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, Jasna Cindrič

Abstract

Hyponym-hypernym extraction is an important task in NLP. Its potential applications 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 and constructed using language-specific rules. Relation classification is also an important NLP task to extract relations between entities. It relies on the information of both the sentence and the two target entities. The state-of-the-art methods for relation classification are primarily based on CNN or RNN. In this paper, we experimented with different state-of-the-art approaches for both tasks and we also present a domain-independent approach for identifying hyponym-hypernym pairs and semantic relations as well. For semantic relation extraction, we used the SemEval and Termframe (Slovene and English) datasets, while for the hyponym-hypernym task we used WCL and OpenNLG datasets besides the TermFrame dataset. The domain-independent models were manually evaluated on the English TermFrame corpus.

Keywords

relation extraction, hyponym, hypernym, semantic relation, domain-independent

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. Given a sentence tagged with entity pairs (e.g. *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 the 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, and 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.

To address the 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 a 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 a decisive effect on classification, without using any NLP systems.

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.

Baldini Soares et al. [17] present domain-independent relation representations solely from entity-linked text. These representations significantly outperform previous work on supervised relation extraction datasets like SemEval 2010 Task 8.

Datasets

TermFrame

The TermFrame [3] dataset consists of small domain-specific corpora and annotated definitions from the domain of Karstology, in three languages: English, Slovenian, and Croatian. The dataset contains relevant works in karstology, including books, reference works (encyclopediae, 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.

SemEval-2010 Task 8

The dataset for the SemEval-2010 Task 8 [18] 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.

English Corpus of Geology, Glaciology and Geomorphology

The data provided for automatic extraction was made-up of an English corpus that contained texts from similar domains, namely geology, glaciology and geomorphology. The 26 English texts were gathered from scientific articles and books and, if needed, converted from .pdf into .txt or .docx files. These files were then uploaded to Sketch Engine, where a corpus containing 1,588,085 tokens was created. To provide a basis for automatic extraction, 104 definitions from the English corpus were extracted from Sketch Engine and manually annotated in WebAnno using the above-mentioned 15 relations.

WebNLG

The WebNLG corpus consists of sentences defined altogether with up to 7 triples describing two entities and the relationship between them. The relations between entities aren't grammatical. The total dataset is composed of 6500 entries but only a subset of them have been used for our experiments.

WCL

WCL dataset has originally been used to perform experimental evaluations in Word-class Latices. The dataset we have used contains around 3000 sentences that have the hyponym-hypernym relation defined between entities. Only a subset of the dataset has been used for our experiments.

Methods

Semantic relation extraction

Data preprocessing

From both SemEval and TermFrame datasets, we extracted Entity-based, Word-based, and Syntactic-based features. Entity-based features consist of Entity1/2, root, POS, and NER. Word-based features are words in-between entities and a word before e1 and after e2. The syntactic features consist of constituency parsing (NP, VP, PP, etc.) and the shortest dependency path between root e1 and root e2. After feature extraction, we encoded 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.

Models

The aim of this paper is to build a domain-independent model that is able to classify the semantic relation type from the sentence. For this purpose, we implemented two types of semantic relation extraction models that differ in the form of the input data. The input of the first type of models are sentences tagged with entity pairs (e.g. *[E1]Dolomite[/E1] is a type of [E2]rock[/E2] that dissolves and is karstified similarly as limestone.*). The input of the second type of models are non-annotated sentences (e.g. *Dolomite is a type of rock that dissolves and is karstified similarly as limestone.*).

We implemented three state-of-the-art models that take sentences tagged with entity pairs as input: (i) Entity Attention Bi-LSTM [2], (ii) R-BERT [16], and (iii) Match-The-Blanks BERT (MTB-BERT) [17]. We also implemented some simple classifiers (SVM, Decision tree, and XGBoost) for which we extracted lexical and syntactical features. One of the drawbacks of these classifiers is, that they match the relation irrespective of direction. We also modified the MTB-BERT to accept non-annotated sentences as input.

For the MTB-BERT and R-BERT model, we used a pre-trained multi-layer bidirectional transformer encoder presented by Devlin et al. [19]. We trained other 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 datasets 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.

We also trained the same models on 1105 sentences (we oversampled them from the initial 580 sentences) from the TermFrame dataset. We only used the sentences with the six most frequent relations: HAS_CAUSE, HAS_RESULT, HAS_FORM, HAS_LOCATION, HAS_ATTRIBUTE, DEFINED_AS. After training, we tested the models on 74 sentences from the TermFrame dataset.

Due to the small sample size of annotated TermFrame definitions, we used the transfer learning approach to improve the efficiency of some of our models (R-BERT, Entity Attention Bi-LSTM, and MTB-BERT). First, we trained the models on a larger dataset to gain the knowledge while solving one problem and then applied it to a TermFrame dataset which is a different but relatable problem. This approach proved to be valuable since it greatly increased our models' efficiency when training on the TermFrame dataset.

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 modeling and cannot be represented by the current one. Nevertheless, this model served as a 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 [20]. 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 consists 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.

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.

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.

Results

Table 2 shows F1 measures of implemented semantic relation extraction models (that take sentences tagged with entity pairs as input) trained on the SemEval dataset and evaluated on SemEval and TermFrame datasets.

Model	SemEval F1	TermFrame F1
R-BERT	0.871	0.014
Entity Attention Bi-LSTM	0.763	0.140
SVM	0.471	0.010
Decision tree	0.497	0.030
XGBoost	0.630	0.021

Table 2. Results of semantic relation extraction models trained on SemEval and evaluated on SemEval and TermFrame.

Table 3 shows F1 measures of implemented semantic relation extraction models (that take sentences tagged with entity pair as input) trained on the TermFrame dataset and evaluated on the English and Slovene TermFrame corpus.

Model	EN F1	SL F1
R-BERT	0.336	0.118
Entity Attention Bi-LSTM	0.248	0.249
SVM	0.131	0.187
Decision tree	0.184	0.236
XGBoost	0.170	0.326

Table 3. Results of semantic relation extraction models trained and evaluated on TermFrame for English and Slovene corpus.

Table 4 shows F1 measurements of implemented semantic relation extraction models (that take sentences tagged with entity pair as input) trained on: (a) the SemEval dataset and (b) the CNN dataset. The models were evaluated on the TermFrame dataset.

	Model	EN F1	SL F1
(a)	R-BERT	0.632	0.621
	Entity Attention Bi-LSTM	0.413	0.381
(b)	MTB-BERT	0.585	/

Table 4. Results of semantic relation extraction models on the TermFrame dataset when transfer learning approach was used to pretrain them on: (a) the SemEval dataset and (b) the CNN dataset. The models were evaluated on the TermFrame dataset.

We also tested the MTB-BERT model on non-annotated sentences from TermFrame and achieved an F1 score of 0.473.

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 presence 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 5.

Task	Precision	Recall	F1	# Corr. pred.
Relation detection	0.998	0.996	0.997	448
Relation and feature extraction	0.342	0.341	0.341	153

Table 5. Results of joint entity and relation extraction using the SPN model tested on TermFrame data.

Manual evaluation results

In order to perform an independent predictions analysis, we employed some of our models on a previously non-annotated section of the TermFrame dataset. For the hypernym-hyponym relation, we used the SPN4RE model, and for semantic relation extraction we used the MTB-BERT model.

The manual evaluation process consisted of an evaluation of 30 sentences of relation predictions and 20 sentences of definition elements predictions that were extracted from the English corpus of geology, glaciology and geomorphology. For relations, the model predicted only the relation type and not the position or phrase that pertained to the specific relation type. The manual evaluation was thus focused only on determining whether the sentence contained the predicted relation. From the 30 sentences that were evaluated, there were 11 HAS_FORM relations (10 of which were accurately predicted), 11 HAS_CAUSE relations (8 of which were accurately predicted), 4 DEFINED_AS relations (1 of which was accurately predicted), 3 HAS_LOCATION relations (2 of which were accurately predicted) and 1 HAS_RESULT relation (which was accurately predicted). Consequently, the model accurately predicted 22 relations out of 30, making the accuracy of the model 0.73 (see Table 6).

Definition elements that were evaluated consisted of the DEFINIENDUM and GENUS elements the model automatically predicted, whereby the exact phrases had been also determined. In the 20 sentences that were evaluated, the model found 21 definiendum elements and 30 genus elements. Each accurately predicted word was first separately marked as correct, and then the accuracy of the whole predicted phrase

Relation type	All predictions	Correct predictions
DEFINED_AS	4	1
HAS_FORM	11	10
HAS_CAUSE	11	8
HAS_RESULT	1	1
HAS_LOCATION	3	2
Total	30	22

Table 6. Manual evaluation results of relation type predictions.

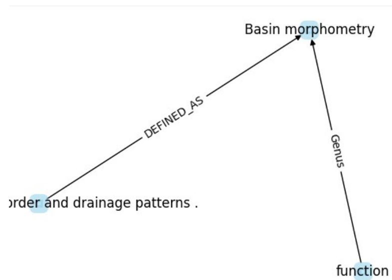

Figure 2. A part of the semantic relation network.

was evaluated as well. The model was partially accurate for all definiendum elements and for 27 genus elements. However, when evaluating the whole phrase, the model incorrectly predicted additional words that followed or preceded the definition element in almost all cases (for example, 14 predicted definiendums also contained the verb "is"). In 7 out of 20 cases, the whole definiendum phrase was accurately predicted, and for the genus elements, the model was correct in 10 out of 30 cases (see Table 7). The reported precision score of the SPN4RE model by manual evaluation is **0.73**.

Definition element	# of elements predicted	Correct phrase
Definiendum	21	7
Genus	30	10

Table 7. Manual evaluation results for definition element predictions.

Visualization of semantic network

We made a visualization of the network for semantic relation extraction (see Figure 1). The nodes present definiendum, genus, and the subsentence that connects them. The edges contain ground truth relations. In case of the correct prediction we denoted the edge with black color. If the classifier failed to predict the correct relation we denoted the edge with red color.

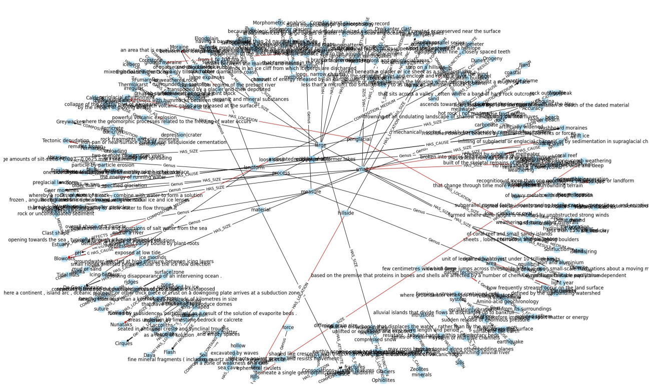

Figure 1. The whole semantic relations network.

Figure 2 shows a part of the semantic relation network.

Discussion

The results from the SPN4RE 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 5 should not be surprising.

Initially, the semantic relation extraction models did not perform well when they were trained on the SemEval and TermFrame datasets. We improved the models quite a bit by utilizing transfer learning where we loaded the weights of the model trained on SemEval and CNN datasets. Of all the used models, MTB-BERT seems the most promising, since it can predict relations on non-annotated sentences with an F1 score of 0.473. For the annotated sentences R-BERT performed the best (with an F1 score of 0.623) when transfer learning approach was used to pre-train on SemEval dataset.

From the preceding section we could see how the SPN4RE model did well at extracting features describing either the definiendum or genus elements but had issues with the exact matches where it included irrelevant elements such as the definitor *is* mostly in the definiendum part. Moreover, the model was able to include other relations in predictions but at a lower number and this was heavily influenced by the dominance of hyponym-hypernym relations in the training data and the lack of the rest.

In the future, the work should be oriented towards a more diverse markup of data that shall cover various relations and ways those relations are connected. To make those models work better with languages different from English, in which they are originally trained, one should also improve the encoders which are not that successful with less popular languages.

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] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy, July 2019. Association for Computational Linguistics.

[18] 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.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] 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.