University *of Ljubljana*
Faculty *of Computer and Information Science*

Natural Language Processing - Final Report (Task 2)

Lara Kuhelj, Gregor Novak, Jan Pelicon, Tomaž Štrus

Abstract

The task of automatic extraction of semantic relationships and definitions has become easier in recent years with the popularity and success of the deep learning models, such as BERT which stands for bidirectional encoder representations from transformers. In our case, we used two different BERT architectures and trained few models for hypernym-hyponym extraction and extraction of other semantic relations. Our models were finetuned on the TermFrame dataset of Karstology provided by the linguists that worked with us in cooperation. We also used the relations from SemRelData dataset for additional comparison and evaluation.
The models successfully learned and were able predict the selected relations with good accuracy and overall score, even when there were plenty of different clasification classes. We also tried two different ways of automatic entity recognition that are then used for relationship extraction - spaCy and entity recognition with rules that were based on empirical observations of distribution of words and phrases. Both approaches fell short of our expectation. While the spaCy model were in some cases able to approximately locate wanted entities it often included other words that messed with the relationship extraction.

Keywords

Karstology, domain-specific, semantic relation extraction, hyponym-hypernym extraction.

Advisors: Slavko Žitnik

Introduction

Even nowadays, problems that originate from the field of natural language processing, usually don't have a simple solution, because of natural language's complexity and its complex word and sentence formation. In the past, there were mostly statistical methods being used to tackle such problems, but nowadays, they are being replaced by machine learning methods, deep neural networks and transformers that are now considered state of the art.

Our goal was to use one or more of these methods and models, train them on a few selected domain-specific corpora and extract hypernym-hyponym pairs. Moreover, based on the existing instances of domain-specific semantic relations, we should be able to discover new instances of semantic relations from provided Karst corpus and other explored datasets and corpora, with a focus on a few chosen relations.

First, we will take a brief overview of the datasets that we decided to use to complete the task. Then we will look at the transformer BERT models that we used for relation extraction.

Since our models were quite successfull at predicting these relations with predetermined entities, we also decided to give a try with implementing a model for automatic entity recognition for karstology domain.

Corpora and datasets

Besides the provided Karst corpus, we began with inspecting different corpora and datasets. We checked WikiData, NewYorkTimes (NYT) dataset and its subsets, NYT24 and NYT27. We also checked SloWnet and dataset form SemEval2010-Task8challenge. In the end, we decided to use the SemRelData dataset, because it has hypernym-hyponym pairs and is structurally similar to Karst. The linguists also provided us with the additional non-annotated karst corpus. Below we explore used datasets in more detail.

TermFrame - The Karst corpus

The TermFrame dataset contains relevant works in karstology in 3 languages – English, Croatian and Slovene. It includes books, reference works, PhD theses and scientific articles and other works related to Karst domain. It is available in plain text and chosen sentences were also manually annotated in WebAnno annotation tool [1] and are stored in .tsv file. Table 2 provides us with a brief overview of corpus by language. We chose to focus on English language since most datasets

are available in this language. There are four types of definition elements present in annotations, with the names and distribution visible on figure 1. Semantic relations and their distributions are visible on figure 2.

We also explored the distribution of words and phrases that are present in definition elements and semantic relations. For example, we prepared a small extract of words that are found in the annotated karst corpus. There is a small set of words that are highly likely to appear for certain types of relations. We later used this observation to derive certain rules for entity recognition.

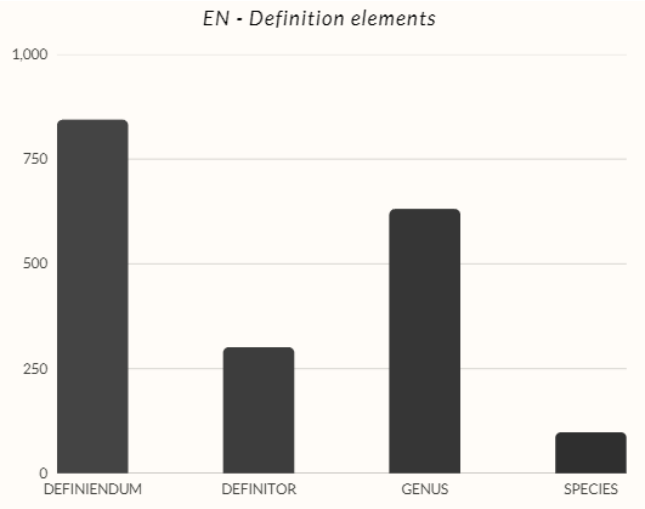

Figure 1. Number of definition elements by type for English in annotated definitions.

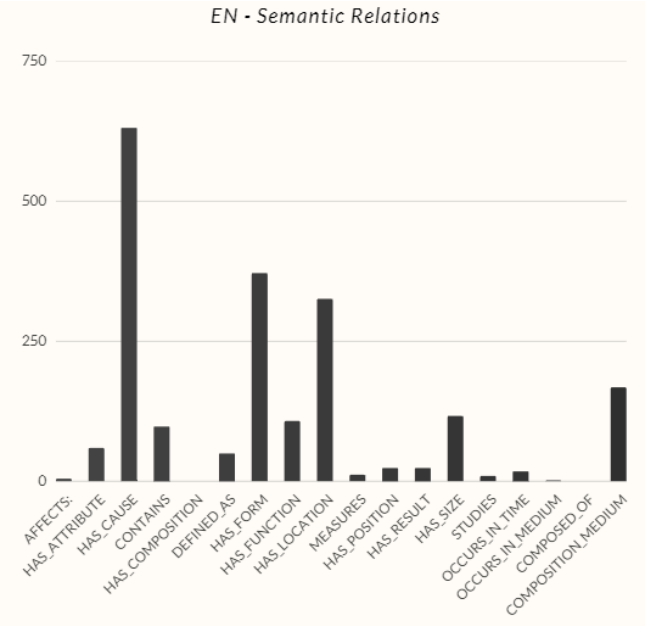

Figure 2. Number of semantic relations by relation type for English language in annotated definitions.

Table 1. Distribution of Semantic relation type for different languages (SemRelData dataset).

	English	German	Russian
Synonym	86	77	226
Co-Hyponym	281	336	296
Hypernym	553	508	296
Holonym	775	798	553

Table 2. Size of Karst corpus by language

	English	Slovene	Croatian
Tokens	2,386,075	1,208,240	1,229,368
Words	1,968,509	987,801	969,735
Sentences	87,713	51,990	53,017
Documents	54	60	43

SemRelData dataset

SemRelData stands for Semantic Relation Dataset which is focused on contextual annotation of classical semantic relations and provides data for English, German and Russian language. It consists of news articles, encyclopedic articles and snippets from literary texts for all three languages and contains ≈ 60.000 tokens, with the distribution of semantic relation types shown in the table 1. The dataset is annotated in WebAnno and can be downloaded from University of Hamburg site [2]. More detailed analysis was done by Darina Benikova [3].

As we mentioned before, this dataset is very similar to the Karst dataset. It also has hypernym-hyponym pairs, which we can use to test and evaluate the models that were trained on Karst.

Non-annotated Karst corpus

The additional corpus was provided to us by the students from the Faculty of Arts. It was meant to evaluate the performance of our models. The non-annotated corpus and was given in plaintext, so we used two preprocessing methods, to prepare the data for our model. Firstly, we used Punkt Sentence Tokenizer [4] from Natural Language Toolkit, which divides a text into a list of sentences by using an unsupervised algorithm to build a model for abbreviation words, collocations, and words that start sentences.

Although Punkt Sentence Tokenizer did manage to extract viable sentences from the whole corpus, there were still too many irrelevant sentences and cases which could not be used directly in BERT for relation classification. From this we extracted 46,604 sentences. Most of these sentences are not applicable for evaluation, so in the second step we filtered over the sentences. There were two steps of filtering, first we removed sentences that were too small or too large or had a specific structure (table of content, header, footers). Second, we used a collection of all words and phrases that occurred as definiendums in the original Karst corpus. We then only kept the sentences that contain any of the definiendum substrings, ex. “is a”, “are defined as”, “are made of”, etc. With this

method, we reduced the number of sentences to 12,143.

Automatic entity recognition

We tried to develop a tool for automatic entity recognition, which could extract entities that occur in a semantic relation. Firstly, we used spaCy, which is an existing tool for automatic entity recognition. Because of its poor performance on non-annotated Karst corpus, we also decided to try using custom rule based entity recognition.

spaCy

For automatic entity extraction, a spaCy model [5] was used before sentences were passed into BERT. The model did the automatic selection of entities, which were then appropriately tagged with E1 or E2.

The main problem of this approach was, that the model was recognizing entities that were often too long for any meaningful use in semantic relation extraction. This meant, that a lot of relations that were extracted from non-annotated corpus were either partly wrong or fully wrong.

Custom rules

We already mentioned in datasets section, that certain words appear more often in specific types of semantic relations. We tried to formulate a simple rules for determining if there is a semantic relationship or definition elements present in the sentence.

Let's take a look at *DEFINITOR* definition element for English. The top 5 definitors are:

1. Are (80 occurrences),
2. Is a (53 occurrences),
3. Is (18 occurrences),
4. Is defined as (11 occurrences),
5. The term (11 occurrences).

Together this sums up to 173 of 300 definitors. This means that by using just the top 5 definitors, we can already cover more than 50% of all definitors. This does not apply for other definition elements such as definiendum, which are mostly unique phrases. Using top 5 definiendums only covers around 4% of all definiendums.

Another problem is trying to figure out the correct order of entities. Is the definiendum located after or before the genus or species? There is often a case, that lots of words are hypernym and hyponym at the same time in relation to some other words. Therefore, it is not trivial to guess which one is correct. Though, we can assume that there is a high probability that the relation is present in the sentence if certain words are matched. This however is often not enough to correctly predict the presence of relations and find entities.

BERT

Bidirectional Encoder Representations from Transformers (BERT) is a NLP machine learning method based on transformers, proposed by Google [6]. Method is designed to pre-train deep bidirectional representations from unlabeled text. When pre-training is executed, method can then be fine-tuned and adapted for various specific tasks. Two BERT-based models were used in SemEval 2010 Task 8, both being one of the best performers. The task was as follows [7]: Given a sentence and two tagged nominals, to predict the relation between those nominals and the direction of the relation

Example:

- "There were apples, **pears** and oranges in the **bowl**."
- Relation: (content-container, pears, bowl)

Model usage for relation prediction

We used BERT model for the purpose of relation prediction in the given sequence, where the two entities from relation are marked with E1 and E2 tags. We used existing BERT implementation [8] and modified it for out domain-specific purposes. With E1 tags we marked first entities in the relation and with E2 tags we marked second entities. For example, if relation between the two entities is predicted as Hyponym-Hypernym relation, entity marked with E1 would be hyponym of entity E2, and the latter would be hypernym of entity E1.

First, we tried to fine-tune the BERT model on the SemRelData dataset, for the purpose of predicting Hypernym relations. We also tried fine-tuning the model on the part of annotated TermFrame dataset. Here we split the TermFrame annotated definitions in the training and testing set in ratio 4:1.

ALBERT

ALBERT, which was proposed by Zhenzhong Lan [9] stands for A Lite BERT and is a modified version of BERT NLP model. It builds on three key points such as Parameter Sharing, Embedding Factorization and Sentence Order Prediction (SOP). ALBERT is a much smaller model in terms of parameters, for example the large ALBERT model is 18 times smaller than the large BERT model.

Model usage for relation prediction

We used the ALBERT model in a similar way as we used BERT model. Again, we used and modified the existing model implementation. We then compared the performance of both resulting fine-tuned models and evaluated the results on the testing part of the TermFrame dataset, as well as manually annotated definitions of the 103 sequences from the additional corpus, provided by the colleagues from the Faculty of Arts.

Results

Befure using the models with automatic sentence and entity extraction, we tested them on the manually annotated results. We extracted E1 and E2 entities from the annotated definition

of each token/word. For example, if word in the sentence was annotated as part of definiendum, it was added to entity that was tagged as E1 and in this case, we considered the entity to be of type hyponym for the ground truth.

Fine-tuning on SemRelData dataset

For Hyponym-Hypernym extraction, we first tried to fine-tune the BERT model on the SemRelData dataset. Because out of all relation, that the SemRelData contains, only Hypernym relation is relevant to our tas, the purpose of the resulting fine-tuned BERT model would be the extraction of Hyponym-Hypernym relations from the corpus. Because of this, we only tested model's performance on the sequences containing E1 and E2 tags for Hyponym-Hypernym relations. For this we used the whole manually annotated dataset from TermFrame as the data was not used for fine-tuning the model. We considered phrases, annotated as definienda to be hyponyms, and phrases, annotated as genera, to be hypernyms. Accuracy of the model prediction are shown in the table 3.

Table 3. Results achieved on annotated Karst corpus by BERT model, fine-tuned with SemRelData dataset with 11 epochs. Ground truth for all 569 relations is Hyponym-Hypernym.

Detected relation	Number of detections	Percentage
Hyponym-Hypernym	414	72.76 %
Other	155	27.24 %
Total	569	100 %

Performance of the model, fine-tuned on the SemRelData was relatively good. But the con of this model was, that we could only use it for the Hyponym-Hypernym relations, since all the other relations that model was able to predict was not relevant for the Karst corpora. Because of this, any other predicted relation that was not Hyponym-Hypernym, was considered as wrong.

We then tried to fine-tune the BERT model on the part of annotated TermFrame dataset. We used 80 % of dataset for training and the remaining part of dataset for testing the models. Comparison between the model fine-tuned on SemRelData dataset and the model, fine-tuned on the training part of the TermFrame dataset is shown in the table 4. Better results are written in bold. Performance was measured on the sequences from TermFrame, that we used as testing split.

Table 4. Comparison between the BERT model, fine-tuned with SemRelData, and BERT model, fine-tuned with part of TermFrame dataset. Ground truth for all 101 relations is Hyponym-Hypernym.

	Fine-tuned on SemRelData		Fine-tuned on TermFrame	
Detected relation	Num. of detections	Perc.	Num. of detections	Perc.
Hyponym-Hypernym	94	93.07 %	**99**	**98.02** %
Other	7	6.93 %	**2**	**1.98** %
Total	101	100 %	101	100 %

Because performance of the model, fine-tuned on TermFrame training split, was slightly better, we decided to continue with fine-tuning on the TermFrame dataset only. We also tried to create ALBERT model, fine-tuned on the mentioned part of the TermFrame dataset, used as training part.

Comparison between BERT and ALBERT

Because ALBERT model is significantly smaller than the traditional BERT, it was able to achieve better training accuracy and loss in smaller amount of training epochs. In order to achieve accuracy, greater than 95 % and loss below 0.01 %, ALBERT model needed only 13 epochs, while BERT model's accuracy after 50 epochs was slightly above 90 % and its loss around 40 %.

First we compared both resulting model on the part of the TermFrame dataset, we used for testing. We decided to consider only predictions for Hyponym-Hypernym relation and 4 other relations, for which model was best performing and number of relation occurrences was high enough on the testing data that the evaluation could be considered as relevant. Predictions for all other relations in the corpus would be discarded and considered as if model is not equipped for their detection.

Comparison between precision, recall and f-score for selected 5 relations, as well as average scores of both models, is represented in table 5. Better results are written in bold.

Table 5. Comparison between the BERT and ALBERT model for selected 5 relations, fine-tuned with training part of TermFrame dataset. Evaluation is done on the testing part of the TermFrame dataset.

	BERT			ALBERT		
Relation	Precision	Recall	F-score	Precision	Recall	F-score
Hyponym-Hypernym	0.94	0.98	0.96	**1.0**	**0.984**	**0.99**
HAS_SIZE	0.76	0.695	0.73	**0.8**	**0.87**	**0.83**
HAS_CAUSE	0.65	0.72	0.69	**0.79**	**0.81**	**0.8**
HAS_FORM	0.61	0.67	0.64	**0.73**	**0.82**	**0.77**
HAS_LOCATION	0.58	0.71	0.64	**0.74**	**0.79**	**0.76**
Average	0.71	0.758	0.73	**0.81**	**0.86**	**0.83**

Judging by the scores, both models performed well on the testing part of the annotated Karst sequences. ALBERT's performance was significantly better, probably because it was able to achieve better accuracy and smaller loss in the fine-tuning process.

Before using model on the additional non-annotated corpus, provided by the Faculty of Arts, we tried the two models on the annotated definitions for the mentioned additional corpus in the same way we tried it on the testing part of the TermFrame annotations. With this, we wanted to see how well the model performs on the new dataset domain. Results for the same 5 selected relations are shown in table 6. Better results are written in bold.

Again, ALBERT performed better than BERT. But we can see, that performance is overall worse. The most drastic negative change is in the precision score, while recall mostly stayed relatively high. Because of the lower precision scores, f-scores are also affected.

Table 6. Comparison between the BERT and ALBERT model for selected 5 relations, fine-tuned with training part of TermFrame dataset. Evaluation is done on the manual annotated definitions, provided for the additional corpus.

	BERT			ALBERT		
Relation	Precision	Recall	F-score	Precision	Recall	F-score
Hyponym-Hypernym	0.93	0.98	0.95	**1.0**	**0.98**	**0.99**
HAS_SIZE	0.5	0.92	0.65	**0.55**	0.92	**0.69**
HAS_CAUSE	0.44	0.55	0.49	**0.64**	**0.72**	**0.68**
HAS_FORM	0.28	0.79	0.41	**0.35**	0.79	**0.49**
HAS_LOCATION	0.53	0.32	0.4	**0.68**	**0.61**	**0.64**
Average	0.54	0.71	0.58	**0.64**	**0.80**	**0.698**

Although performance is worse on the new domain, we argue that the model still performed adequately for most of the selected relations.

Because ALBERT's performance was better on both tested domains, we selected the fine-tuned ALBERT model as the final model of our work.

Using ALBERT with automatically extracted sentences and E1/E2 entities

After computational evaluation was done, we tried to automatically extract sentences from the non-annotated part of the additionally provided corpus. We then tried to automatically tag potential E1 and E2 entities for relation prediction in the resulting sentences.

Our method of manual sentence extraction appeared to be sub-optimal. Often, sentences were not split correctly, resulting in the sequences that were, for example, divided by comma instead of final punctuation. Additionally, automatic E1 and E2 tagging performed poorly. Most of the time, resulting entities contained tokens (words, punctuations, etc.) from half of the sequence, instead of consisting only of the relevant phrases, such as nouns or noun phrases for definiendums (hyponyms). This was the main bottleneck in our final solution, because ALBERT model's performance was relatively good on correctly extracted entities, it was significantly crippled by the low quality of extracted seqeneces and E1/E2 entities.

Linguist contribution

The main task in building an English corpus was collecting English texts on geomorphology, glaciology and geology that were, then, incorporated into the non-annotated Karst corpus created via Sketch Engine, ultimately consisting of 1,588,085 tokens.

Manual evaluation

Following the annotation process, the linguist wanted to analyze the predictions submitted by the model. This part proved to be quite burdensome, as the model was only able to indicate phrases containing a relation, but was unable to find it by itself. Another drawback of the model was the fact that it was not able to correctly separate the sentences, accounting for difficulties when attempting to locate full phrases. After examining the predictions, the linguist manually identified 20 sentences and annotated them according to hypernym-hyponym (genus-definiendum) relations. The team found that though the model had successfully recognized that all 20 sentences included hypernym-hyponym relations, it also listed many other sentences that did not contain such relations. Linguist annotation highlighted another obstacle: many of the manually annotated hypernyms and hyponyms were complex, containing more than one word, which likely confused the model when detecting definienda and genera. Ultimately, compared to manual annotations, model predictions proved to be insufficient and required thorough double-checking.

Conclusion

Relationship extraction models performed as expected and did, in some cases, even surpassed our expectation. Even with a short fine-tuning time, models were able to learn quickly and they can also be used interchangeably between two chosen datasets with decent performance. We compared the performance of two transformer models, BERT and ALBERT and concluded, that ALBERT is more fit to the task of relationship extraction.

However, we have to point out, that we were largely unsuccessful in achieving a good automatic entity recognition. It seems, that because of the Karst domain specifics, it is not as easy to suggest entities since they can contain very specific words, and also vary a lot.

In general, we accomplished the basic tasks. There is also a lot of space for improvement. With the knowledge we acquired during working on this project, we would definitely approach some things differently, such as the construction of a general pipeline for preparing training and testing data using different datasets.

References

[1] WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. Available at: https://webanno.github.io/webanno/. [Accessed: 20. 5. 2022].

[2] SemRelDataset. Available at: https://www.inf.uni-hamburg.de/en/inst/ab/lt/resources/data/semreldata.html. [Accessed: 20. 4. 2022].

[3] Darina Benikova and Chris Biemann. Semreldata — multilingual contextual annotation of semantic relations between nominals: Dataset and guidelines. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA).

[4] Punkt Sentance Tokenizer. Available at: https://www.nltk.org/_modules/nltk/tokenize/punkt.html. [Accessed: 24. 5. 2022].

[5] Spacy for entity Recognition. Available at: https://spacy.io/usage/linguistic-features. [Accessed: 20. 5. 2022].

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[7] Relationship Extraction | NLP-progress. Available at: http://nlpprogress.com/english/relationship_extraction.html. [Accessed: 18. 3. 2022].

[8] BERT relation Extraction. Available at: https://github.com/plkmo/BERT-Relation-Extraction. [Accessed: 24. 5. 2022].

[9] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2019.