Self-supervised Relation Extraction Using UMLS

Roland Roller and Mark Stevenson

Department of Computer Science
University of Sheffield
Regent Court, 211 Portobello
S1 4DP Sheffield, England

Abstract. Self-supervised relation extraction uses a knowledge base to automatically annotate a training corpus which is then used to train a classifier. This approach has been successfully applied to different domains using a range of knowledge bases. This paper applies the approach to the biomedical domain using UMLS, a large biomedical knowledge base containing millions of concepts and relations among them. The approach is evaluated using two different techniques. The presented results are promising and indicate that UMLS is a useful resource for semi-supervised relation extraction.

1 Introduction

Medline is a large database which contains millions of biomedical articles and scientific abstracts. Every month several thousand new medical abstracts are published on Medline. The volume of documents available make it difficult to identify relevant document. Information Extraction can help with this problem by identifying pre-specified types of information within documents. Relation extraction is a sub-area of Information Extraction that tries to identify relationships between entities (words or concepts) within sentences. The output from relation extraction systems can be used to populate knowledge bases. For example, a relationship could be *DRUG-may_be_used_to_treat-DISEASE* and described in a sentence such as example 1.

Example 1. Dosing regimen effects of **[DRUG:***modafinil***]** for improving daytime wakefulness in patients with **[DISEASE:***narcolepsy***]**. *(PMID: 14520165)*

The goal of relation extraction is to identify pairs of entities within sentences that are connected by a pre-specified relation. Supervised learning approaches have proved successful for this problem. They require positive and negative training examples of the target relation and use machine learning techniques to train a classifier. These approaches have proved successful but require training data (annotated corpus) which is not always available. The generation of an appropriate corpus may require expert knowledge and can be time-consuming. Self-supervision (aka. distant supervision) avoids this bottleneck by using a knowledge base which contains information about the relation of interest to automatically annotate a data set. The baseline assumption is that a sentence which contains entity pairs representing (or not representing) a relation will also express the relationship as well.

E. Kanoulas et al. (Eds.): CLEF 2014, LNCS 8685, pp. 116–127, 2014.

Self-supervised learning has been used in different domains and for different data bases. The technique was originally developed for the biomedical domain by [8] who used the Yeast Protein Database to automatically annotate relation instances. The Unified Medical Language System (UMLS) is a large biomedical knowledge base which contains millions of medical concepts and the relations among them. This work explores the usage of this knowledge base for self-supervised relation extraction from biomedical publications. In particular it provides techniques to measure the efficiency of a self-supervised relation extractor using UMLS. The goal of this paper is to show, that it is possible to use UMLS for this purposes and to provide some baseline evaluation results which can be utilised as a benchmark for further work.

The paper is structured as follows: The next section presents related work. An overview of UMLS is provided in section 3. The following section 4 shows how the baseline data set is generated by matching known facts to the Medline repository. Section 5 explains the classification method used. Section 6 reports an analysis of the annotated examples generated by this process by comparing them against human judgements. The annotated examples are used to train a supervised relation extraction system, the evaluation of which is reported in Section 7. The paper's conclusions are reported in Section 8.

2 Related Work

Relation extraction is the task of detecting or extracting relationships between entities. Supervised relation extraction is a well studied method which uses machine learning techniques to address this problem. This method requires a sufficient amount of training data, consisting of positive and negative training examples. Performance normally improves when more training data is available and when that training data accurately describes the target relation. The support vector machine (SVM) [12] is a popular machine learning technique for relation extraction. A successful SVM-based approach for relation extraction is the TEES system of [3,4]. It was the winning system at the BioNLP Shared Task 2011 and one of the best systems at the BioNLP Shared Task 2013. Other successful approaches for relation extraction use particular kernel methods [19] or combine different models by stacking [17] or ensemble learning [22].

Supervised Machine Learning techniques require annotated training data. This data might be not always available for all different tasks. Furthermore, the generation of an annotated data set for training is time consuming and expensive. Depending on the domain it may even require expert knowledge to carry out the annotation. Self-supervised learning techniques face this problem and avoid using annotated data sets. Instead they utilise already known information and apply an automatic annotation, similar to seed and bootstrapping approaches such as [5] or [1]. While bootstrapping techniques are provided with limited information (in the form of seed examples), self-supervised learning uses a knowledge base with a large amount of information representing a relationship. There are many different data bases available which contain known entity pairs representing different relationships. [8] introduced self-supervision for relation extraction. The authors used the Yeast Protein Database (YPD) which includes subcellar localisation fields for many proteins. The knowledge base refers to certain PubMed articles which contain information about known relationships and were utilised to extract training examples. Some years later their idea of self-supervision for relation extraction was used outside the biomedical domain. [20] focussed on the identification of hypernyms (is-a relationship) using WordNet. [14] introduced self-supervision using Freebase, a large semantic database containing thousands of relations. For each entity pair of one of the main relations they find sentences in Wikipedia containing these entities and extract them. Negative data is produced by generating random pairs which do not appear in Freebase. Unlike previous approaches, this classifier takes the occurrence of an entity pair in several relations at the same time, into account. [11] use the infoboxes of Wikipedia as knowledge source and annotate the information the the articles of Wikipedia. [18] instead use Freebase but annotate the New York Times corpus with the entity pairs. Their work focuses on the three relations *nationality*, *place_of_birth* and *contains*. To train a classifier, the authors introduce the usage of a multi-instance learning [9] approach for this context. Later work still focuses on the same knowledge base and the same corpus, but try to consider the fact that information occur in different relations at once (overlapping relations) [10], try to reduce wrong annotations (labels) [21] or facing the problem of knowledge base gaps [24]. Knowledge base gaps for example can lead to information annotated as negative training data (false negatives) and will influence the classification results.

Applications of self-supervised approaches to relation extraction in the biomedical domain have been limited, the best known approach being [8]. [23] use self-supervision to train a classifier for protein-protein interactions (PPI). Similar to many other approaches in the biomedical domain, the authors use a SVM with a shallow linguistic kernel as classifier. The knowledge about interacting proteins is taken from the database IntAct. Different to for example [14] or other approaches, negative instance pairs are extracted from an additional knowledge base Negatome, which contains proteins which never interact with each other.

This work focuses on self-supervised relation extraction using the Unified Medical Language System (UMLS) as knowledge base. UMLS is a large biomedical knowledge base with millions of medical concepts and relations among them. This knowledge base is much more complex and contains many more relations between them than other self-supervised approaches in the biomedical domain. To our knowledge there is no data set annotated with UMLS relations which could be used to directly evaluate a UMLS-based relation extraction system. In this paper evaluation is carried out using two techniques. A first set of experiments uses a set of UMLS Metathesaurus relations based on the National Drug File - Reference Terminology (NDF-RT) vocabulary which provides information such as *diseases* treated by *drugs*. Evaluation in these experiments is carried out using a *held-out* approach. In addition, the system is also applied to a small set of sentences and the results evaluated manually.

3 Unified Medical Language System

The *Unified Medical Language System* (UMLS) is a set of files and software which combines different biomedical vocabularies, knowledge bases and standards. It includes three tools: Metathesaurus, Semantic Network and SPECIALIST Lexicon. The UMLS

Metathesaurus is a knowledge source containing several million biomedical and health related names and concepts and relationships among them. For this work just the Metathesaurus (version 2013AA) is utilised. The knowledge in the Metathesaurus is a unification of different data sources (source vocabularies) such as the National Drug File (NDFRT), the Medical Subject Heading (MSH) or Authorized Osteopathic Thesaurus (AOT) for example. The utilised UMLS version contains 121 different source vocabularies (including different language variations). UMLS is growing with every new release in terms of further concepts, relations and also source vocabularies. The different source vocabularies have a certain amount of overlap to each other. Overall UMLS is a large knowledge base trying to unify different medical knowledge sources and bring it to a common standard.

The information within UMLS Metathesaurus is spread across different files. The most important ones for this work are MRCONSO and MRREL. MRCONSO contains all medical concepts with all its different names, variants and spellings and unifies them with the **Concept Unique Identifiers** (CUI). The concepts are taken across all different vocabularies. MRREL uses these CUIs and defines binary relations between them. Many of these relations are child-parent relationships, express a synonymy or are vaguely defined as broader or narrower relation. Some other relationships define a better defined relation name (RELA) between two CUIs, such as *moved_from*, *has_location*, *mechanism_of_action_of* or *drug_contraindicated_for*. In general each relationship between two CUIs is defined in both ways, e.g. if there is a relations such as has_location(CUI-A,CUI-B), then there is also a relation location_of(CUI-B,CUI-A). MRREL contains in the UMLS version 2013AA, 52,388,978 instance pairs (relations).

4 Generation of Annotated Corpus

The corpus used to generate annotated data is the Medline repository, which contains abstracts of millions of publications from medicine and related fields. Sentences from Medline containing information of interest are used to generate training examples for distant supervision. Therefore it is necessary to process the Medline abstracts to identify related information. In UMLS relations are expressed by a pair of CUIs. To find out whether a sentence contains two possibly related CUIs, a mapping of UMLS concepts to the sentences of Medline is required. Manual annotation would be impractical so the MetaMap system[1] [2] is used. MetaMap identifies concepts mentioned in text in the form of UMLS CUIs (Concept Unique Identifiers). It first divides the text into segments and then identifies possible UMLS concepts for each segment. This is carried out by identifying the possible CUIs that could related to the sentence (known as *candidates*). Depending on the context and the different possible candidates, MetaMap will provide different possible *mappings* to annotate the sentence with the previously found UMLS concepts. A mapping will always involve only concepts of the candidate list, but it just selects the most probable (or best) ones depending on the context. That means that a mapping usually involves fewer concepts than the candidates. MetaMap can be configured to provide several possible different mappings. For this work the mapping MetaMap considers to be the best is taken as the annotation and all other

---

[1] `http://metamap.nlm.nih.gov/`

candidates ignored since using all possible mappings could generate false annotations. For the experiments a MetaMapped version of the year 2012 (using UMLS release 2011AB) was used, which was provided by the Medline Baseline Repository[2] (MBR). Altogether 3,000,000 abstracts from 1997-2003 of this MetaMapped Medline Repository were utilised. All extracted instance pairs for distant supervision base on the UMLS release 2013AA.

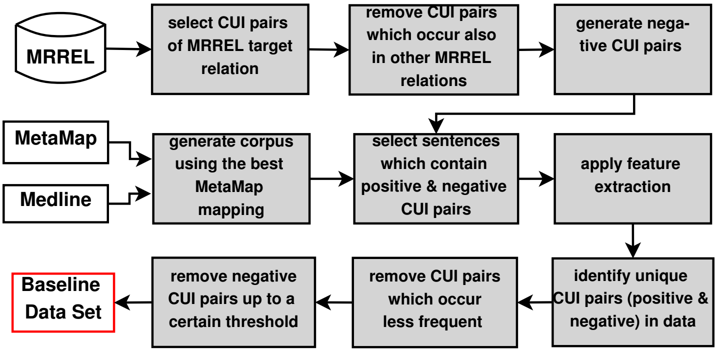

Fig. 1. Processing pipeline to generate the baseline data set

Training examples are based on the relations from the UMLS' MRREL table. To train a self-supervised classifier for a particular UMLS relation, self-supervised training examples have to be generated (positive and negative examples). To generate a classifier for a certain target relation, all CUI pairs for that relation are extracted from MRREL and taken as a set of positive instance pairs. Self-supervision uses the baseline assumption that the occurrence of a positive entity pair in a sentence will describe the relation of interest. Any CUI pairs which also occur in another MRREL relations are removed from the list of positive instance pairs. In the next step, negative instance pairs will be generated. Based on the positive instance pair set, new CUI pair combinations will be generated (combining all CUIs from the first position with all CUIs from the second position). Only if a newly generated CUI pair is not in the positive list and not contained in another MRREL relation, it will be used as negative instance pair. This random generation of new negative instance pairs can usually generate many more CUI combinations than the known positive instance pairs. On the other hand, it might happen that many of these negative instance pairs are not found together within a sentence and are therefore less useful to generate negative training examples.

In the next step sentences of the MetaMapped Medline repository are scanned for positive and negative instance pairs. If a sentence contains a positive (or a negative) CUI pair, it will be taken and processed to be a positive (or negative) training example. The generated examples serve as baseline training set of positive and negative training examples to train the MRREL relational classifier. Figure 1 visualises the different processing steps to generate the baseline data set. Table 1 shows the amount of instance

---

[2] `http://mbr.nlm.nih.gov/Download/MetaMapped_Medline/2012/`

pairs which are available to generate the baseline data set. Furthermore, it shows that just a small number of instance pairs can be found in the utilised subset of the Medline repository. The MRREL relation *drug_contraindicated_for* for instance contains 36,251 CUI pairs. After the removal of CUI pairs (due to the occurrence in other MRREL tables), 28,867 pairs remain. These remaining positive instance pairs can generate 4,103,724 different negative combinations. In the given MetaMapped Medline repository 2,015 sentences containing positive instance pairs have been found. These 2,015 sentences contain just 566 unique CUI pairs.

Table 1. Amount of CUI pairs for the generation of the baseline data set (all); #given: amount of CUI pairs in MRREL, #pos: amount of positive CUI pairs, #neg (g): amount of negative CUI pairs generated from the positive pairs, #u-pos: unique positive pairs, #u-neg: unique negative pairs

MRREL relation	instance pairs			CUI pairs found in baseline data set			
	#given	#pos	#neg (g.)	#pos	#u-pos	#neg	#u-neg
may_treat	48,298	35,271	8,826,775	10,819	2,062	58,719	24,148
drug_contraindicated_for	36,251	28,867	4,103,724	2,015	566	61,609	20,340
physiologic_effect_of	27,684	21,356	4,863,838	694	110	11,612	1,850
mechanism_of_action_of	16,696	12,321	3,265,878	1,091	233	9,252	3,465
may_prevent	6,048	2,337	722,584	2,787	215	16,770	5,383
contraindicating_class_of	2,228	1,756	90,991	1,090	167	20,412	2,668
may_diagnose	967	791	51,535	1,070	61	2,462	570

Depending on the evaluation, further methods to select more useful training examples will be applied. First unique CUI pairs are identified. Sentences with pairs which occur less frequently are removed (for these experiments the threshold is set to 2). Since the amount of unique positive and unique negative CUI pairs are often strongly biased, some unique negative CUIs with their sentences are removed. For the following experiments the amount of utilised unique negative CUI pairs is reduced to a maximum of three times the amount of different unique positive CUI pairs.

5 Relation Classifier

For the self-supervised relation extraction a support vector machine (SVM) is used. This work utilises SVM-Light [12] with the implementation SVM-Light-TK 1.2[3] [15] which takes a combination of a Subset-Tree Kernel (SST) [7] and a polynomial kernel as input. The Charniak-Johnson Parser [6] is used to generate part of speech tags and a syntax tree. The Stanford parser [13] takes this data as input and generates a dependency tree. In addition, words of the sentence are reduced to their stem using the Porter Stemmer [16]. Next the features will be extracted. The syntax tree is used to generate the input for the SST-kernel. Words in the two related entities are replaced with a place-holder and the smallest sub-tree which contains the two entities is extracted. This sub-tree will be used as input for the SST-kernel.

---

[3] `http://disi.unitn.it/moschitti/Tree-Kernel.htm`

The polynomial kernel takes token features and dependency chain features as input, similar to those described in [3]. To adjust the SVM the parameters *cost*, *d* (for the polynomial kernel) and *decay factor* have to be defined.

6 Data Analysis

To analyse the quality of the training examples and automatic annotation, a set of 100 positive examples were selected at random and examined in more detail. Each of the positively annotated entity pairs and sentences will be examined to determine whether they express the relation or not. The examination was carried out by one author, a Computer Science PhD student and biomedical non-expert. To reduce the amount of annotation errors, a simple UMLS relations has been selected: *may_treat*. Some sentences are easy to understand and clearly contain the relation of interest, such as in examples 1 and 2. Sentences which do not contain the relations are more difficult to detect. In some cases the mentioned drug stands in another relationship to the disease, than the relation *may_treat* (such as example 3). Sometimes the sentences express, that a certain drug is not useful to treat a certain disease (see example 4). In that case the sentence is also annotated as negative.

Example 2. We retrospectively studied 9 children (6 with **[DISEASE:***congenital adrenal hyperplasia***]**, CAH) receiving **[DRUG:***hydrocortisone***]** replacement after switching to prednisolone (dose ratio, 1:5). *(PMID: 14517528)*

Example 3. Among the remaining cases, probable **[DISEASE:***type 2 diabetes***]** was defined when a child had one or more of the following characteristics: weight per age > or =95th percentile or acanthosis nigricans at diagnosis, elevated C-peptide or **[DRUG:***insulin***]**, family history of type 2 diabetes; *(PMID: 14517522)*

Example 4. Oral **[DRUG:***insulin***]** doesn't prevent **[DISEASE:***type 1 diabetes***]**. *(PMID: 14528584)*

Overall, 64 sentences were annotated as positive and 36 as negative. This shows that the data contains many false positive and that filtering methods to remove false training examples are required. Furthermore, that amount of false positives in the given set is comparable to the manual examined data set in [18] for the context of Freebase relations.

7 Evaluation Methods

In this section two evaluation methods for self-supervised relation extraction using UMLS are presented. First different relational classifiers are trained and evaluated using the held-out approach (Section 7.1). Next the classifiers are evaluated on a small gold standard evaluation set (Section 7.2).

7.1 Held-Out

Held-out is an evaluation technique which uses part of the data in the knowledge base to train the self-supervised classifier. The remaining parts of the data are used for the evaluation. After removing the less frequent unique CUI pairs and reduction of the negative CUI pairs, the remaining pairs will be divided into a set of training pairs and evaluation pairs. For this work, 3/4 of the remaining positive and negative CUI pairs will be used to generate the training set. The remaining 1/4 of the pairs are used to generate the evaluation set. After splitting into sets of training and evaluation CUI pairs, sentences containing these pairs are sorted into the training and evaluation set.

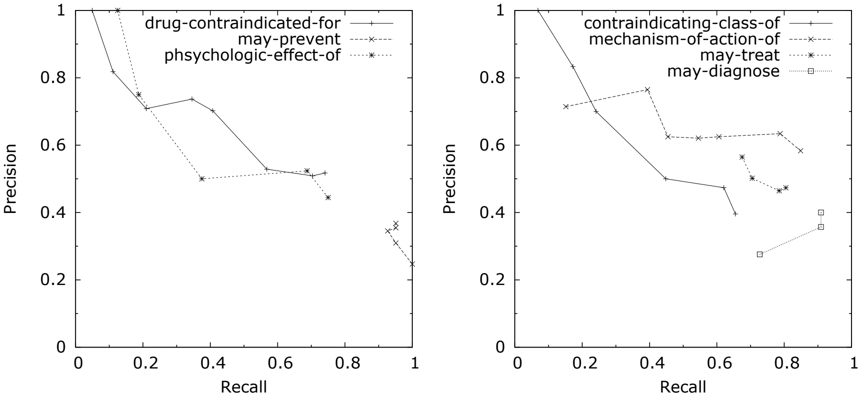

Fig. 2. Held-out evaluation graphs

The relation classifier is trained using the training set and evaluated on the evaluation set. The approach is evaluated as follows: The evaluation set contains different CUI pairs. Some are known (according to MRREL) to describe the target relation. It is assumed, that the other CUI pairs which are not found in the MRREL target relation do not describe the relationship. In the ideal case, sentences containing the positive CUI pairs describe the relation of interest and sentences containing negative CUI pairs do not describe the target relation. The task of the relational classifier is to detect the CUI pairs which are supposed to describe the relation. If a CUI pair is predicted at least once as positive (CUI pairs often occur several times), it will be considered as predicted positive. Otherwise the CUI pair will be considered as predicted as negative.

Figure 2 shows the relation between precision and recall using different configurations in terms of SVM parameter *cost*, *d* and *decay factor* (DF). Depending on the configuration of the decay factor, the different classifiers vary from a high precision with low recall to a low precision with a high recall. Three of the relations (*may_treat*, *may_diagnose* and *may_prevent*) only provide a good recall with a lower precision, but not the other way. This could be an indicator that the chosen training examples contain too many false positives and false negatives. Table 2 presents the best held-out results for the different relations. The evaluation set of the relation *may_prevent* for instance, contains 41 positive and 125 negative unique CUI pairs. 105 different CUI pairs have been predicted at least once as positive and 39 of them have been predicted correct.

The results in the table are compared to a naive baseline approach, which classifies every instance as positive. Compared to the naive results, all self-supervised result provide much better results. The improvements vary between 34% and 75% in terms of F-Score.

Table 2. Best results using held-out

MRREL relation	Unique CUI pairs				Results			Naive Baseline		
	#pos	#neg	#pred	#cor	Prec.	Rec.	F-Score	Prec.	Rec.	F-Score
may_treat	200	597	239	135	0.565	0.675	0.615	0.25	1.0	0.4
may_prevent	41	125	105	39	0.371	0.951	0.534	0.25	1.0	0.4
drug_contraindicated_for	81	246	116	60	0.517	0.741	0.609	0.25	1.0	0.4
physiologic_effect_of	16	48	21	11	0.524	0.688	0.595	0.25	1.0	0.4
mechanism_of_action_of	33	100	41	26	0.634	0.788	0.703	0.25	1.0	0.4
contraindicating_class_of	29	87	38	18	0.474	0.621	0.537	0.25	1.0	0.4
may_diagnose	11	33	25	10	0.400	0.909	0.556	0.25	1.0	0.4

7.2 Manual Evaluation

Although held-out is a useful way to measure the efficiency of a classifier it relies on the data representing the relation of interest. A data set annotated with correct relations is preferable for evaluation purposes. But, to the authors knowledge, there is no existing data set with annotated relations of UMLS. Therefore a small evaluation set is generated manually by one of the authors, a biomedical non-expert and Computer Science PhD student. 100 sentences from the baseline data set for a simple target relation (*may_treat*) are chosen and annotated. The resulting manual annotations are compared with the automatic annotation of the baseline data set in table 3. It shows, that just 3 of the 20 instances automatically annotated as positive were also annotated manually as positive. Further on, 8 instances of the automatic negative annotation were changed to positive instances. The resulting evaluation set contains 11 positive and 89 negative instances.

Table 3. *may_treat:* manual versus automatic annotation

		annotation based on UMLS		
		positives	negatives	
manual annotation	positives	**3**	8	11
	negatives	17	**72**	89
		20	80	

It is important to mention that some sentences are difficult to annotate. Their annotation highly depends on the view of the annotator. The sentence in example 5 for instance was annotated as negative in the evaluation set but could conceivably be annotated differently. The sentence expresses that there is an effect on the disease *Parkinson's disease* using the drug *pramipexole*. It is not clear whether is has a positive or negative effect and whether the author of the underlying sentences suggests the treatment of the disease using *pramipexole*.

Example 5. We compared the antitremor effect of **[DRUG:***pramipexole***]**, pergolide, or placebo in **[DISEASE:***Parkinson's disease***]** (PD). *(PMID: 14639675)*

A further experiment was carried out in which the relation classifier was trained using examples of the *may_treat* relation created using the self-supervision process and evaluated against the manually annotated corpus. Similar to the previous experimental setup, all CUI pairs which occur fewer than two times are removed from the set and the set of negative CUI pairs is reduced to a maximum of three times the number of positive CUI pairs. Since the *may_treat* baseline set contains around 70,000 instances, the set was reduced to decrease the runtime of the classifier before identifying the unique CUI pairs. Table 4 presents the results of this experiment. The *naive* method is a simple baseline technique which predicts that each instance is positive. The best configuration of the basic self-supervised approach with an restriction of the baseline data set to 10,000, easily outperforms the naive approach, with a F-Score of 0.571. This resulting F-score is very close to the result of the *may_treat* classifier within the held-out experiment.

Table 4. Results on evaluation set using *may_treat*

method	**Precision**	**Recall**	**F-Score**
naive	0.110	1.000	0.198
basic DS (max 5,000 training instances)	0.273	0.273	0.273
basic DS (max 10,000 training instances)	0.600	0.545	0.571
basic DS (max 20,000 training instances)	0.417	0.455	0.435

8 Conclusion

This work presented a self-supervised relation extraction system which uses information from the UMLS to generate training examples. The results of this approach are highly promising. They show that UMLS relations can be used to train a relational classifier and extract related entities of biomedical publications. Results based on a standard self-supervised relation extraction platform provide an average F-Score of around 0.6.

This paper demonstrates that, in general, it is possible to use relations from UMLS for self-supervised relation extraction. It also reports evaluation of this approach using two techniques. The results reported here serve as a baseline against which future approaches can be compared.

Acknowledgements. The authors are grateful to the Engineering and Physical Sciences Research Council for supporting the work described in this paper (EP/J008427/1).

References

1. Agichtein, E., Gravano, L.: Snowball: extracting relations from large plain-text collections. In: Proceedings of the Fifth ACM Conference on Digital libraries, DL 2000, pp. 85–94 (2000)
2. Aronson, A., Lang, F.: An overview of MetaMap: historical perspective and recent advances. Journal of the American Medical Association 17(3), 229–236 (2010)
3. Björne, J., Salakoski, T.: Generalizing biomedical event extraction. In: Proceedings of BioNLP Shared Task 2011 Workshop, pp. 183–191. Association for Computational Linguistics, Portland (2011)
4. Björne, J., Salakoski, T.: Tees 2.1: Automated annotation scheme learning in the bionlp 2013 shared task. In: Proceedings of the BioNLP Shared Task 2013 Workshop, pp. 16–25. Association for Computational Linguistics, Sofia (2013)
5. Brin, S.: Extracting patterns and relations from the world wide web. In: Atzeni, P., Mendelzon, A.O., Mecca, G. (eds.) WebDB 1998. LNCS, vol. 1590, pp. 172–183. Springer, Heidelberg (1999)
6. Charniak, E., Johnson, M.: Coarse-to-fine n-best parsing and maxent discriminative reranking. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL 2005, pp. 173–180 (2005)
7. Collins, M., Duffy, N.: New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In: Proceedings of 40th Annual Meeting of the Association for Computational Linguistics, pp. 263–270. Association for Computational Linguistics, Philadelphia (2002)
8. Craven, M., Kumlien, J.: Constructing biological knowledge bases by extracting information from text sources. In: Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology (ISMB), pp. 77–86. AAAI Press (1999)
9. Dietterich, T.G., Lathrop, R.H., Lozano-Perez, T., Pharmaceutical, A.: Solving the multiple-instance problem with axis-parallel rectangles. Artificial Intelligence 89, 31–71 (1997)
10. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D.S.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, ACL 2011, pp. 541–550 (2011)
11. Hoffmann, R., Zhang, C., Weld, D.S.: Learning 5000 relational extractors. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL 2010, pp. 286–295 (2010)
12. Joachims, T.: Making Large-scale Support Vector Machine Learning Practical. In: Advances in Kernel Methods, pp. 169–184 (1999)
13. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp. 423–430 (2003)
14. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: ACL 2009, vol. 2, pp. 1003–1011 (2009)
15. Moschitti, A.: Making tree kernels practical for natural language learning. In: EACL, pp. 113–120 (2006)
16. Porter, M.F.: An Algorithm for Suffix Stripping. In: Readings in Information Retrieval, pp. 313–316 (1997)
17. Riedel, S., McClosky, D., Surdeanu, M., McCallum, A.: D. Manning, C.: Model combination for event extraction in bionlp 2011. In: Proceedings of BioNLP Shared Task 2011 Workshop, pp. 51–55. Association for Computational Linguistics, Portland (2011)

18. Riedel, S., Yao, L., McCallum, A.: Modeling relations and their mentions without labeled text. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010, Part III. LNCS, vol. 6323, pp. 148–163. Springer, Heidelberg (2010)
19. Segura-Bedmar, I., Martínez, P., de Pablo-Sánchez, C.: Using a shallow linguistic kernel for drug-drug interaction extraction. Journal of Biomedical Informatics 44(5), 789–804 (2011)
20. Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. In: Advances in Neural Information Processing Systems (NIPS 2004) (November 2004)
21. Takamatsu, S., Sato, I., Nakagawa, H.: Reducing wrong labels in distant supervision for relation extraction. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, ACL 2012, vol. 1, pp. 721–729 (2012)
22. Thomas, P., Neves, M., Solt, I., Tikk, D., Leser, U.: Relation extraction for drug-drug interactions using ensemble learning. In: DDIExtraction2011: First Challenge Task: Drug-Drug Interaction Extraction at SEPLN 2011, vol. 4, pp. 11–18 (2011)
23. Thomas, I.P., Solt, Klinger, R., Leser, U.: Learning protein protein interaction extraction using distant supervision. In: Proceedings of Robust Unsupervised and Semi-Supervised Methods in Natural Language Processing, pp. 34–41 (2011)
24. Xu, W., Hoffmann, R., Zhao, L., Grishman, R.: Filling knowledge base gaps for distant supervision of relation extraction. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 665–670. Association for Computational Linguistics, Sofia (2013)