

MIT at SemEval-2017 Task 10: Relation Extraction with Convolutional Neural Networks

Ji Young Lee*
MIT
jjylee@mit.edu

Franck Dernoncourt*
MIT
francky@mit.edu

Peter Szolovits
MIT
psz@mit.edu

Abstract

Over 50 million scholarly articles have been published: they constitute a unique repository of knowledge. In particular, one may infer from them relations between scientific concepts. Artificial neural networks have recently been explored for relation extraction. In this work, we continue this line of work and present a system based on a convolutional neural network to extract relations. Our model ranked first in the SemEval-2017 task 10 (ScienceIE) for relation extraction in scientific articles (subtask C).

1 Introduction and related work

The number of articles published every year keeps increasing (Druss and Marcus, 2005; Larsen and Von Ins, 2010), with well over 50 million scholarly articles published so far (Jinha, 2010). While this repository of human knowledge contains invaluable information, it has become increasingly difficult to take advantage of all available information due to its sheer amount.

One challenge is that the knowledge present in scholarly articles is mostly unstructured. One approach to organize this knowledge is to classify each sentence (Kim et al., 2011; Amini et al., 2012; Hassanzadeh et al., 2014; Dernoncourt et al., 2016). Another approach is to extract entities and relations between them, which is the focus of the ScienceIE shared task at SemEval-2017 (Augenstein et al., 2017).

Relation extraction can be seen as a process comprising two steps that can be done jointly (Li and Ji, 2014) or separately: first, entities of interest need to be identified, and second, the relation among the entities has to be determined. In

this work, we concentrate on the second step (often referred to as relation extraction or classification) and on binary relations, i.e. relations between two entities. Extracted relations can be used for a variety of tasks such as question-answering systems (Ravichandran and Hovy, 2002), ontology extension (Schutz and Buitelaar, 2005), and clinical trials (Frunza and Inkpen, 2011).

In this paper, we describe the system that we submitted for the ScienceIE shared task. Our system is based on convolutional neural networks and ranked first for relation extraction (subtask C).

Existing systems for relation extraction can be classified into five categories (Zettlemoyer, 2013): systems based on hand-built patterns (Yangarber and Grishman, 1998), bootstrapping methods (Brin, 1998), unsupervised methods (Gonzalez and Turmo, 2009), distant supervision (Snow et al., 2004), and supervised methods. We focus on supervised methods, as the ScienceIE shared task provides a labeled training set.

Supervised methods for relation extraction commonly employ support vector machines (Uzuner et al., 2010, 2011; Minard et al., 2011; GuoDong et al., 2005), naïve Bayes (Zayaraz and Kumara, 2015), maximum entropy (Sun and Grishman, 2012), or conditional random fields (Sutton and McCallum, 2006). These methods require the practitioner to handcraft features, such as surface, lexical, syntactic features (Grouin et al., 2010) or features derived from existing ontologies (Rink et al., 2011). The use of kernels based on dependency trees has also been explored (Bunescu and Mooney, 2005; Culotta and Sorensen, 2004; Zhou et al., 2007).

More recently, a few studies have investigated the use of artificial neural networks for relation extraction (Socher et al., 2012; Nguyen and Grishman, 2015; Hashimoto et al., 2013). Our approach follows this line of work.

* These authors contributed equally to this work.

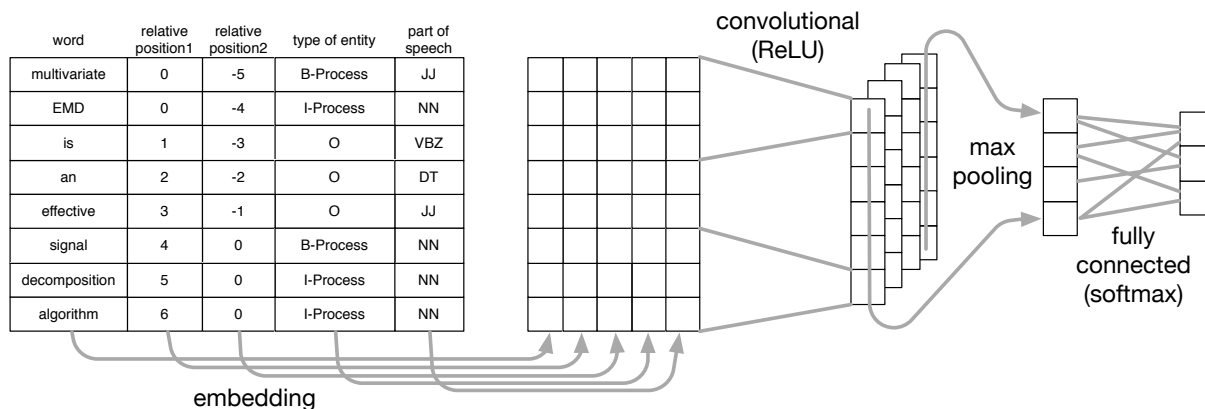


Figure 1: CNN architecture for relation extraction. The left table shows an example of input to the model.

Examples	Rule format	Relations detected
transmission electron microscopy (TEM)	A (B)	If B is an abbreviation of A, then A and B are synonyms.
high purity standard metals (Sn, Pb, Zn, Al, Ag, Ni)	A (B, C, ..., Z)	If any of B, C, ..., Z is a hyponym of A, then all of them are hyponyms of A.
(TEMs), scanning electron microscopes	(A) B	A and B have no relation.
DOTMA/DOPE	A/B	A and B have no relation.

Table 1: Rules used for postprocessing. We considered B to be an abbreviation of A if the first letters of each token in A form B. The examples are from the training and development sets

2 Model

Our model for relation extraction comprises three parts: preprocessing, convolutional neural network (CNN), and postprocessing.

2.1 Preprocessing

The preprocessing step takes as input each raw text (i.e., a paragraph of a scientific article in ScienceIE) as well as the location of all entities present in the text, and output several examples. Each example is represented as a list of tokens, each with four features: the relative positions of the two entity mentions, and their entity types and part-of-speech (POS) tags. Figure 1 shows an example from the ScienceIE corpus in the table on the left.

Sentence and token boundaries as well as POS tags are detected using the Stanford CoreNLP toolkit (Manning et al., 2014), and every pair of entity mentions of the same type within each sentence boundary are considered to be of a potential relation. We also remove any references (e.g. [1, 2]), which are irrelevant to the task, and ensure that the sentences are not too long by eliminating the tokens before the beginning of the first entity mention and after the end of the second entity mention.

2.2 CNN architecture

The CNN takes each preprocessed sentence as input, and predicts the relation between the two entities. The CNN architecture, illustrated in Figure 1, consists of four main layers, similar to the one used in text classification (Collobert et al., 2011; Kim, 2014; Lee and Deroncourt, 2016; Gehrmann et al., 2017).

1. the **embedding layer** converts each feature (word, relative positions 1 and 2, type of entity, and POS tag) into an embedding vector via a lookup table and concatenates them.
2. the **convolutional layer** with ReLU activation transforms the embeddings into feature maps by sliding filters over the tokens.
3. the **max pooling layer** selects the highest feature value in each feature map by applying the max operator.
4. the **fully connected layer** with softmax activation outputs the probability of each relation.

2.3 Rule-based postprocessing

The postprocessing step uses the rules in Table 1 to correct the relations detected by the CNN, or to detect additional relations. These rules were developed from the examples in the training set, to be consistent with common sense.

annotation	A (arg1) is a Hyponym of (rel) B (arg2)					
order in text	... A ... B B ... A ...		
strategy	rel	arg1	arg2	rel	arg1	arg2
correct order	Hypo	A	B	Hypo	A	B
correct order	Hypo	A	B	Hypo	A	B
w/ neg. smpl.	None	B	A	None	B	A
fixed order	Hypo	A	B	Hyper	B	A
any order	Hypo	A	B	Hyper	A	B
	Hyper	B	A	Hypo	B	A

annotation	A (arg1) is a Synonym of (rel) B (arg2)					
order in text	... A ... B B ... A ...		
strategy	rel	arg1	arg2	rel	arg1	arg2
correct order	Syn	A	B	Syn	A	B
correct order	Syn	A	B	Syn	A	B
w/ neg. smpl.	Syn	B	A	Syn	B	A
fixed order	Syn	A	B	Syn	B	A
any order	Syn	A	B	Syn	A	B
	Syn	B	A	Syn	B	A

Table 2: Argument ordering strategies. “w/ neg. smpl.”: with negative sampling (Xu et al., 2015), “rel”: relation, “arg”: argument. “Syn”, “Hypo”, “Hyper”, and “None” refers to the “Synonym-of”, “Hyponym-of”, “Hypernym-of”, and “None” relations. Note that the “Hypernym-of” relation is the reverse of the “Hyponym-of” relation, introduced in addition to the relations annotated for the dataset.

2.4 Implementation

During training, the objective is to maximize the log probability of the correct relation type. The model is trained using stochastic gradient descent with minibatch of size 16, updating all parameters, i.e., token embeddings, feature embeddings, CNN filter weights, and fully connected layer weights, at each gradient descent step. For regularization, dropout is applied before the fully connected layer, and early stop with a patience of 10 epochs is used based on the development set.

The token embeddings are initialized using publicly available² pre-trained token embeddings, namely GloVe (Pennington et al., 2014) trained on Wikipedia and Gigaword 5 (Parker et al., 2011). The feature embeddings and the other parameters of the neural network are initialized randomly.

To deal with class imbalance, we upsampled the synonym and hyponym classes by duplicating the examples in the positive classes so that the *upsampling ratio*, i.e., the ratio of the number of positive examples in each class to that of the negative examples, is at least 0.5. Without the upsampling, the trained model would have poor performances.

3 Experiments

3.1 Dataset

We evaluate our model on the ScienceIE dataset (Augenstein et al., 2017), which consists of 500 journal articles evenly distributed among the domains of computer science, material sciences and physics. Three types of entities are annotated: process, task, and material. The relation between each pair of entities of the same type within a sentence are annotated as either “Synonym-of”,

² <http://nlp.stanford.edu/projects/glove/>

“Hyponym-of”, or “None”. Table 3 shows the number of examples for each relation class.

Relation	Train	Dev	Test
Hyponym-of	420	123	95
Synonym-of	253	45	112
None	5355	1240	1503
Total	6028	1408	1710

Table 3: Number of examples for each relation class in ScienceIE. “Dev”: Development.

3.2 Hyperparameters

Table 4 details the experiment ranges and choices of hyperparameters. The results were quite robust to the choice of hyperparameters within the specified ranges.

Hyperparameter	Choice	Experiment range
Token embedding dim.	100	50 – 300
Feature embedding dim.	10	5 – 50
CNN filter height	5	3 – 15
Number of CNN filters	200	50 – 500
Dropout probability	0.5	0 – 1
Upsampling ratio	3	0.5 – 5

Table 4: Experiment ranges and choices of hyperparameters.³

3.3 Argument ordering strategies

One of the main challenges in relation extraction is the ordering of arguments in relations, as many relations are *order-sensitive*. For example, consider the sentence “A dog is an animal.” If we set “dog” to be the first argument and “animal” the second,

³For these experiments, we used the official training set as the training/development set with a 75%/25% split, and the official development set as the test set.

Labels used	Training strategy	Evaluation strategy	Hyponym-of			Synonym-of			Micro-averaged		
			P	R	F1	P	R	F1	P	R	F1
All	correct order	any order	0.193	0.101	0.132	0.782	0.640	0.703	0.409	0.245	0.306
	corr. w/ n. s.	any order	0.431	0.127	0.196	0.826	0.756	0.788	0.638	0.295	0.404
	any order	any order	0.482	0.197	0.279	0.784	0.756	0.769	0.621	0.346	0.444
	any order	fixed order	0.486	0.195	0.278	0.773	0.753	0.763	0.621	0.345	0.443
	fixed order	any order	0.372	0.218	0.274	0.743	0.756	0.749	0.516	0.362	0.425
	fixed order	fixed order	0.425	0.213	0.283	0.803	0.753	0.777	0.578	0.358	0.441
Hyponym	correct order	any order	0.108	0.069	0.084	-	-	-	-	-	-
	corr. w/ n. s.	any order	0.215	0.115	0.148	-	-	-	-	-	-
	any order	any order	0.384	0.246	0.299	-	-	-	-	-	-
	any order	fixed order	0.410	0.235	0.298	-	-	-	-	-	-
	fixed order	any order	0.385	0.249	0.301	-	-	-	-	-	-
	fixed order	fixed order	0.409	0.237	0.297	-	-	-	-	-	-
Synonym	any order	any order	-	-	-	0.855	0.771	0.811	-	-	-
	any order	fixed order	-	-	-	0.852	0.776	0.812	-	-	-
Hyp+Syn	any + any	any + fixed	0.385	0.228	0.285	0.857	0.771	0.812	0.553	0.373	0.445

Table 5: Results for various ordering strategies on the development set of the ScienceIE dataset, averaged over 10 runs each.³ “corr. w/ n. s.”: correct order with negative sampling. Hyp+Syn is obtained by merging the output of the best hyponym classifier and that of the best synonym classifier.

then the corresponding relation is “Hyponym-of”; however, if we reverse the argument order, then the “Hyponym-of” relation does not hold any more.

Therefore, it is crucial to ensure that 1) the CNN is provided with the information about the argument order, and 2) it is able to utilize the given information efficiently. In our work, the former point is addressed by providing the CNN with the two relative position features compared to the first and the second argument of the relation respectively. In order to certify the latter point, we experimented with four strategies for argument ordering, outlined in Table 2.

4 Results and Discussion

Table 5 shows the results from experimenting with various argument ordering strategies. The correct order strategy performed the worst, but the negative sampling improved over it slightly, while the fixed order and any order strategies performed the best. The latter two strategies performed almost equally well in terms of micro-averaged F1-score. This implies that for relation extraction it may be advantageous to use both the original relation classes as well as their “reverse” relation classes for training, instead of using only the original relation classes with the “correct” argument ordering (with or without the negative sampling). Moreover, ordering the argument as the order of appearance in the text and training once per relation

(i.e., fixed order) is as efficient as training each relation as two examples in two possible argument ordering, one with the original relation class and the other with the reverse relation class (i.e., any order), despite the small size of the dataset.

The difference in performance between the correct order versus the fixed or any order strategies is more prominent for the “Hyponym-of” relation than for the “Synonym-of” relation. This is expected, since the argument ordering strategy is different only for the order-sensitive “Hyponym-of” relation. It is somewhat surprising though, that the correct order strategy performs worse than the other strategies even for order-insensitive “Synonym-of” relation. This may be due to the fact that the model does not see any training examples with the reversed argument ordering for the “Synonym-of” relation. In comparison, the negative sampling strategy, which learns from both the original and reversed argument ordering for the “Synonym-of” relation, the performance is comparable to the two best performing strategies.

We have also experimented with different evaluation strategies for the models trained with the any order and fixed order strategies. When the model is trained with the any order strategy, the choice of the evaluation strategy does not impact the performance. In contrast, when the model is trained with the fixed order strategy, it performs better if the same strategy is used for evaluation. This may

be the reason why the model trained with the correct order strategy does not perform as well, since it has to be evaluated with a different strategy from training, namely the any order strategy, as we do not know the correct ordering of arguments for examples in the test set.

We have also tried training binary classifiers for the “Hyponym-of” and the “Synonym-of” relations separately and then merging the outputs of the best classifiers for each relations. While the binary classifiers individually performed better than the multi-way classifier for each corresponding relation class, the overall performance based on the micro-averaged F1-score did not improve over the multi-way classifier after merging the outputs of the hyponym and the synonym classifiers.

Based on the results from the argument ordering strategy experiments, we submitted the model trained using the fixed order strategy, which ranked number one in the challenge. The result is shown in Table 6.

Relation	Precision	Recall	F1-score
Synonym-of	0.820	0.813	0.816
Hyponym-of	0.455	0.421	0.437
Micro-averaged	0.658	0.633	0.645

Table 6: Result on the test set of the ScienceIE dataset, using the official train/dev/test split.

To quantify the importance of various features of our model, we trained the model by gradually adding more features one by one, from word embeddings, relative positions, and entity types to POS tags in order. The results on the importance of the features as well as postprocessing are shown in Figure 2. Adding the relative position features improved the performance the most, while adding the entity type improved it the least. Note that even without the postprocessing, the F1-score is 0.63, which still outperforms the second-best system with the F1-score of 0.54.

Figure 3 quantifies the impact of the two preprocessing steps, deleting brackets and cutting sentences, introduced to compensate for the small dataset size. Cutting the sentence before the first entity and after the second entity resulted in a dramatic impact on the performance, while deleting brackets (i.e., removing the reference marks) improve the performance modestly. This implies that the text between the two entities contains most of the information about the relation between them.

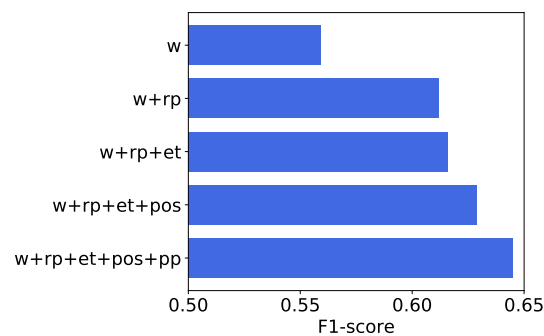


Figure 2: Importance of features of CNN and post-processing rules. w: word embeddings, rp: relative positions to the first and the second arguments, et: entity types, pos: POS tags.

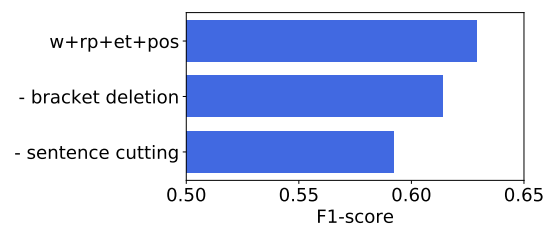


Figure 3: Impact of bracket deletion and sentence cutting. “w+rp+et+pos” represents the CNN model trained using all features with both bracket deletion and sentence cutting during preprocessing. “-bracket deletion” is the same model trained only without bracket deletion, and “-sentence cutting” only without sentence cutting.

5 Conclusion

In this article we have presented an ANN-based approach to relation extraction, which ranked first in the SemEval-2017 task 10 (ScienceIE) for relation extraction in scientific articles (subtask C). We have experimented with various strategies to incorporate argument ordering for ordering-sensitive relations, showing that an efficient strategy is to fix the arguments ordering as appears on the text by introducing reverse relations. We have also demonstrated that cutting the sentence before the first entity and after the second entity is effective for small datasets.

Acknowledgments

The authors would like to thank the ScienceIE organizers as well as the anonymous reviewers. The project was supported by Philips Research. The content is solely the responsibility of the authors and does not necessarily represent the official views of Philips Research.

References

- Iman Amini, David Martinez, and Diego Molla. 2012. Overview of the ALTA 2012 Shared Task. In *Australasian Language Technology Association Workshop 2012*. volume 7, page 124.
- Isabelle Augenstein, Mrinal Kanti Das, Sebastian Riedel, Lakshmi Nair Vikraman, and Andrew McCallum. 2017. SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications. In *Proceedings of the International Workshop on Semantic Evaluation*. Association for Computational Linguistics, Vancouver, Canada.
- Sergey Brin. 1998. Extracting patterns and relations from the world wide web. In *International Workshop on The World Wide Web and Databases*. Springer, pages 172–183.
- Razvan C Bunescu and Raymond J Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*. Association for Computational Linguistics, pages 724–731.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research* 12:2493–2537.
- Aron Culotta and Jeffrey Sorensen. 2004. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd annual meeting on association for computational linguistics*. Association for Computational Linguistics, page 423.
- Franck Deroncourt, Ji Young Lee, and Peter Szolovits. 2016. Neural networks for joint sentence classification in medical paper abstracts. *EACL 2017*.
- Benjamin G Druss and Steven C Marcus. 2005. Growth and decentralization of the medical literature: implications for evidence-based medicine. *Journal of the Medical Library Association* 93(4):499.
- Oana Frunza and Diana Inkpen. 2011. Extracting relations between diseases, treatments, and tests from clinical data. In *Canadian Conference on Artificial Intelligence*. Springer, pages 140–145.
- Sebastian Gehrmann, Franck Deroncourt, Yeran Li, Eric T. Carlson, Joy T. Wu, Jonathan Welt, David W. Grant, Patrick D. Tyler, and Leo A. Celi. 2017. Comparing rule-based and deep learning models for patient phenotyping. *arXiv:1703.08705*.
- Edgar Gonzalez and Jordi Turmo. 2009. Unsupervised relation extraction by massive clustering. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*. IEEE, pages 782–787.
- Cyril Grouin, Asma Ben Abacha, Delphine Bernhard, Bruno Cartoni, Louise Deleger, Brigitte Grau, Anne-Laure Ligozat, Anne-Lyse Minard, Sophie Rosset, and Pierre Zweigenbaum. 2010. Caramba: concept, assertion, and relation annotation using machine-learning based approaches. In *i2b2 Medication Extraction Challenge Workshop*.
- Zhou GuoDong, Su Jian, Zhang Jie, and Zhang Min. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 427–434.
- Kazuma Hashimoto, Makoto Miwa, Yoshimasa Tsuruoka, and Takashi Chikayama. 2013. Simple customization of recursive neural networks for semantic relation classification. In *EMNLP*. pages 1372–1376.
- Hamed Hassanzadeh, Tudor Groza, and Jane Hunter. 2014. Identifying scientific artefacts in biomedical literature: The evidence based medicine use case. *Journal of biomedical informatics* 49:159–170.
- Arif E Jinha. 2010. Article 50 million: an estimate of the number of scholarly articles in existence. *Learned Publishing* 23(3):258–263.
- Su Nam Kim, David Martinez, Lawrence Cavedon, and Lars Yencken. 2011. Automatic classification of sentences to support evidence based medicine. *BioMed Central (BMC) Bioinformatics* 12(2):1.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics (ACL), pages 1746–1751.
- Peder Olesen Larsen and Markus Von Ins. 2010. The rate of growth in scientific publication and the decline in coverage provided by science citation index. *Scientometrics* 84(3):575–603.
- Ji Young Lee and Franck Deroncourt. 2016. Sequential short-text classification with recurrent and convolutional neural networks. In *Human Language Technologies 2016: The Conference of the North American Chapter of the Association for Computational Linguistics, NAACL HLT*.
- Qi Li and Heng Ji. 2014. Incremental joint extraction of entity mentions and relations. In *ACL (1)*. pages 402–412.
- Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. [The Stanford CoreNLP natural language processing toolkit](http://www.aclweb.org/anthology/P/P14/P14-5010). In *Association for Computational Linguistics (ACL) System Demonstrations*. pages 55–60. <http://www.aclweb.org/anthology/P/P14/P14-5010>.

- Anne-Lyse Minard, Anne-Laure Ligozat, and Brigitte Grau. 2011. Multi-class svm for relation extraction from clinical reports. In *Ranlp*. volume 59, pages 604–609.
- Thien Huu Nguyen and Ralph Grishman. 2015. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of NAACL-HLT*. pages 39–48.
- Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English Gigaword fifth edition. Technical report, Linguistic Data Consortium, Philadelphia.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: global vectors for word representation. *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2014)* 12:1532–1543.
- Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 41–47.
- Bryan Rink, Sanda Harabagiu, and Kirk Roberts. 2011. Automatic extraction of relations between medical concepts in clinical texts. *Journal of the American Medical Informatics Association* 18(5):594–600.
- Alexander Schutz and Paul Buitelaar. 2005. Relext: A tool for relation extraction from text in ontology extension. In *International semantic web conference*. Springer, volume 2005, pages 593–606.
- Rion Snow, Daniel Jurafsky, Andrew Y Ng, et al. 2004. Learning syntactic patterns for automatic hypernym discovery. In *NIPS*. volume 17, pages 1297–1304.
- Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, pages 1201–1211.
- Ang Sun and Ralph Grishman. 2012. Active learning for relation type extension with local and global data views. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, pages 1105–1112.
- Charles Sutton and Andrew McCallum. 2006. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning* pages 93–128.
- Ozlem Uzuner, Jonathan Mailoa, Russell Ryan, and Tawanda Sibanda. 2010. Semantic relations for problem-oriented medical records. *Artificial intelligence in medicine* 50(2):63–73.
- Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association* 18(5):552–556.
- Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2015. Semantic relation classification via convolutional neural networks with simple negative sampling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 536–540. <https://aclweb.org/anthology/D/D15/D15-1062>.
- Roman Yangarber and Ralph Grishman. 1998. Nyu: Description of the proteus/pet system as used for muc-7. In *In Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Citeseer.
- Godandapani Zayaraz and Suresh Kumara. 2015. Concept relation extraction using naïve bayes classifier for ontology-based question answering systems. *Journal of King Saud University-Computer and Information Sciences* 27(1):13–24.
- Luke Zettlemoyer. 2013. Relation extraction. *CSE517: Natural Language Processing*.
- Guodong Zhou, Min Zhang, Dong-Hong Ji, and Qiaoming Zhu. 2007. Tree kernel-based relation extraction with context-sensitive structured parse tree information. In *EMNLP-CoNLL*. Citeseer, volume 2007, pages 728–736.