Exploring Fine-grained Entity Type Constraints for Distantly Supervised Relation Extraction

Yang Liu   Kang Liu   Liheng Xu   Jun Zhao
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
Zhongguancun East Road #95, Beijing 100190, China
{yang.liu, kliu, lhxu, jzhao}@nlpr.ia.ac.cn

Abstract

Distantly supervised relation extraction, which can automatically generate training data by aligning facts in the existing knowledge bases to text, has gained much attention. Previous work used conjunction features with coarse entity types consisting of only four types to train their models. Entity types are important indicators for a specific relation, for example, if the types of two entities are "PERSON" and "FILM" respectively, then there is more likely a "DirectorOf" relation between the two entities. However, the coarse entity types are not sufficient to capture the constraints of a relation between entities. In this paper, we propose a novel method to explore fine-grained entity type constraints, and we study a series of methods to integrate the constraints with the relation extracting model. Experimental results show that our methods achieve better precision/recall curves in sentential extraction with smoother curves in aggregated extraction which mean more stable models.

1   Introduction

Relation Extraction is the task of extracting semantic relations between a pair of entities from sentences containing them. It can potentially benefit many applications, such as knowledge base construction, question answering (Ravichandran and Hovy, 2002), textual entailment (Szpektor et al., 2005), etc. Traditional supervised approaches for relation extraction (Zhou et al., 2005)(Zhou et al., 2007) need to manually label training data, which is expensive and limits the ability to scale up. Due to the shortcoming of supervised approaches mentioned above, recently, a more promising approach named distantly supervised relation extraction (or distant supervision for relation extraction) (Mintz et al., 2009) has become popular. Instead of manual labeling, it automatically generates training data by aligning facts in existing knowledge bases to text.

However, the paradigm of distant supervision also causes new problems of noisy training data both in positive training instances and negative training instances. To overcome the false positive problem caused by the distant supervision assumption, researches in (Riedel et al., 2010)(Hoffmann et al., 2011)(Surdeanu et al., 2012) proposed multi-instance models to model noisy positive training data, where they assumed that at least one sentence in those containing an entity pair is truly positive. Takamatsu et al. (Takamatsu et al., 2012) claimed that the at-least-one assumption in multi-instance models would fail when there was only one sentence containing both entities. They proposed a method to learn and filter noisy pattern features from training instances to overcome the false positive problem. Researchers (Xu et al., 2013)(Zhang et al., 2013)(Ritter and Etzioni, 2013) tried to address the problem of false negative training data caused by the incomplete knowledge base. Xu el al. (Xu et al., 2013) used the pseudo-relevance feedback method trying to find out the false negative instances and add them into positive training instances. Zhang et al. (Zhang et al., 2013) employed some rules to select negative training instances carefully, hoping not to include the false negative instances. And Ritter et al. (Ritter and Etzioni, 2013) used hidden variables to model the missing data in databases based on a graphical model. The training data generation process for all the above work is under the framework of (Mintz et al., 2009),

one important step of which is to recognize entity mentions from text and assign them entity types which are used to compose features for training the model. The entity types they used are very coarse only consisting of four categories (PERSON, ORGANIZATION, LOCATION, NONE). We argue that the coarse entity types are not sufficient to indicate relations.

A specific relation constrains the entity types of its two entities. For instance, the *SingerOf* relation limits the entity type of its first entity as PERSON or more fine-grained ARTIST, and the entity type of its second entity as ART or more fine-grained MUSIC. Therefore, when extracting a relation instance, the entity types of its two entities are important indicators for a specific relation. Previous work used conjunction features (Details in Section 3.3) by combining the coarse entity types of entity mentions with its contextual lexical and syntactic features. However, the conjunction features may fail to distinguish the relations. For example, the following two sentences contain two relation instances, one is *DirectorOf(Ang Lee, Life of Pi)*, and the other is *AuthorOf(George R.R. Martin, A Song of Ice and Fire)*.

1. **Ang Lee**'s **Life of Pi** surprised many by scoring a leading four Oscars on Sunday night...

2. Westeros is the premiere fansite for **George R.R. Martin**'s **A Song of Ice and Fire**.

Only using the above conjunction features, we cannot tell the difference between the two entity pairs, and are probable to incorrectly classify them as the same relation. By contrast, if we can assign each entity with fine-grained entity types, for example, *Ang Lee* as the entity type ARTIST and *George R.R. Martin* as AUTHOR, we may succeed in classifying the two entity pairs correctly.

To achieve the goal mentioned above, there are mainly three challenges: (1) how to define the fine-grained type set; (2) how to assign the types to entity mentions; (3) how to integrate the fine-grained entity type constraints with the relation extracting model. To address these challenges, in this paper, we propose a novel approach to explore the fine-grained entity type constraints for distantly supervised relation extraction. First, we use the types defined in (Ling and Weld, 2012) stemmed from Freebase[1] as the fine-grained entity type set (introduced in Section 3.1). Second, we leverage Web knowledge to train a fine-grained entity type classifier and predict entity types for each entity mention. Third, we study several methods to integrate the type constraints with an existing system *MULTIR*, a multi-instance multi-label model in (Hoffmann et al., 2011), to train the extractor.

In summary, the contribution of this paper can be concluded as follows.

(a) We explore the effect of fine-grained entity type constraints on distantly supervised relation extraction. A novel method is proposed to leverage Web knowledge to automatically train a fine-grained entity type classifier, which is used to predict the fine-grained types of each entity mention.

(b) We study a series of methods for integrating the fine-grained entity type constraints with the extracting model and compare their performance with different parameter settings.

(c) We conduct experiments to demonstrate the effects of the newly exploited fine-grained entity type constraints. It shows that our method achieves a much better precision/recall curves over the baseline system in sentential extraction, and improves the performance with a smoother precision/recall curve in aggregated extraction, which means a more stable model.

2   Distant Supervision for Relation Extraction

We define a relation instance (or a fact), which means a binary relation, as $r(e_1, e_2)$. r is the relation, and e_1 and e_2 mean the two entities in the relation instance, for example, $BornIn(Yao\ Ming, Shanghai)$. Distant supervision supplies a method to automatically generate training data. In this part, we will introduce the general steps in distant supervision for relation extraction. First, we define the notations we use. Σ denotes sentences comprising the corpus, E denotes entity mentions in the corpus which are consecutive words with the same named entity tags assigned by an NER system, Δ denotes the facts (or relation instances) in the existing knowledge base. R denotes the relations in Δ.

---

[1]http://www.freebase.com/

person		organization		terrorist_organization
actor	doctor	airline		government_agency
architect	engineer	company		government
artist	monarch	educational_institution		political_party
athlete	musician	fraternity_sorority		educational_department
author	politician	sports_league		military
coach	religious_leader	sports_team		news_agency
director	soldier			
	terrorist			

location	body_of_water	product		camera	art		written_work
city	island	engine		mobile_phone	film		newspaper
country	mountain	airplane		computer	play		music
county	glacier	car		software			
province	astral_body	ship		game	event		military_conflict
railway	cemetery	spacecraft		instrument	attack		natural_disaster
road	park	train		weapon	election		sports_event
bridge					protest		terrorist_attack

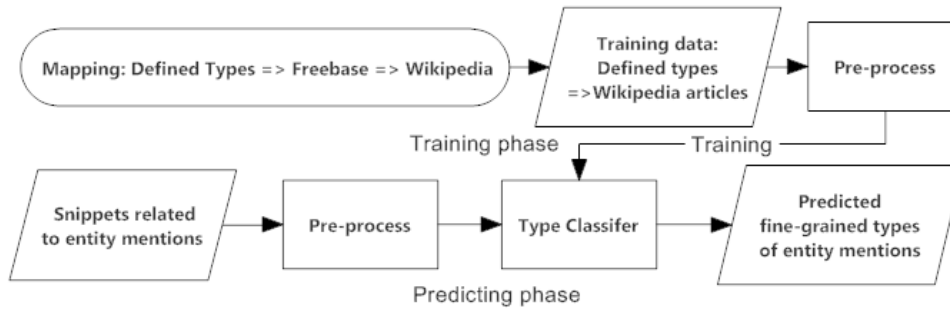| building	time	chemical_thing	website
airport	color	biological_thing	broadcast_network
dam	award	medical_treatment	broadcast_program
hospital	educational_degree	disease	tv_channel
hotel	title	symptom	currency
library	law	drug	stock_exchange
power_station	ethnicity	body_part	algorithm
restaurant	language	living_thing	programming_language
sports_facility	religion	animal	transit_system
theater	god	food	transit_line

Figure 1: Fine-grained entity type set.

Figure 2: Framework of fine-grained entity type classifier.

To generate training data, we align pairs of entity mentions in the same sentence with Δ. The aligned entity mentions E_{train} and their sentences Σ_{train} along with R_{train} are used as training data. Features are extracted from them to train the relation extracting model.

To predict the unknown data for extracting new relation instances, we input pairs of entity mentions $E_{predict}$ and the sentences containing them $\Sigma_{preidct}$ into the trained extracting model for extracting new relation instances.

3 Fine-grained Entity Type Constraints

Entity mentions in sentences are considered consecutive words with the same entity types (Section 2). The entity types are part of the lexical and syntactic features(Mintz et al., 2009), and the feature setting is followed by other related work. Their entity types are assigned by an NER system and consist of four categories (PERSON, ORGANIZATION, LOCATION, NONE). The types of entity mentions in a relation are important indicators for the very type of relation. However, the coarse (only four types) entity types may not capture sufficient constraints to distinguish a relation. In this section, we explore fine-grained entity type constraints and study different methods to integrate them with the extracting model.

This section first introduces the fine-grained entity type set(Section 3.1), and then describes our method which leverages Web knowledge to train the fine grained entity type classifier and assign entity mentions with the fine-grained entity types (Section 3.2). At last, we illustrate methods to integrate fine-grained entity type constraints with the relation extracting model.

2109

Entity pair	[Hank Ratner], [Cablevision]					
Sentence	Cablevision's $600 million offer came in the form of a letter to Peter S.Kalikow, chairman of the M.T.A., from the Garden's vice chairman, Hank Ratner.					
Conjunction	Reverse	Left	NE1	Middle	NE2	Right
Feature examples	False		PER		ORG	
	False	Hank[NMOD]	PER	[NMOD]chairman ... offer[SBJ]	ORG	
	True	B_-1	ORG	POS $ ... NN NN,	PER	.B_1

Table 1: Examples of conjunction features.

3.1 Fine-grained Entity Types

Figure 1 is the type set we use. It was introduced in (Ling and Weld, 2012) and was derived from Freebase types. The bold types in each small box of Figure 1 are upper-class types for others in that small box. For example, $/actor$ is a lower-class type of $/person$ which is denoted as $/person/actor$. And $/person$ and $/person/actor$ coexist in the type set.

3.2 Fine-grained Entity Type Classifier

In this section, we describe our method that leverages Web knowledge to train a fine-grained entity type classifier and predict entity types of each entity mention. Its architecture is shown in Figure 2.

3.2.1 Training

The training data are obtained from Wikipedia. Because the defined fine-grained types are tailored based on Freebase types, we can find the mappings between the two type sets, for example, $/person/doctor$ maps to two Freebase types $/medicine/physician$ and $/medicine/surgeon$. And Freebase WEX[2] supplies a mapping between Freebase types to Wikipedia articles. As a result, we can map Wikipedia articles to defined fine-grained types.

Based on the mappings, we obtain Wikipedia articles for each type as training data and negative training examples are sampled from articles not contained in the mappings. We preprocess the articles by: stop words filtering, stemming, and term frequency filtering and use a maxent model to train the classifier.

3.2.2 Predicting

To predict types of each entity mention, we first use search engines to expand entity mentions. Specifically, each entity mention is used as a query sent to the search engine[3]. Titles and descriptions of top k returned snippets are selected (We keep the top 20 in the experiments). The obtained text are preprocessed with the same method as training examples. Then we use the trained fine-grained entity type classifier to predict the types of each entity mention.

After predicting, we obtain a ranked list of types for each entity mention, which are ranked by the predicting scores.

3.3 Integrating Fine-grained Entity Type Constraints into the Extracting Model

This section introduces our methods to integrate the fine-grained entity type constraints with the extracting model. First of all, we briefly review the features used in previous models which derived from (Mintz et al., 2009) and (Riedel et al., 2010). Their features mainly comprise two types: lexical features (POS tags, words and entity types) and syntactic features (dependency parsing tags, words and entity types). Each feature is a conjunction with several parts: entity types of two entity mentions, the left context window of the first entity mention, the right context window of the second entity mention and the part between them (the window contains none or one or two words). Table 1 shows an example of the conjunction features.

---

[2]http://wiki.freebase.com/wiki/WEX
[3]We use Bing search API. http://datamarket.azure.com/dataset/bing/search

To integrate the exploited fine-grained entity type constraints with the extracting model, we proposed three methods (substitution, augment and selection) to make the type constraints take effects.

3.3.1 Substitution Method

In this method, we substitute coarse entity types of the features with the entity mentions' fine-grained types, and use the new features to train the model. Instead of substituting directly, an entity mention is first represented by its fine-grained types and the upper-class of the fine-grained type, for example, $/person/politician$ derives two types $/person$ and $/person/politician$ itself. The reason is that the extracting model can benefit from the related types like the upper-class types. And then we use the obtained entity types to substitute the old coarse entity types as new features greedily, which means that all the possible combinations of types between the entity pair are considered. For example, "$Barack\ Obama$" has the fine-grained type $/person/politician$ and his birth place "$Hawaii$" has the type $/location/island$, then there are 4 combinations between the two entities, they are ($/person$, $/location$), ($/person$, $/location/island$), ($/person/politician$, $/location$) and ($/person/politician$, $/location$).

3.3.2 Augment Method

In this method, we generate new features by substituting the coarse entity types with predicted fine-grained types, and expand the old features with new features. Different from the substitution method, we do not add the upper-class types, for that we think the coarse types in old features have the same effect. In this method, we use the fine-grained constraints as a complementary.

3.3.3 Selection Method

The selection method is similar to the augment method. The difference is that we do not expand all old features with new features. We select some of them to expand. The reason is that some of the conjunction features are of high-precision themselves, it can clearly indicate the relations with its left, middle and right parts, even without the entity types (informative ones). If we expand these features, it may cause more noisy features. So we expect to only expand the ones that lack of the indicating abilities (non-informative ones). In this paper, we employ a simple method to distinguish between the informative ones and non-informative ones by the length of the features, which means that the longer is more informative than the shorter. In our experiments, the length threshold is set as 20.

In the predicting phase (Section 3.2), we obtain a ranked type list for each entity mention. The top list types are considered in our methods. Experiments in Section 4.3 are conducted on top k $\{k \in 1, 2, 3\}$ type/types in the obtained ranked list. And they are combined with a greedy method similar to that in the substitution method explained above.

4 Experiments

4.1 Settings

We use the same data sets as (Riedel et al., 2010) and (Hoffmann et al., 2011), where NYTimes sentences in the years 2005-2006 are used as training corpus Σ_{train} for distant supervision and sentences in 2007 are used as testing corpus $\Sigma_{predict}$. The data was first tagged with an NER system (Finkel et al., 2005) and consecutive words with the same tag are extracted as entity mentions. And then, entity mentions E_{train} in training corpus are aligned to facts Δ in Freebase as training examples to train the models.

We integrate our fine-grained entity type constraint with MULTIR, an existing multi-instance multi-label extracting model in (Hoffmann et al., 2011). Following their setttings, we conduct experiments on aggregated extraction and sentential extraction to show the effect of fine-grained entity type constraints.

- **Aggregated extraction**: Aggregated extraction is corpus-level extraction. When given an entity pair, it predicts its relation types based on the whole corpus. After extraction, the precision and recall are computed by comparing the results with facts in Freebase. The evaluation underestimates the accuracy because there may be correct facts in the extracted results but not existing in Freebase, these facts are labeled as incorrect by mistake here. Because aggregated extraction is an automatic evaluation, it is used to tune parameters like held-out evaluation in (Mintz et al., 2009).
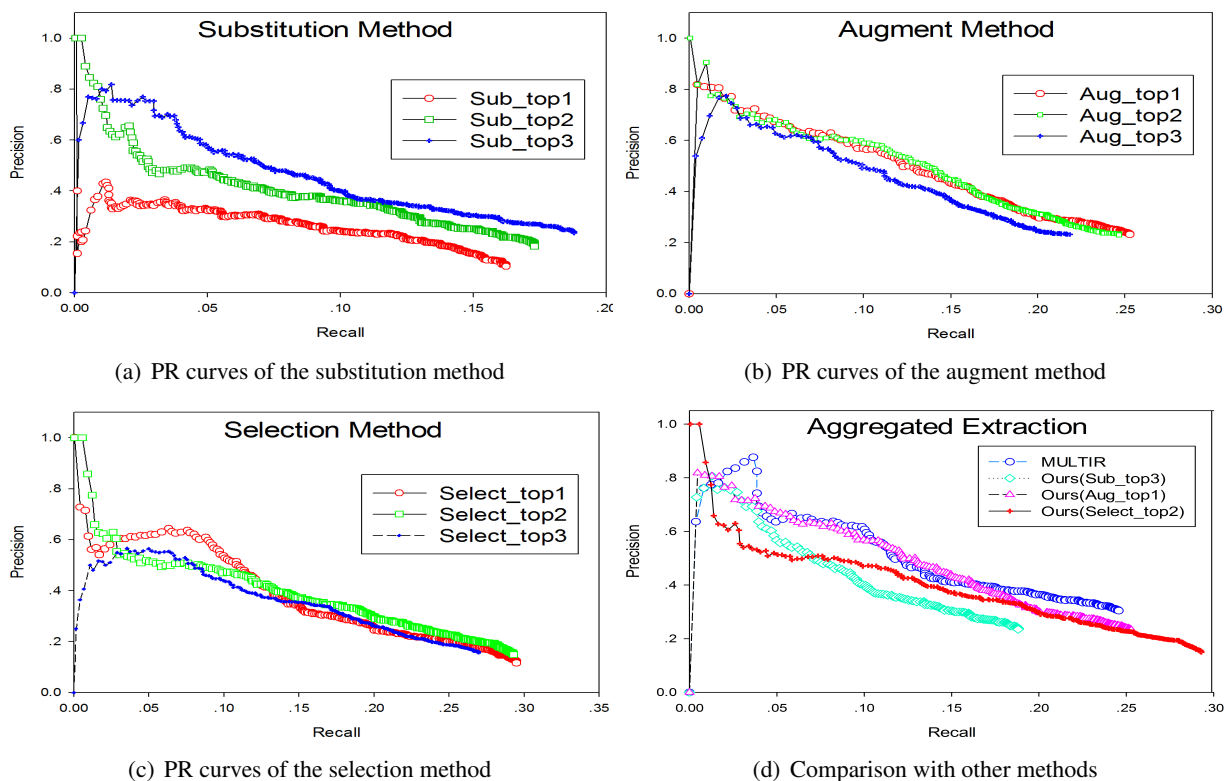
(a) PR curves of the substitution method

(b) PR curves of the augment method

(c) PR curves of the selection method

(d) Comparison with other methods

Figure 3: Precision-recall (PR) curves of the aggregated extraction.

- **Sentential extraction**: Sentential extraction predicts an entity pair only based on a specified sentence containing the pair of entities. We use manually labeled data in (Hoffmann et al., 2011) as benchmark. The data consist of 1,000 sentences and are sampled from the results their system outputs and sentences aligned with facts in Freebase. As they stated in their paper, these results provide a good approximation to the true precision but can overestimate the actual recall.

4.2 Experimental Results

In aggregated extraction, we first evaluate the three type-constraint integration methods (substitution, augment and selection) with the top k $\{k \in 1, 2, 3\}$ type/types (Section 3.3). And then, we compare the best parameter setting methods with previous work. In sentential extraction, we compare methods tuned in aggregated extraction with $MULTIR$.

4.2.1 Aggregated Extraction

Figure 3 shows the precision-recall (PR) curves of the aggregated extraction. In it, Sub_topk $\{k \in 1, 2, 3\}$ means using the substitution method (Section 3.3) with top k fine-grained entities types returned by the type classifier in Section 3.2. Correspondingly, Aug_topk is for the augment method and $Select\_topk$ is for the selection method.

Figure 3(a) shows that Sub_top3 outperforms the other two settings of k in the substitution method, it seems that more fine-grained types produce better curves. In Figure 3(b), Aug_top1 and Aug_top2 achieve similar performances. However, when adding one more type with $k = 3$, we obtain a lower curve, which contradicts the trend showed in the curves of the substitution method (Figure 3(a)). Figure 3(c) shows the PR curves of three selection methods, $Select\_top1$ has a better performance at the beginning. Then $Select\_top2$ exceeds it a bit consistently.

In Figure 3(d), we demonstrate the comparison of best tuned methods above with previous work. They are Sub_top3, Aug_top1 and $Select\_top2$. From Figure 3(d), it shows that, among the three of our methods, Aug_top1 achieves better precisions along the PR curves, and $Select\_top2$ reaches the best
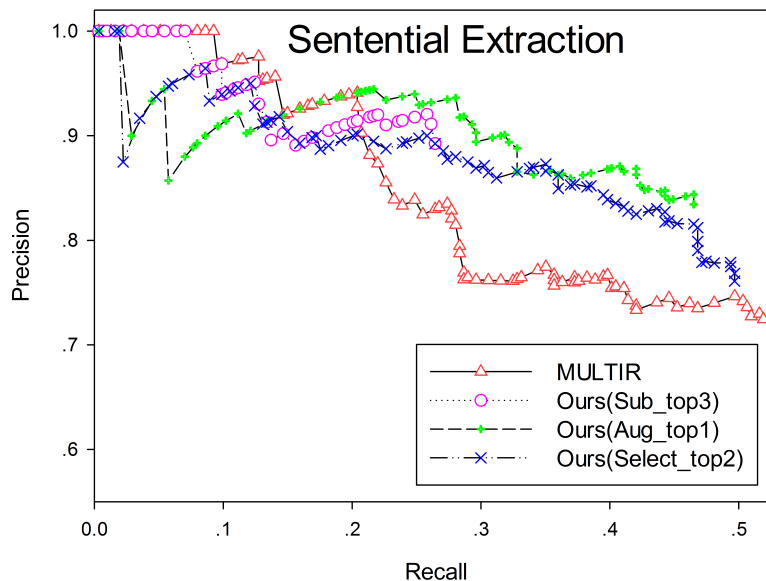
2112

Figure 4: Comparison with MULTIR

recall at the highest recall point. Comparing to other methods, the PR curve of Aug_top1 reaches a higher recall with 29.3% at the highest recall point than *MULTIR* (24.5%). $Select\_top2$ achieves 29.3% at the highest recall point, best among all methods. And by integrating the fine-grained entity type constraints, they improve the PR curve of *MULTIR* with a more smoother curve without most of the depressions seen in *MULTIR*. As stated in (Hoffmann et al., 2011), the smoother curve indicated a more stable model.

4.2.2 Sentential Extraction

Figure 4 shows the precision-recall (PR) curves of the sentential extraction. In the evaluation, we compare the three best integration methods tuned in aggregated extraction with original *MULTIR*. Among our three method, Aug_top1 outperforms in precision and achieves a better curve in general among the three methods, however, $Select\_top2$ gains a better recall at the end. Sub_top3 has the worst recall. In general, our methods have much better precisions than *MULTIR*. Aug_top1 and $Select\_top2$ achieve better curves than *MULTIR*. Since the evaluation of sentential extraction is a good approximation of precision, it implies that the proposed methods are effective.

4.2.3 Analysis

On one hand, among the three proposed integration methods, generally, the augment method and selection method get better performance. The reason is that substitution method uses predicted fine-grained entity types to replace the old coarse features in the conjunction features completely, and the conjunction features are sensitive to entity types for different entity types indicate different conjunction features, as a result, if we can not promise a good accuracy in the type classification which is hard to achieve in classifying hundreds of fine-grained types, the performance will be badly influenced. Different from the substitution method, augment method and selection method keep the old features with coarse features, they use the features with fine-grained entity type constraints as extra information to help the extraction and achieve better results.

On the other hand, comparing to other methods, by integration the exploited fine-grained entity type constraints, our methods achieve improvements in both aggregated and sentential extraction. It proves that the fine-grained entity type constraints we exploit are effective, and our proposed integration methods succeed in integrating the constraints into the extracting model. Our augment method outperforms *MULTIR* in precision along the PR curves in sentential extraction and improve it performance with a more smoother PR curve in aggregated extraction, which indicates a more stable model. Moreover, the method gets a better recall. And our selection method consistently outperforms *MULTIR* in sentential

2113

	k=1	k=2	k=3
Recall@k	0.596	0.740	0.806

Table 2: Evaluation of the fine-grained type classifier.

extraction. In aggregated extraction, it also achieves a smoother curve and an impressive promotion at the highest recall point. Since the evaluation of aggregated extraction only considers the facts existing in Freebase which may incorrectly label the right extracting results and underestimate the true precision, and based on its better performance of precision in sentential extraction, we consider it is a more promising method. This paper only employs very naive method to select the non-informative features by its length (Section 3.3.3), a more effective selecting method may lead further improvements.

4.3 Performance of Entity Type Classifier

We evaluate the performance of the fine-grained entity type classifier (Section 3.2). In section 3.2, we sample the training examples from a collection of Wikipeida articles mapped with the fine-grained types. To generate test entity mentions, we first remove the sampled training articles from the collection, and then sample the articles from it, where the titles of sampled articles are used as the test entity mentions (we sample 12,000 test entity mentions) and their mapped fine-grained types are used as benchmark. After that, the predicting method in Section 3.2.2 is used to expand mentions and predict the types of each test entity mention. After predicting, we obtain a ranked list of types for each test entity mention.

To evaluate, we define a notation of *Hit@k*, which equals 1 if the true type of an entity mention is hit in the top k predicted types, otherwise equals 0. And then we evaluate it by the $Recall@k$ defined bellow.

$$Recall@k = \frac{\Sigma_{i=1}^{12000} Hit@k_i}{12000} \quad (1)$$

In equation (1), i means the ith test entity mention. Table 2 shows the results for the top 3 predicted types.

5 Related Work

Distant supervision (also known as weak supervision or self supervision) is used to a broad class of methods in information extraction which aims to automatically generate labeled data by aligning with data in knowledge bases. It is introduced by Craven and Kumlien (Craven et al., 1999) who used the Yeast Protein Database to generate labeled data and trained a naive-Bayes extractor. Bellare and McCallum (Bellare and McCallum, 2007) used BibTex records as the source of distant supervision. The KYLIN system in (Wu and Weld, 2007) used article titles and infoboxes of Wikipedia to label sentences and trained a CRF extractor aiming to generate infoboxes automatically. The Open IE systems TEXTRUN-NER (Yates et al., 2007) and WOE (Wu and Weld, 2010) trained their extractors with the automatic labeled data from Penn Treebank and Wikipedia infoboxes respectively.

Mintz (Mintz et al., 2009) first introduced their work that performed distant supervision for relation extraction. It used Freebase as the knowledge base to align sentences in Wikipedia as training data and trained a logistic regression classifier to extract relations between entities.Distant supervision supplied a method to generate training data automatically, however it also bring the problem of noisy labeling. After their work, a variety of methods focused to solve this problem. Riedel (Riedel et al., 2010) proposed a multi-instance model to model the false positive noise in training data with the assumption that at least one of the labeled sentences truly expressed their relation. After their work, Hoffmann (Hoffmann et al., 2011) and Surdeanu (Surdeanu et al., 2012) tried to not only model the noisy training data, but also overcame the problem of multi-label where two entities may exist more than one relation, they proposed graphic models as kinds of multi-instance multi-label learning methods and made improvements over previous work. The at-least-one assumption would fail when encountering entity pairs with only one aligned sentence. Takamatsu (Takamatsu et al., 2012) employed an alternative approach without the mentioned assumptions. Their work predicted negative patterns using a generative model and remove labeled data containing negative patterns to reducing noise in labeled data.

Besides the problem of false positive training examples caused by distant supervision. There were a bunch of researches trying to solve the problem of false negative training examples caused by incomplete knowledge bases. Zhang (Zhang et al., 2013) made heuristic rules to filter the false negative training examples. And Xu (Xu et al., 2013) tried to overcom this problem by pseudo-relevance feedback. Min (Min et al., 2013) improved MIML in (Surdeanu et al., 2012) by adding a new layer in their 3-layer graphic model to model the incomplete knowledge base. Ritter (Ritter and Etzioni, 2013) employed similar intuition with (Xu et al., 2013) that they thought rear entities missing in the database would be often mentioned in the text. They proposed a latent-variable approach to model it and showed its improvement over aggregate and sentential extraction.

6 Conclusion

In this paper, we propose a novel approach to explore the fine-grained entity type constraints for distantly supervised relation extraction. We leverage Web knowledge to automatically train a fine-grained entity type classifier and predict entity types of each entity mention. And we study a series of methods to integrate the type constraints with a relation extraction model. At last, thorough experiments are conducted. The experimental results imply our methods are effective with better precision/recall curves in sentential extraction and smoother precision/recall curves in aggregated extraction, which indicate more stable models.

In the future we hope to explore more details of integration methods that integrates fine-grained entity type constraints with relation extraction models, especially the selection integration method. We consider that a more effective method to distinguish between the informative and non-informative features will lead more improvements.

Acknowledgements

References

Kedar Bellare and Andrew McCallum. 2007. Learning extractors from unlabeled text using relevant databases. In *Sixth International Workshop on Information Integration on the Web*.

Mark Craven, Johan Kumlien, et al. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86. Heidelberg, Germany.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 541–550.

Xiao Ling and DS Weld. 2012. Fine-Grained Entity Recognition. In *AAAI*.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of NAACL-HLT*, pages 777–782.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. 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: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

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*, pages 41–47. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Alan Ritter and Oren Etzioni. 2013. Modeling Missing Data in Distant Supervision for Information Extraction. *Transactions of the Association for Computational Linguistics*, 1:367–378.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics.

Idan Szpektor, Hristo Tanev, Ido Dagan, Bonaventura Coppola, et al. 2005. *Scaling Web-based aquisition of entailment relations*. Ph.D. thesis, Tel Aviv University.

Shingo Takamatsu, Issei Sato, and Hiroshi Nakagawa. 2012. Reducing wrong labels in distant supervision for relation extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 721–729. Association for Computational Linguistics.

Fei Wu and Daniel S Weld. 2007. Autonomously semantifying wikipedia. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50. ACM.

Fei Wu and Daniel S Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127. Association for Computational Linguistics.

W Xu, RH Le Zhao, and R Grishman. 2013. Filling Knowledge Base Gaps for Distant Supervision of Relation Extraction. *Proceedings of Association for Computational Linguistics*.

Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead, and Stephen Soderland. 2007. Textrunner: open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 25–26. Association for Computational Linguistics.

Xingxing Zhang, jianwen Zhang, Junyu Zeng, Jun Yan, Zheng Chen, and Zuifang Sui. 2013. Towards Accurate Distant Supervision for Relational Facts Extraction. In *Proceedings of Association for Computational Linguistics*.

GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 427–434. Association for Computational Linguistics.

GuoDong Zhou, Min Zhang, Dong Hong Ji, and Qiaoming Zhu. 2007. Tree kernel-based relation extraction with context-sensitive structured parse tree information. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.