FRI Natural language processing course 2021
Project report

University of Ljubljana
Faculty of Computer and
Information Science

Automatic semantic relation extraction

Kastriot Kadriu, Gal Lindič, Ožbej Golob

Abstract

Hyponym-hypernym extraction is an important task in NLP. Its potential application include information retrieval, information extraction, text summarization, machine translation and question answering. Most of the progress for this task has been done using pattern-based models. The issue with such models is that they are domain-specific constructed using language-specific rules. In this paper, we present to you a domain independent approach for identifying such relations.

Keywords

relation extraction, hyponym, hypernym, semantic relation

Advisors: Slavko Žitnik

Introduction

Relation extraction is the task of extracting semantic relations from a text and it plays a vital role in NLP tasks (question answering, information extraction, etc.) [1]. Relation classification is defined as predicting semantic relationship between two or more entities of a certain type (e.g. Person, Organisation, Location) in a sentence. Eg. given a sentance tagged with entity pairs (The <e1>highly corroded zone</e1> near the surface is termed the <e2>epikarst</e2> or subcutaneous zone), where one entity is *highly corroded zone* and the other is *epikrast*, we can classify their relation into *Entity-Origin(e2,e1)*. The classification models for relation classification rely on high-level lexical and syntactic features, such as part-of-speech (POS), named entity recognizer (NER) and word representations (Word2Vec, dependency parser). Such classifiers suffer from propagation of implicit error of the tools and are computationally expensive [2]. Recently, deep learning has made significant progress in natural language processing.

Hypernym extraction is a sub-problem of relationship extraction where the task is to identify hierarchical (hyponym-hypernym) relationships in the text (e.g. for the sentence *"Dog is an animal"*; *dog* is a hyponym and *animal* is a hypernym).

This paper aims to extract hierarchical (hyponym-hypernym) and non-hierarchical (domain-specific) relations from the TermFrame dataset [3]. The dataset consists of small domain-specific corpora and annotated definitions from the domain of Karstology, in three languages: English, Slovenian, Croatian.

Related work

Hypernym extraction relies on sentences where both hyponym and hypernym co-occur in characteristic contexts. Hearst [4] proposed lexical-syntactic patterns to extract hypernyms based on such contexts from the text.

A pattern-based approach investigates texts to find linguistic means involved in the production of semantic relations. This approach follows two assumptions, the first being that the target relation is a specific relation, which attempts to find general similarity relations between words [5, 6]. Such approaches are based on methods that follow Harris's distributional hypothesis. The second approach attempts to discover the meaning of implicitly expressed relations as found in multi-word expressions and noun-compounds [7, 8].

Snow [9] learned such patterns automatically based on a set of hyponym-hypernym pairs. They introduce a general-purpose formalization and generalization of these patterns. Given a training set of text containing known hypernym pairs, their algorithm automatically extracts useful dependency paths and applies them to new corpora to identify novel pairs.

Pantel and Pennacchiotti [10] presented a similar approach for harvesting semantic relations. Recent approaches explore the use of distributional word representations for extraction of hypernyms and co-hyponyms [11, 12].

Dai et al. [13] propose an approach to exploit semantic relations extracted from the sentence to improve the use of context. The used semantic relations are mainly those that are between the mention and the other words or phrases in the sentence. They investigate the use of two types of semantic relations: hypernym relation, and verb-argument relation. Their approach combines the predictions made based on different semantic relations and the predictions of a base neural model to produce the final results.

Shen and Huang [14] propose a novel attention-based convolutional neural network architecture for semantic relation extraction. Their model makes full use of word embedding, part-of-speech tag embedding and position embedding information. Word level attention mechanism is able to better determine which parts of the sentence are most influential with respect to the two entities of interest. This architecture enables learning some important features from task-specific labeled data, forgoing the need for external knowledge such as explicit dependency structures.

Lee et al. [2] propose a bidirectional LSTM network with entity-aware attention using Latent Entity Typing (LET). They propose a model that doesn't rely on the high-level lexical and syntatic features obtained by NLP tools such as WordNet, dependency parser, part-of-speech (POS) tagger, and named entity recognizers (NER). The proposed model also utilizes information of entity that may be the most crucial features for relation classification. The proposed model is a novel end-to-end recurrent neural model which incorporates an entity-aware attention mechanism with the LET method.

Zhou et al. [15] propose an attention-based bidirectional LSTM network for relation classification. The proposed netork is independent of lexical resources such as WordNet or NLP systems like dependency parser and NER. The proposed model accounts for the fact that important information can appear at any position in the sentence.

Wu and He [16] propose a model that both leverages the pre-trained BERT language model and incorporates information from the target entities to tackle the relation classification task. They locate the target entities and transfer the information through the pre-trained architecture and incorporate the corresponding encoding of the two entities.

Methods

0.1 Datasets

0.1.1 TermFrame

The TermFrame [3] dataset consists of small domain-specific corpora and annotated definitions from the domain of Karstology, in three languages: English, Slovenian, Croatian. The dataset contains relevant works in karstology, including books, reference works (encyclopaediae, glossaries), PhD theses and scientific articles. Definitions of karst terms have been automatically extracted and manually validated. The dataset contains 15 semantic relation types: AFFECTS, HAS_ATTRIBUTE, HAS_CAUSE, CONTAINS, HAS_COMPOSITION, DEFINED_AS, HAS_FORM, HAS_FUNCTION, HAS_LOCATION, MEASURES, HAS_POSITION, HAS_RESULT, HAS_SIZE, STUDIES, OCCURS_IN_TIME.

0.1.2 SemEval-2010 Task 8

The dataset for the SemEval-2010 Task 8 [17] is a dataset for multi-way classification of mutually exclusive semantic relations between pairs of nominals. The dataset contains 10 semantic relation types: Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection, Message-Topic, and Other.

0.2 Semantic relation extraction

The aim of our experiments is to build a domain-independent model that is able to detect the existence of a relation between pairs of entities and classify each detected relation type.

For this purpose, we implemented three state-of-the-art models: (i) Entity Attention Bi-LSTM [2], (ii) Attention Bi-LSTM [15], and (iii) R-BERT [16] and we also tried using some simple classifiers for which we extracted lexical and syntactical features.

From both SemEval and TermFrame dataset we extracted features (Entity-based, Word-based and Syntactic-based). Entity-based features consist of Entity1/2, root, POS and NER. Word based features are words in-between entities and word before e1 and after e2. Last the syntactic features consist of constituency parsing (NP, VP, PP, etc.) and shortest dependency path between root e1 and root e2. The next step was to encode the extracted features. POS and NER are categorical features, hence we converted them into numbers using binary encoding. For the word representations we used Word2Vec to present each word with a 100 dimensional vector and then we summed each word vector in a sentence into a single 100 dimensional vector. We trained our dataset over Decision tree, SVM and XGBoost.

To address these issues of implicit error of tools, [2] proposed a novel end-to-end recurrent neural model which incorporates an entity-aware attention mechanism with a latent entity typing (LET). To capture the context of sentences, we obtain word representations by self attention mechanisms and build the recurrent neural architecture with Bidirectional Long Short-Term Memory (LSTM) networks. Entity-aware attention focuses on the most important semantic information considering entity pairs with word positions relative to these pairs and latent types obtained by LET.

Deep learning methods provide an effective way of reducing the number of handcrafted features. However these approaches still use resources such as WordNet or NLP systems to get high-level features. Zhou [15] proposed a novel neural network called Attention Bidirectional LSTM from relation classification. The model uses neural attention mechanism with Bidirectional LSTM to capture the most important semantic information and it does not utilize any features derived from lexical resources or NLP systems. The main contribution of the model is that it automatically focuses on the words that have decisive effect on classification, without using any NLP systems.

[16] proposed a model that levaraged the pretrained BERT

language model and incorporated information from the target entities, for the purpose of relation classification.

First, we trained the models on 8000 sentences from the SemEval dataset. After training, we tested the models on 2717 sentences from the SemEval dataset. We also tested the models on the TermFrame dataset. However, the TermFrame and SemEval dataset contain different relation types. To tackle this problem, we mapped the TermFrame relations to SemEval relations. The HAS_CAUSE and HAS_RESULT were mapped to Cause-Effect; HAS_COMPOSITION was mapped to Content-Container and Entity-Origin; CONTAINS was mapped to Content-Container, Component-Whole, Member-Collection; and HAS_LOCATION was mapped to Entity-Origin. The remaining 10 TermFrame relation types were disregarded during testing.

0.3 Joint entity and relation extraction

In the previous iteration, we researched regarding the task of relation extraction. However for our cases just that is not enough. We have continued the research in the discipline of joint entity and relation extraction. More specifically, we have tasked the models with the job of finding what relations are present in the given sentence, the entities involved with them and if there are any overlapping entities shared by different relations.

Initially, we set up a basic Bert Model with the task of entity extraction. In our case, we experimented with extracting three entities; the hyponym, hypernym ones and others. We do not report quantitative results but from our experiments, we have noticed that the model is considerably good at predicting the hypernyms whereas for the rest the task of prediction was affected by the quality (and quantity) of data. The work with this model was discontinued since the other relations we need to predict need more detailed modelling and cannot represented by the current one. Nevertheless, this model served as good starting point for understanding the role of BERT as an encoder and further fine-tunings that can be done with it.

The best results were obtained using the SPN model [18]. This state-of-the-art model treats the extraction of relational triples as a set prediction problem. This model architecture combines a BERT encoder with a non-autoregressive decoder. The data used for training and validation is a combination of a small subset of the TermFrame data, WCL data, and the provided WebNLG dataset. The former two datasets have only one type of relation mapped per sentence (hyponym-hypernym) whereas the latter is a larger dataset consisting of various multiple relations per sentence. The test dataset solely consist of TermFrame data unseen during training or validation. The train dataset size is 3000, the validation one is 300 and the testing one is 500. The detailed data markup is depicted in Table 1.

The aim of our such markup here was to have the results unbiased by not having hyponym/hypernym pair as the majority pair during training.

Dataset	Train	Validation	Test
TermFrame	110	130	449
WCL	30	0	0
OpenNLG	2870	170	0

Table 1. Training, validation and testing data markup for the SPN model.

Results

0.4 Semantic relation extraction

Table 2 shows F1 measures of implemented semantic relation extraction models on SemEval and TermFrame datasets. We also report the F1 score that authors reported in their respective articles on the SemEval dataset.

Model	SemEval f1	TermFrame f1
R-BERT	80,3	/
Entity Attention Bi-LSTM	76,3	0,14
Attention Bi-LSTM	74,3	0,138
SVM	0,471	0,01
Decision tree	0,497	0,03
XGBoost	0,63	0,021

Table 2. Results of relation extraction on SemEval and TermFrame.

0.5 Joint entity and relation extraction

We report quantitative results for the SPN model. We have trained the model for 4 epochs. When evaluated on the test data, the model has high accuracies for predicting the relation present in the sentence (excelling in 99% of cases) but it performs poorly at detecting the entities between which the relation is present, excelling in only 30% of the cases. The entity extraction results refer to the exact match - that is both extracted sequences describing the entities present in a relation match exactly the ones from the ground truth represented by their head and tail words signifying the start and the end of such sequences. Detailed results are depicted in Table 3.

Discussion

The results from the SPN model are promising. The model was initially trained in two different large datasets holding relations different from ours. SPN was successful in detecting relations and extracting entities in both of these use-cases (in more than 90% of the cases) so the results from Table 3 should

Task	Precision	Recall	F1	# of corr. pred.
Relation extraction	0.9977	0.9955	0.9966	448
Entity extraction	0.3415	0.3407	0.3411	153

Table 3. Results of joint entity and relation extraction using the SPN model tested on TermFrame data.

not be surprising. The weak results at detecting entities from the TermFrame data should come from the specific nature of the domain. That could be solved with more training data, however we will not be necessarily counting on that. The next step of our research is going to be concerned with new ways of encoding the data and including more relations in there as to be able of making more predictions. We will also be exploring different parameters for the model as the current tests have been isolated in that perspective. Detailed evaluation metrics will also be described.

Acknowledgments

References

[1] Thien Huu Nguyen and Ralph Grishman. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st workshop on vector space modeling for natural language processing*, pages 39–48, 2015.

[2] Joohong Lee, Sangwoo Seo, and Yong Suk Choi. Semantic relation classification via bidirectional lstm networks with entity-aware attention using latent entity typing. *Symmetry*, 11(6):785, 2019.

[3] TermFrame project. https://termframe.ff.uni-lj.si/. Accessed: 2022-03-15.

[4] Marti A Hearst. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*, 1992.

[5] Ido Dagan, Shaul Marcus, and Shaul Markovitch. Contextual word similarity and estimation from sparse data. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 164–171, 1993.

[6] Dekang Lin. Automatic retrieval and clustering of similar words. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pages 768–774, 1998.

[7] Roxana Girju, Adriana Badulescu, and Dan Moldovan. Learning semantic constraints for the automatic discovery of part-whole relations. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 80–87, 2003.

[8] Roxana Girju, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. Classification of semantic relations between nominals. *Language Resources and Evaluation*, 43(2):105–121, 2009.

[9] Rion Snow, Daniel Jurafsky, and Andrew Ng. Learning syntactic patterns for automatic hypernym discovery. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004.

[10] Patrick Pantel and Marco Pennacchiotti. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the Association for Computational Linguistics*, pages 113–120, 2006.

[11] Stephen Roller, Katrin Erk, and Gemma Boleda. Inclusive yet selective: Supervised distributional hypernymy detection. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pages 1025–1036, 2014.

[12] Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, 2014.

[13] Hongliang Dai, Yangqiu Song, and Xin Li. Exploiting semantic relations for fine-grained entity typing. In *Automated Knowledge Base Construction*, 2020.

[14] Yatian Shen and Xuan-Jing Huang. Attention-based convolutional neural network for semantic relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2526–2536, 2016.

[15] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany, August 2016. Association for Computational Linguistics.

[16] Shanchan Wu and Yifan He. Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 2361–2364, 2019.

[17] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[18] Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, Xiangrong Zeng, and Shengping Liu. Joint entity and relation extraction with set prediction networks. *arXiv preprint arXiv:2011.01675*, 2020.