the relations between the utterances and their types.

7.1 Problems and discussion

The initial research plan was to build an abstract representation of the discourse by analyzing dialogues on different levels of abstraction. By establishing such an abstraction we believed to better understand and improve classification of patients in the interviews. The scope of the research was quite broad; therefore, we have decided to reuse the already existing tools freely available online. Our supervisor was in charge of contacting the author of the *DAIC* corpus to obtain the data. We were already aware of the research of *STAC* which was considered as part of the entire pipeline to build the abstract structure of discourse. On top of that, we have found two other promising candidates for the different layers of abstraction: 1) *Graph-based Dependency Parser*[1], 2) *Dependency Parser for Spoken Dialog Systems*[2], 3) *Conversational Banking*[3], 4) *ISO-compliant Dialogue Act Tagger**github.com/ColingPaper2018/DialogueAct-Tagger* repository 5) *Dialogue Discourse Parsing*[4]. We have faced difficulties at running some of them. The data we have chosen could not have been easily adjusted to be used with them. After some struggle (which ended successfully) with reusing the already mentioned *STAC* dataset and the model 5) provided by the authors, we have decided to investigate this path and continue the experiments using this tool.

We have rewritten the entire codebase into the newest version of *python* which would still be compatible with the archaic *Tensorflow's* version utilized by the authors. We have planned to switch the deep learning framework too, but after the discussion, we have decided not to spend more time on the development and begin the experiments' phase. At this point, we have had two datasets at our disposal - *STAC* and *DAIC*. The parser was adjusted and customized for our task; initially, it was retrieving the utterances from *XML* files which was not the case for the *DAIC* data.

We have discussed whether we should annotate the *DAIC* dataset, and if so, then how. The *DAIC* data has originally not been annotated in any way. We considered hand-annotation since it has been the most common approach in the research we have encountered. We have even implemented a tool for randomizing a subsample of the data. Following the hand annotation by the annotator, the data could have been merged back to the original data automatically. But the *DAIC* dataset turned out to be significantly larger than other discourse test datasets. We also did not consider ourselves experienced enough to annotate the relations properly. We have considered other methods, such as the *Educe* [5] tool which has been used for the *STAC* data but the required formats were too different.

[1] *github.com/tdozat/Parser-v3* repository
[2] *gitlab.com/ucdavisnlp/dialog-parsing* repository
[3] *github.com/tpimentelms/fast-conversational-banking* repository
[4] *github.com/shizhouxing/DialogueDiscourseParsing* repository
[5] *Educe* reference URL link

At this point, we have chosen to look for the other data. The other reason for it was the fact that *STAC* and *DAIC* were way different in the difference of discourse. We wanted to avoid, or at least - notice the introduction of the bias in the experiments. Since we have already had access to the updated tool for parsing the *SDRT-annotated* data into a proper *JSON* format needed by the model, we have opted for another, similarly annotated data. We have chosen to use *Molweni* data. We have then merged it with the *STAC* data to acquire more reliable and universal data. We wanted to test whether more variable data would be more representative for various related tasks. The initial *DAIC* dataset was substantially larger than the other ones; therefore, we have shrunk the *DAIC* dataset for testing purposes. On top of that *DAIC*, dataset transcriptions did not contain the punctuation. For this reason, we have obtained alternatives of all datasets by removing the punctuation.

We then compared all the datasets with each other by performing training on either one and then testing it against all the others except the *DAIC* dataset that was not suitable for training since it did not contain the relations' annotation so it could have only been used for testing.

We wanted to both improve and test the evaluation by introducing another discourse annotation of the utterances' annotations. We have approached to use the *GUM* dataset for this reason. We have tested one version available freely on the *github*[6] page, but all the tokens have been replaced by underscores "_" and we did not have access to the data needed for recreating the original data. We have then tried another annotated data again from *github*[7] but the utterances did not contain speakers' annotation. We had to merge the annotated data with the *Santa Barbara Corpus of Spoken American English*[8] which contained this information. The *GUM* data for the conversations was only a small subsample of the *Santa Barbara* data. The data was, however, not suitable for the task; the *conversation* was not a real dialogue - one person was speaking, the rest was responding by providing a number. For this reason, we had to reject the idea of using this data.

7.2 Future work

The model should be rewritten to a more up-to-date deep learning framework, such as *PyTorch*[9] working with newer versions of *CUDA*[10] architecture to improve the speed of evaluation and learning. We have rewritten the entire pipeline's implementation to a more recent *Python* version, namely 3.5 that is the highest possible version compatible with the archaic version of *Tensorflow*[11] - 1.3 which has been used by the authors to implement the model. This update in the implementation should allow further extensibility of both the model and research, giving more possibilities for the research domain's exploration. We would propose the work on a multi-lingual model training focused on a specific domain of application, e.g. patients' early depression detection. We strongly believe there should be some middle ground between the two datasets - English *DAIC* and French *SLAM - Schizophrénie et Langage*

[6] *github.com/DISRPT/sharedtask2021* repository

[7] *github.com/amir-zeldes/gum* repository

[8] *Santa Barbara Corpus of Spoken American* reference URL link

[9] *PyTorch* official website

[10] *CUDA* official website

[11] *Tensorflow* official website

: Analyse et Modélisation (Amblard, Musiol, and Rebuschi, 2015) developed as part of the *Sémagramme* team's research project. To investigate the topic, one could add another hidden layer to the model and perform transferred learning by first training on one data and then reusing the pretrained model to further update embeddings using the information provided in another dataset. It would also be an interesting topic of the research to better investigate the influence of the backchannels on the overall dialogue.

References

Afantenos, Stergos et al. (2015). "Discourse parsing for multi-party chat dialogues". In: Association for Computational Linguistics (ACL).

Amblard, Maxime, Michel Musiol, and Manuel Rebuschi (2015). "SLAM Schizophrénie et Langage: Analyse et Modélisation". In: *Journée de restitution CNRS PEPS HuMaIn*.

Asher, Nicholas et al. (2016). "Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus". In: *10th International Conference on Language Resources and Evaluation (LREC 2016)*, pp. 2721–2727.

Asher, Nicolas and Alex Lascarides (2003). "Logics of Conversation". In:

Biran, Or and Kathleen McKeown (2015). "Pdtb discourse parsing as a tagging task: The two taggers approach". In: *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 96–104.

Braud, Chloé, Maximin Coavoux, and Anders Søgaard (Apr. 2017). "Cross-lingual RST Discourse Parsing". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Valencia, Spain: Association for Computational Linguistics, pp. 292–304. URL: https://www.aclweb.org/anthology/E17-1028.

Cho, Kyunghyun et al. (2014). "Learning phrase representations using RNN encoder-decoder for statistical machine translation". In: *arXiv preprint arXiv:1406.1078*.

DeVault, David et al. (2014). "SimSensei Kiosk: A virtual human interviewer for healthcare decision support". In: *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pp. 1061–1068.

Gardner, Matt W and SR Dorling (1998). "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences". In: *Atmospheric environment* 32.14-15, pp. 2627–2636.

Gratch, Jonathan et al. (2014). "The distress analysis interview corpus of human and computer interviews." In: *LREC*, pp. 3123–3128.

Ji, Yangfeng and Jacob Eisenstein (2014). "Representation learning for text-level discourse parsing". In: *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 13–24.

Jia, Yanyan et al. (2018a). "Improved discourse parsing with two-step neural transition-based model". In: *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)* 17.2, pp. 1–21.

Jia, Yanyan et al. (2018b). "Modeling discourse cohesion for discourse parsing via memory network". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 438–443.

Joty, Shafiq et al. (2019). "Discourse analysis and its applications". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pp. 12–17.

Kamp, Hans, Josef Van Genabith, and Uwe Reyle (2011). "Discourse representation theory". In: *Handbook of philosophical logic*. Springer, pp. 125–394.

Lascarides, Alex and Nicholas Asher (2008). "Segmented discourse representation theory: Dynamic semantics with discourse structure". In: *Computing meaning*. Springer, pp. 87–124.

Li, Jiaqi et al. (2020). "Molweni: A Challenge Multiparty Dialogues-based Machine Reading Comprehension Dataset with Discourse Structure". In: *arXiv preprint arXiv:2004.05080*.

Li, Sujian et al. (2014). “Text-level discourse dependency parsing”. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25–35.

Liu, Zhengyuan, Ke Shi, and Nancy Chen (Dec. 2020). “Multilingual Neural RST Discourse Parsing”. In: *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, pp. 6730–6738. DOI: 10.18653/v1/2020.coling-main.591. URL: https://www.aclweb.org/anthology/2020.coling-main.591.

Mann, William C. and Sandra A. Thompson (1988b). *Rhetorical Structure Theory: Toward a functional theory of text organization*.

Mann, William C and Sandra A Thompson (1988a). “Rhetorical structure theory: Toward a functional theory of text organization”. In: *Text* 8.3, pp. 243–281.

Montague, R. (1970). *Universal grammar*.

Muller, Philippe et al. (2012). “Constrained decoding for text-level discourse parsing”. In: *COLING-24th International Conference on Computational Linguistics*.

Perret, Jérémy et al. (2016). “Integer linear programming for discourse parsing”. In: *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pp. 99–109.

Shi, Zhouxing and Minlie Huang (2019). “A deep sequential model for discourse parsing on multi-party dialogues”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01, pp. 7007–7014.

Uthus, David C and David W Aha (2013). *The ubuntu chat corpus for multiparticipant chat analysis*. Tech. rep. NAVAL RESEARCH LAB WASHINGTON DC.

Zeldes, Amir (2017). “The GUM Corpus: Creating Multilayer Resources in the Classroom”. In: *Language Resources and Evaluation* 51.3, pp. 581–612. DOI: http://dx.doi.org/10.1007/s10579-016-9343-x.