Uncovering Business Relationships: Context-sensitive Relationship Extraction for Difficult Relationship Types

Zhe Zuo, Michael Loster, Ralf Krestel, Felix Naumann

Hasso Plattner Institute
Prof.-Dr.-Helmert-Straße 2-3, 14482 Potsdam, Germany
{firstname.lastname}@hpi.de

Abstract. This paper establishes a semi-supervised strategy for extracting various types of complex business relationships from textual data by using only a few manually provided company seed pairs that exemplify the target relationship. Additionally, we offer a solution for determining the direction of asymmetric relationships, such as "ownership_of". We improve the reliability of the extraction process by using a holistic pattern identification method that classifies the generated extraction patterns. Our experiments show that we can accurately and reliably extract new entity pairs occurring in the target relationship by using as few as five labeled seed pairs.

1 Business Networks

Extracting structured data from text, and thus harnessing the valuable information on the web and hidden in the vast amounts of other textual data, is a well-known and well-studied research area. As the text corpora and the kind of information to be extracted from them can vary greatly, many research works have focused on specific types of information, on specific corpora, on specific application domains, on specific languages, or any combination of the above. In this paper, we regard the problem of extracting *relationships* of several specific types among *companies* from *news articles*.

Many tasks, such as building business networks, predicting risks, or valuating companies, can significantly benefit from accurately extracting relationships between companies. Imagine a scenario in which Dell wants to acquire EMC. Dell plans to finance the deal by taking out a loan. The chosen bank has to decide whether to award the loan based on the careful assessment of the risk associated with this transaction. With the explosive growth of the textual data on the web, it becomes possible to discover not only the information of Dell and EMC but also the dependencies by extracting business relationships and building up a company network. In the same example, by analyzing the network structure of both companies, the bank might reach the conclusion that the risk of granting a loan is too high, because many of EMC's subsidiaries, as given by the relationship network, are struggling. With this knowledge the bank might award a smaller or no loan at all or propose a higher interest rate.

To build up a business network between companies, it is critical to reliably extract business relationships. Companies often connect to each other via the activities in which they participate. Business relationships represent a subset of these activities; examples include ownership_of, partnership_with, supplier_of, and so on. Only very few of them can be found in structured knowledge base like Freebase [4] or semi-structured data like Wikipedia infoboxes – a substantial amount of relationships is hidden in unstructured data sources. Aggravating this situation, both Freebase and infoboxes contain only the major subsidiaries of some companies (i.e., ownership_of relationship). Other relationships, such as partnership_with or supplier_of, are not covered.

Given a corpus of unstructured textual data, we aim to (1) discover whether two co-occurring companies *participate* in a business relationship, (2) identify the *type* of the relationship, and (3) in the case of an asymmetric business relationship, determine its *direction*.

The task of business relationship extraction is challenging due to the complex nature of the relationships between companies. First, multiple types of relationships can exist between two companies. Samsung as one of the biggest competitors of Apple is also the supplier of displays for Apple's products. Moreover, as an example of resolving the direction of asymmetric relationships, such as the ownership_of relationship, consider that Walt Disney owns ABC Studios but not the other way around. Being able to successfully derive the direction of the relationships is of vital importance for many subsequent tasks.

The Snowball system addresses the general problem of relationship extraction [1], and our work is based in parts on its general idea. It takes a small set of entity pairs as a seed set and generates candidate patterns that are based on the context of these pairs. Subsequently, the most prominent patterns are selected according to a scoring function and used to extract new entity pairs that participate in the target relationship. In the end, the newly selected pairs are added to the seed set and the process repeats to generate more patterns. However, Snowball functions only correctly if there is a one-to-many relationship between the participating entities, e.g., in the headquarter_of(Microsoft, Redmond) relationship, Microsoft has exactly one headquarter. Business relationships do not adhere to this characteristic, which is the reason Snowball is unable to solve the problem at hand.

We extend the Snowball idea by introducing a key-phrase extraction strategy, which allows us to remove irrelevant parts of the context surrounding the company pairs. To determine the direction of asymmetric relationships, we propose a process that leverages information contained in the seed set. Since Snowball cannot deal with many-to-many business relationships, we propose a generalization of their tuple- and pattern-evaluation strategy by specifying a new selection method to select patterns and new seeds. We further define a holistic pattern identification strategy, which enables us to extract multiple relationship types simultaneously.

In summary, we propose a system to perform *(directed) relationship extraction (RE) between companies* from textual data. Addressing this problem, we present a novel, semi-supervised relationship extraction method, which requires only a minimum amount of manually specified company pairs to efficiently extract new ones that belong to the same target relationship. Additionally, we provide a straightforward solution to reliably identify the direction of asymmetric relationships. We show that our approach is superior to more advanced distant learning approaches for the particularly difficult case of many-to-many relationships.

2 Background and Related Work

The most related work is the Snowball system [1], which we have already introduced in Section 1. Despite the fact that there is a large body of work that focuses on the topic of relationship extraction, the subject of extracting business relationships between companies from unstructured data has not been sufficiently addressed by research.

One way to approach the general relationship extraction problem is to use supervised learning techniques by classifying whether two entities participate in a specific relationship. Kambhatla [6] employed Maximum Entropy models to solve the relationship extraction task. Zhou et al. [13] also applied a feature-based relationship extraction strategy that uses Support Vector Machines (SVM) [8]. Further, kernel methods with string-kernels have successfully been applied to deal with the RE problem [11]. The major drawback of these techniques is that a large amount of labeled data is required for training. As a representative example, Kambhatla [6] uses a training set that contains around 9,752 instances of relationships to generate their results. Moreover, relabeling and retraining of the model becomes necessary, as soon as either the underlying characteristics of the data sources or the target relationship change substantially.

Mintz et al. [7] introduced a distant supervision approach, which avoids the expensive labeling process. The idea is to automatically label the training data according to the relationships included in knowledge bases, i.e., Freebase [4]. One of the limitations is that it is highly rely on the given knowledge base, only the types of relationships that are included can be extracted, while most of the business relationships are not covered at all, such as partnership_with, competitor_of and supplier_of .

Another way to address the problem was presented by Banko et al. [2]. They introduced an unsupervised approach called TextRunner to extract all possible relationships in a given corpus without requiring any labeled data. This task is known as the *open information extraction* (Open IE) task. Wu and Weld [10] proposed the WOE system, which enhances TextRunner by including additional information from Wikipedia infoboxes to construct a training dataset. Although the Open IE approaches can automatically extract all possible relationships from a given corpus, their results cannot directly be used in further applications. They can neither disambiguate mentions nor provide semantic information about the extracted relationships automatically.

We avoid labeling large amounts of training data and predefining a specific type of relationship by using a few examples of a target relationship for bootstrapping. This idea was first introduced by Brin [5] in the context of the DIPRE system, which focused on extracting relationships between authors and their corresponding book titles. Some other approaches were developed based on this bootstrapping strategy, e.g., Snowball [1] and StatSnowball [14].

We focus on reliably extracting business relationships between companies. By applying the semi-supervised algorithm, we can extract more complicated many-to-many relationships from large amounts of unlabeled data without requiring the expensive initial labeled data. A user only has to supply a very small number (3–5) of seeds to achieve good results, which makes our approach flexible to be applied to variant target relationships or data sources by simply provide another small seed set. Furthermore, our approach is able to determine the direction of asymmetric relationships. This enables us to directly use the generated results in subsequent applications.

3 Overview of our Approach

Figure 1 gives a high-level overview of our relationship extraction approach: Given some textual data and a seed set of multiple company pairs that occur as members of a particular relationship, our system outputs new company pairs participating in the same relationship type. As a preprocessing step we simplified the algorithm introduced by Zuo et al. [15] to recognize and link the mentions of companies to their corresponding Wikipedia pages.

Given these disambiguated company mentions we generate patterns from their contexts. To this end, we follow the intuition that if a company pair from the seed set co-occurs in the same sentence, it is likely that the context characterizes the relationship specified by the seed (see Section 4.1). Therefore, the sentences that contain two or more distinct companies are selected as the input for the relationship extraction phase. An example tagged sentence is "…[[Verizon Communications|Verizon]]'s acquisition of [[MCI Inc.|MCI]]", where the original mentions "Verizon" and "MCI" are separately linked to Verizon Communications and MCI Inc. From the contexts surrounding company pairs, we generate possible candidate extraction patterns that are likely to represent the target relationship (see Section 4.1). Suppose we are interested in the ownership_of relationship and the company pair (Verizon Communication, MCI Inc.) is contained in the seed set, then a candidate pattern *pattern* = ⟨COMP1, COMP2, acquisition of, →⟩ can be generated. The last element in *pattern* describes the direction of the ownership_of relationship (see Section 4.4). After generating a list of candidate patterns, we select the most promising ones according to the measurements to be introduced in Section 4.2.

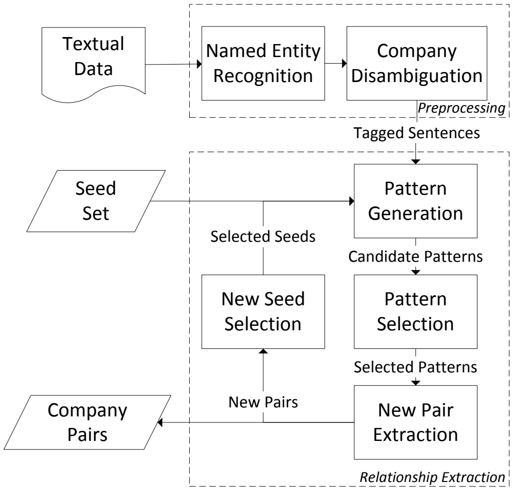

Fig. 1. Processing pipeline of our approach

We then use the selected patterns to discover new company pairs from the input. If the previous pattern *pattern* is selected, we can extract a new company pair (The Walt Disney Company, Pixar) from a sentence like "…after Disney's acquisition of Pixar Animation Studios". Afterwards, we select the most prominent newly extracted pairs to extend the seed set (see Section 4.3). We then iterate the procedure to extract new patterns using the extended seed set until no more new company pairs can be selected as seeds or the iteration number reaches a predefined limit. The company pairs that are extracted based on the current set of patterns are considered to participate in the same type of relationship as the target one. Our evaluation shows that this is indeed almost always the case regardless of the initial choice of seed pairs.

4 Extraction of Business Relationships

This section introduces our semi-supervised relationship extraction strategy, which iteratively extracts new company pairs that participate in a given target relationship.

4.1 Pattern generation

Generating the extraction patterns represents a crucial step in our approach. The context surrounding a company pair represents the main source to identify relationships occurring in textual data. To capture the key information that represents the relationship between two companies we extract the most determining phrases from the context as a key-phrase. This key-phrase is then used to generate a pattern.

Candidate pattern An extracted *pattern* includes two company variables COMP1 and COMP2, the key-phrase extracted from the context in between those companies, and a direction. We explain each of these parts in the following. From an example sentence "... YouTube, the video-sharing Web site owned by Google ... " we can generate the pattern $\langle \texttt{COMP1}, \texttt{COMP2}, \text{owned by}, \leftarrow\rangle$. By applying this pattern to this sentence we obtain the following instantiation of the pattern $\langle \texttt{YouTube}, \texttt{Google}, \text{owned by}, \leftarrow\rangle$, indicating that Google owns YouTube.

Key-phrase extraction The quality of a pattern depends on the key-phrase it contains. A good pattern should satisfy two criteria: characterize a single type of relationship (which in turn improves the precision of the extraction result) and be as general as possible (to extract many new company pairs). For this reason, it is beneficial to generalize the context and don't keep idiosyncratic key-phrases. The key-phrase should be as compact as possible, while maintaining the information in the context. Extracting patterns for business relationships in the news is particularly challenging since journalists are used to introduce the same type of business relationships using different writing styles spanning a relatively large context. This can be shown using the excerpt "... News Corporation, which owns a minority interest in DirecTV". In this sentence, we can easily figure out that News Corporation is one of the owners of DirecTV by finding the verb "owns" in the intermediary context. If we now use the entire context (i.e., ", which owns a minority interest in") between the two companies as a pattern to extract additional company pairs, we would find only very few since the pattern is not general enough. The problem can be solved by extracting the key-phrase "owns" that defines the ownership relationship. Thus we can conceptually simplify the original sentence to "New Corporation owns DirecTV". To this end, we developed a key-phrase extraction strategy to automatically extract the most determining phrases from the intermediary context. Intuitively, relationships in sentences are often conveyed by verbs or nouns. In [3] most of the binary relationships are indicated by four types of phrases, which cover over 86% of the cases. These key-phrase types are "Verb", "Noun+Prep.", "Verb+Prep.", and "Infinitive" located in between two entities in English text. To extract key-phrases from contexts, we apply the Stanford Part-Of-Speech(POS) Tagger [1].

[1] `http://nlp.stanford.edu/software/`

Based on the POS tags, we keep the phrases that match any of the four types above. We abandon the context if the containing verb is "to be", because it usually does not indicate any business relationship, or if the context contains multiple key-phrases.

4.2 Pattern selection

In each iteration, we generate candidate patterns based on the (extended) seed set. However, patterns that do not represent the target relationship might also end up in the candidate list. Therefore it is important to keep only the most representative patterns while filtering out unfavorable patterns. In the following, we introduce two strategies to select the best patterns.

Hit score Building on the intuition that patterns that frequently match company pairs in the seed set are likely to be representative ones, we introduce a *Hit* score for each pattern as follows,

$$\mathit{Hit}(\mathit{pattern}|\mathbf{Pair_{seed}}, \mathbf{S}) = \sum_{\mathit{pair}_i \in \mathbf{Pair_{seed}}} \sum_{s_j \in \mathbf{S}} [\mathit{match}(\mathit{pair}_i, p, s_j)] \tag{1}$$

Thus *Hit* is defined as the summation of how frequently a *pattern* matches a company $pair_i \in \mathbf{Pair_{seed}}$ in the set of input sentences $\mathbf{S}$. A pattern with a high *Hit* score denotes that the corresponding key-phrase is more likely to represent the target relationship. Given a list of candidate patterns that are sorted in descending order by their respective *Hit* score, we select the top-k ranked patterns to extend the set of the current extraction patterns.

Coverage score A good pattern should frequently be used in the context between different company pairs to describe a particular relationship. If the pattern can be extracted by using only one of the seed pairs, it is either too specific or it describes some other type of relationship between the corresponding companies. We introduce a Coverage (*Cov*) score, which represents the percentage of company pairs from the seed set that are able to generate this pattern.

$$\mathit{Cov}(\mathit{pattern}|\mathbf{Pair_{seed}}, \mathbf{S}) = \frac{\sum_{\mathit{pair}_i \in \mathbf{Pair_{seed}}} [\sum_{s_j \in \mathbf{S}} [\mathit{match}(\mathit{pair}_i, p, s_j)] > 0]}{|\mathbf{Pair_{seed}}|} \tag{2}$$

The *Cov* score of a pattern equals 1.0 when all seed pairs match the pattern at least once. All patterns that have a *Cov* score greater than a threshold τ are selected.

4.3 New seeds selection

We introduce a similar strategy for selecting newly extracted company pairs to extend the seed set. Using the selected patterns, we compute the *Hit* score for each of the extracted company pairs. We select the top-k pairs by their *Hit* scores. We can also compute the *Cov* score of an extracted company pair, which is the percentage of selected patterns that match the company pair in the text. In a similar fashion to the pattern selection, we extend the seed set by selecting the company pairs that have a greater *Cov* score than the same given threshold τ for pattern selection.

4.4 Direction of relationship

In Section 1 we introduced the challenge of determining the direction of asymmetric business relationships. Compared to the extraction of symmetric relationships, extracting asymmetric ones, such as supplier_of, ownership_of, and sued_by require not only the extraction of a new company pair occurring in the target relationship, but also the detection of its correct semantic direction.

Previous work, such as Snowball [1], naturally avoids this direction problem, since they focus on relationships that relate two objects of different entity types (i.e., organization, location). However, in our case, the entities are of the same type (i.e., company). Zhu et al. [14] present a similar challenge, e.g. an entity of type person e_1 is the husband of e_2. They solve this problem by manually adding new rules, such as IsHusband(e_1, e_2) $\Rightarrow$ IsWife(e_2, e_1), during their iterations.

We introduce an elegant strategy to automatically classify the direction of newly extracted relationships. The idea is to include the direction information already in the seed set. When the target relationship is asymmetric, the company pairs in the initial seed set must be specified by also providing the direction of the relationship. E.g., in the case of the ownership_of relationship, we specify a forward direction, denoting that the first company is the owner of the second.

Given this directed seed set, we can identify the direction of the generated patterns as follows: When two companies are mentioned in the same order as in the seed pair, the pattern is annotated with the same direction as the seed pair. Finally, the direction of a pattern is derived by assigning the direction that is more frequently marked. Table 1 in the evaluation section shows some examples of determined directions of patterns.

4.5 Multiple types of relationships

With our semi-supervised business relationship extraction approach, we can independently extract different relationship types by providing multiple initial seed sets each characterizing one type of relationship.

As mentioned in Section 1, different types of business relationships can exist between two companies at the same time. Therefore, the patterns generated from the seed set do not always represent the desired relationship type. Even worse, once a pattern that represents an undesired relationship type is selected, the following iterations can be negatively influenced in a way that they yield more and more irrelevant patterns, which leads to incorrect extraction results comparable to a topic drift in pseudo-relevance feedback methods. We can avoid this problem by assigning each pattern that is generated for multiple relationship types exclusively to one single type.

We followed the intuition that each pattern characterizes one kind of relationship and implemented a holistic pattern identification strategy by using the *Cov* score. In case the same pattern is generated for multiple relationship types, we exclusively assign the pattern to the type that yields the highest *Cov* score.

As a preliminary experiment to show the effect of this holistic strategy, we applied our approach to extract the ownership_of and partnership_with relationships at the same time. The selection of patterns and new seeds was made using the *Hit* score. By applying the holistic pattern identification strategy, most of the patterns, especially the top-ranked ones, characterize the partnership_with relationship. Without our holistic strategy, the top-ranked patterns (i.e., "stake in", "deal with", and "buy") mainly represent the ownership_of relationship. This problem was caused by a falsely selected pattern (i.e., "owned by"), which led to more and more patterns that characterize the ownership_of relationship.

5 Experiments

In our evaluation we focus on the extraction of an asymmetric relationships (i.e., ownership_of) from articles of the New York Times corpus.

5.1 NYTimes corpus and seeds

The full New York Times corpus contains 1,855,658 news articles, spanning a period of 20 years from Jan. 1987 to Jun. 2007. We observed that about 74% of all company pairs within a sentence occurred in the "Technology" and "Business" categories. Thus, we reduced our corpus to articles with at least one of those two labels. Our final corpus (called NYTimes from now on) consists of 359,459 articles.

An initial seed set serves as the input for our approach and predefines the relationship type we would like to extract. We investigated two different seed sets to evaluate their influence on the results. To this end, we generated a list of distinct company pairs that co-occur in the NYTimes corpus and sorted it in descending order by co-occurrence frequency. We manually labeled the relationship type for the first 100 pairs and then randomly selected five company pairs (*FreqSeed*) that share the ownership_of relationship from the top-100 list entries. Following this random selection strategy, we also generated a seed set called *InfreqSeed* from the top-1000 company pairs. Keep in mind that seed selection and our evaluation is based on a corpus dating from 1987 to 2007, resulting in relationships that might not hold today. *FreqSeed* contains company pairs, such as (AOL, Netscape), (Viacom, Viacom Media Networks), (Ford, Jaguar), (Time Warner, TBS), and (GE, NBC Sports), while *InfreqSeed* contains less frequently mentioned pairs, such as (Disney, ESPN), (IPC, Campbell Mithun), (GM, Saturn), (Chrysler, American Motors), and (Investcorp, Saks Fifth Avenue).

5.2 Experimental results

We first show which patterns were generated and then evaluate the quality of the actual business relationships we extracted.

Results of pattern generation Based on the two randomly generated seed sets we applied our approach to extract new company pairs that are also members of the ownership_of relationship. Table 1 shows the key-phrases of the selected patterns that are automatically generated by using *FreqSeed* and *InfreqSeed*. In this experiment, we applied the *Hit* score in each iteration for selecting the top-10 candidate patterns and the respective company pairs. The first column shows the key-phrases of the selected patterns. By using either *FreqSeed* or *InfreqSeed*, the extraction process terminates after

Table 1. Key-phrases of selected patterns for extracting ownership_of relationship

Extracted Patterns (key-phrase)	**Rank (Iteration)**						**Direction**
	FreqSeed			**InfreqSeed**			
	1	**2**	**3**	**1**	**2**	**3**	
unit of	1	1	1	4	1	1	←
parent of	–	4	2	–	4	2	→
owned by	2	2	3	1	3	4	←
part of	4	3	4	2	2	3	←
division of	5	5	5	3	5	5	←
owns	3	6	6	7	6	6	→
company of	–	–	7	–	9	8	→
acquisition of	7	7	8	6	7	7	←
subsidiary of	–	8	9	–	8	9	←
owner of	–	–	10	–	–	10	→
including	9	9	11	–	–	–	→
include	8	10	12	–	–	–	→
bought	10	11	13	9	11	12	→
acquired	6	12	14	5	12	13	→
buy	–	–	–	10	10	11	←
bought by	–	–	–	8	13	14	←

three iterations resulting in 14 selected patterns. These patterns are sorted in descending order by their *Hit* score. We also include the ranks of the patterns per iteration to show the changes that occur from iteration to iteration.

Further, Table 1 shows that most of the automatically generated key-phrases are typical phrases frequently used to describe an ownership_of relationship. Already in the first iteration, our approach can generate representative patterns. Differences between the two sets of generated patterns can be observed mainly in the tail. Thus, our approach is not particularly sensitive towards the chosen seed set (we made similar observations for various other seed sets, both in terms of size and co-occurrence frequency).

The last column in Table 1 contains the extracted direction of the patterns as determined by the strategy introduced in Section 4.4. Only 2 out of the 16 directions are incorrect Although the direction of these two patterns is classified incorrectly, most directions of the newly extracted company pairs, are identified correctly as the statistics in Section 5.2 show. This is because the direction of newly extracted company pairs is determined by multiple patterns.

Quality of extraction results We applied our approach using different settings for both pattern and seed selection to verify the extraction result. We conducted experiments with the *Hit* and *Cov* scores strategies introduced in Section 4. To show the effect of our key-phrase extraction strategy, we also executed our algorithm without using this strategy. In other words, we employed the original context to generated patterns, which is similar to previous work, e.g., [1, 5]. As a baseline, we select the most frequently co-occurred company pairs to check how many of them are in an ownership_of relationship.

We had to manually check relationships between company pairs, because no gold standard with known business relationships is available. The design of our approach is

Table 2. Average precision and precision values for the top-50, top-100, and top-200 extracted ownership_of relationships (including error analysis of the top-200 results)

Strategy	**P@50**	**P@100**	**P@200**	**Avg Prec**	**Error Type (Top 200)**			
					Rel.	**Dir.**	**Pre.**	**Sem.**
Baseline	30.0%	36.0%	30.5%	28.3%	-	-	-	-
Hit@10 w/o KP	18.0%	19.0%	20.0%	18.9%	139	4	4	13
Hit@5	94.0%	90.0%	88%	91.4%	14	0	8	2
Hit@10	94.0%	89.0%	87.5%	90.5%	9	4	7	5
Hit@15	94.0%	88.0%	85%	90.0%	9	8	7	6
Cov($\tau = 0.7$) w/o KP	20.0%	20.0%	23.0%	21.0%	135	3	3	13
Cov($\tau = 0.6$)	94.0%	87.0%	81.0%	88.7%	17	4	7	10
Cov($\tau = 0.7$)	**94.0%**	**90.0%**	**90.0%**	**91.9%**	8	1	6	5
Cov($\tau = 0.8$)	94.0%	88.0%	87.0%	90.6%	8	7	6	5

mainly concerned with achieving a high precision value because we aim to use it in the context of risk-analysis, which has only a small tolerance for incorrectly extracted information. Therefore, we mainly focus on evaluating the precision performance of our approach. We manually examined the top-200 most frequently extracted company pairs from each result set produced by our algorithm with different parameterizations.

Table 2 presents the evaluation results using the *FreqSeed* seed set to extract the ownership_of relationships. As this table shows, by applying *Cov ($\tau = 0.7$)* score, 90% of the top-200 extracted company pairs indeed participate in the ownership_of relationship. The performance of our approach, excluding the key-phrase extraction strategy, also shows the significant effect of including it. In comparison to the baseline, our approach can produce much better results.

Apart from the precision measure, we also present a detailed error analysis based on the top-200 extracted company pairs: The first error type is that company pairs that do not participate in an ownership_of relationship are extracted (Rel.). Another error case is that our approach extracted the correct company pair, but failed to identify the correct direction (Dir.). A third error case is caused by recognition or disambiguation errors made by the preprocessing steps (Pre.). An incorrect result can also be due to misinterpretation of the semantics (Sem.). E.g., one company finally canceled the plan of acquiring another one, such as the abandoned merger between EMI and Time Warner. Such events are covered by a series of New York Times articles, but our approach was unable to successfully capture the final cancellation of the deal. As the result shows, only around half of the incorrectly extracted relationships (i.e., Rel. and Dir.) are caused by our RE strategy.

Furthermore, according to the mechanism of our approach, when a relationship is mentioned in the given corpus more frequently, the probability that our approach can extract that relationship is higher. Thus, by including more documents the recall of our approach increases. We iteratively applied our approach (with the setting *Cov ($\tau = 0.7$)*) to an NYTimes corpus of increasing size, starting from 10 years of data up to 21 years. In Figure 2, the red line denotes the total number of tagged sentences after our preprocessing step. The blue bars show the accumulated count of extracted company pairs. As the figure shows, by enlarging the size of the dataset more unique aimed relationships can be extracted.

We compared our approach with a state-of-the-art distant learning approach developed by Zeng et al. [12]. They applied the piecewise convolutional neural networks with multi-instance learning for relationship extraction, which we refer as PCNNs[2]. In their experiments, the dataset[3] which contains the New York Times articles labeled with Freebase relationships was used. The sentences from 2005 − −2006 were used for training, while the ones from 2007 were used for testing. To compare the performance between PCNNs and our approach, we apply our approach on this dataset. As we have introduced in Section 1, Freebase only contains the major acquisitions of companies, which can be considered as the ownership_of relationship. However, all of the instances of the ownership_of relationship were mislabeled to be negative in the original dataset. Therefore, to compare PCNNs with our approach for extracting the ownership_of relationship, we had to relabel the training set according to the corresponding Freebase relationships (99 pairs are matched in the training set). Since only 14 Freebase relationships can be matched in the test set, we randomly picked and manually validated 100 company pairs (including 50 positives and 50 negatives) from the articles in 2007. For this specific type of relationship, PCNNs labels 5 pairs as positive, which are all correct. Our approach extracts 19 pairs, where 18 of them are correct. In this experiment, our approach outperforms PCNNs in both recall and F-measure.

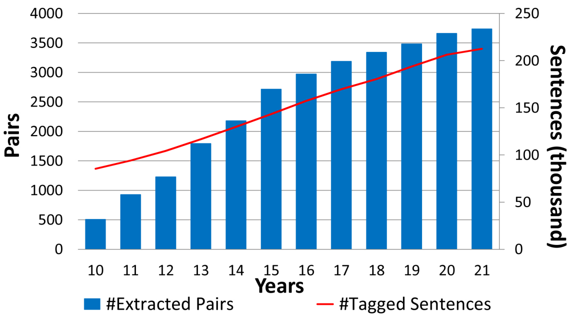

Fig. 2. The accumulated count of extracted company pairs from subsets of NYTimes corpus

Regarding efficiency, our approach can extract business relationships at a rate of about 650 documents per minute on a standard consumer PC, with most of the time spent on preprocessing. The efficiency can be further improved by implementing a distributed system to apply our approach as the strategy introduced in [9].

More detailed statistics as well as the annotated data are available online[4].

6 Conclusion and Future Work

The focus of this work was to efficiently extract complex business relationships from news articles. We are the first to focus on the class of many-to-many relationships. To this end, we proposed a relationship extraction approach that not only extracts new relationships from text but also indicates their direction in case of non-symmetric ones, such

[2] The original code is available online: http://www.nlpr.ia.ac.cn/cip/~liukang/publications.html

[3] http://iesl.cs.umass.edu/riedel/ecml/

[4] https://hpi.de/naumann/projects/knowledge-discovery-and-mining/business-relationship-extraction.html

as the ownership_of relationship. Another contribution is the holistic pattern identification strategy, which is used to avoid the semantic drift of generated extraction patterns while dealing with multiple business relationships simultaneously.

Further, we would like to include the duration and domain information of relationships. Moreover, the performance of our approach can be further improved by understanding the semantics of the underlying sentences to avoid incorrect extractions caused by misinterpretations.

References

1. Agichtein, E., Gravano, L.: Snowball: Extracting relations from large plain-text collections. In: Proceedings of the International Conference on Digital Libraries (DL). pp. 85–94 (2000)
2. Banko, M., Cafarella, M.J., Soderland, S., Broadhead, M., Etzioni, O.: Open information extraction for the web. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). pp. 2670–2676 (2007)
3. Banko, M., Etzioni, O.: The tradeoffs between open and traditional relation extraction. In: Proceedings of the Meeting of the Association for Computational Linguistics (ACL). pp. 28–36 (2008)
4. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the International Conference on Management of Data (SIGMOD). pp. 1247–1250 (2008)
5. Brin, S.: Extracting patterns and relations from the world wide web. In: The World Wide Web and Databases, pp. 172–183. Springer (1999)
6. Kambhatla, N.: Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In: Proceedings of the ACL 2004 on Interactive poster and demonstration sessions. p. 22 (2004)
7. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the Meeting of the Association for Computational Linguistics (ACL) and the International Joint Conference on Natural Language Processing of the AFNLP. pp. 1003–1011 (2009)
8. Vapnik, V.N.: Statistical learning theory, vol. 1. Wiley, New York (1998)
9. Wang, T., Min, H.: Entity relation mining in large-scale data. In: Database Systems for Advanced Applications: DASFAA 2015 International Workshops, SeCoP, BDMS, and Posters. p. 109 (2015)
10. Wu, F., Weld, D.S.: Open information extraction using Wikipedia. In: Proceedings of the Meeting of the Association for Computational Linguistics (ACL). pp. 118–127 (2010)
11. Zelenko, D., Aone, C., Richardella, A.: Kernel methods for relation extraction. The Journal of Machine Learning Research 3, 1083–1106 (2003)
12. Zeng, D., Liu, K., Chen, Y., Zhao, J.: Distant supervision for relation extraction via piecewise convolutional neural networks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1753–1762 (2015)
13. Zhou, G., Su, J., Zhang, J., Zhang, M.: Exploring various knowledge in relation extraction. In: Proceedings of the Meeting of the Association for Computational Linguistics (ACL). pp. 427–434 (2005)
14. Zhu, J., Nie, Z., Liu, X., Zhang, B., Wen, J.R.: StatSnowball: a statistical approach to extracting entity relationships. In: Proceedings of the International World Wide Web Conference (WWW). pp. 101–110 (2009)
15. Zuo, Z., Kasneci, G., Gruetze, T., Naumann, F.: BEL: Bagging for entity linking. In: Proceedings of the International Conference on Computational Linguistics (COLING). pp. 2075–2086 (2014)