ZORE: A Syntax-based System for Chinese Open Relation Extraction

Likun Qiu and Yue Zhang
Singapore University of Technology and Design, Singapore
qiulikun@gmail.com, yue_zhang@sutd.edu.sg

Abstract

Open Relation Extraction (ORE) overcomes the limitations of traditional IE techniques, which train individual extractors for every single relation type. Systems such as ReVerb, PATTY, OLLIE, and Exemplar have attracted much attention on English ORE. However, few studies have been reported on ORE for languages beyond English. This paper presents a syntax-based Chinese (Zh) ORE system, ZORE, for extracting relations and semantic patterns from Chinese text. ZORE identifies relation candidates from automatically parsed dependency trees, and then extracts relations with their semantic patterns iteratively through a novel double propagation algorithm. Empirical results on two data sets show the effectiveness of the proposed system.

1 Introduction

Traditional Information Extraction (IE) systems train extractors for pre-specified relations (Kim and Moldovan, 1993). This approach cannot scale to the web, where target relations are not defined in advance. Open Relation Extraction (ORE) attempts to solve this problem by shallow-parsing-based, syntax-based or semantic-role-based pattern matching without pre-defined relation types, and has achieved great success on open-domain corpora ranging from news to Wikipedia (Banko et al., 2007; Wu and Weld, 2010; Nakashole et al., 2012; Etzioni et al., 2011; Moro and Navigli, 2013). Many NLP and IR applications, including selectional preference learning, commonsense knowledge and entailment rule mining, have benefited from ORE (Ritter et al., 2010). However, most existing ORE systems focus on English, and little research has been reported on other languages. In addition, existing ORE techniques are mainly concerned with the extraction of textual relations, without trying to give semantic analysis, which is the advantage of traditional IE.

Our goal in this paper is to present a syntax-based Chinese (Zh) ORE system, ZORE, which extracts relations by using syntactic dependency patterns, while associating them with explicit semantic information. An example is shown is Figure 1, where the relation *(奥巴马 (Obama) 总统 (President) , Pred[毕业 (graduate)],* 哈佛 *(Harvard)* 法学院 *(Law School))* is extracted from the given sentence "奥巴马 *(Obama)* 总统 *(President)* 毕业 *(graduate)* 于 *(from)* 哈佛 *(Harvard)* 法学院 *(Law School)*", and generalized into the syntactic-semantic pattern {*nsubj-NR(Af) Pred[毕业 (graduate)] prep-于 (from) pobj-NN(Di)*}. Here, *Af* and *Di* stand for human and institution, respectively, according to a Chinese taxonomy *Extended Cilin* (Che et al., 2010).

Rather than extracting binary relations and then generalizing them into semantic patterns, which most previous work does (Mausam et al., 2012; Nakashole et al., 2012; Moro and Navigli, 2012; Moro and Navigli, 2013), we develop a novel method that extracts relations and patterns simultaneously. A double propagation algorithm is used to make relation and pattern information reinforce each other, so that negative effects from automatic syntactic and semantic analysis errors can be mitigated. In this way, semantic pattern information is leveraged to improve relation extraction.

We manually annotate two sets of data, from news text and Wikipedia, respectively. Experiments on both data sets show that the double propagation algorithm gives better precision and recall compared to the baseline. To our knowledge, we are one of the first to report empirical results on Chinese ORE. The ZORE system, together with the two sets of test data we annotated, and the sets of 5 million relations and 344K semantic patterns extracted from news and Wikipedia, is freely re-

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1870–1880,
October 25-29, 2014, Doha, Qatar.

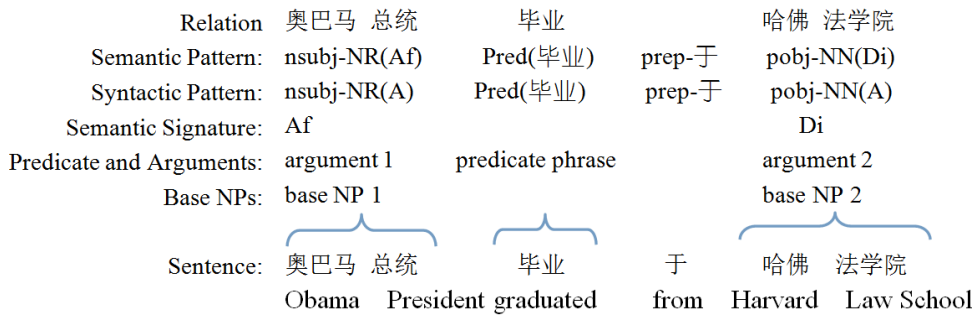

Figure 1: A sample sentence analyzed by ZORE.

leased[1].

2 Basic Definitions for Open Information Extraction

ZORE is applied to web text to extract general relations and their semantic types. Our definition of relations follow previous work on ORE (Moro and Navigli, 2013), but with language-specific adjustments. In this section, we use the sentence in Figure 1 as an instance to describe the basic definitions for ZORE.

Definition 1 (predicate phrase) A ***predicate phrase*** is a sequence of words that contains at least one verb or copula, and governs one or more noun phrases syntactically. For instance, a predicate phrase for the sentence in Figure 1 is "毕业 *(graduate)*". Following Fader et al. (2011), Mausam et al. (2012) and Nakashole et al. (2012), in case of light verb constructions, the verb and its direct object jointly serve as predicate phrase. We do not include prepositions into the predicate phrases.

Definition 2 (argument) An ***argument*** is a base noun phrase governed by a predicate phrase *directly* or *indirectly* with a preposition. For instance, "奥巴马 (Obama) 总统 (President)" and "哈佛 (Harvard) 法学院 (Law School)" are two arguments of the predicate phrase "毕业 *(graduate)*".

Definition 3 (relation) A ***binary relation*** is a triple that consists of the predicate phrase *Pred* and its two arguments *x* and *y*. Accordingly, an *n-ary relation* contains *n* arguments. For instance, the sentence in Figure 1 contains the binary relation *(奥巴马 (Obama) 总统 (President), Pred[毕业 (graduate)], 哈佛 (Harvard) 法学院 (Law School))*. In English, the two arguments of a binary relation are usually positioned on the left and right of *Pred*, respectively. Hence, shallow patterns are highly useful for English relation extraction (Banko et al., 2007). In Chinese, however, the two arguments can be both on the left, both on the right or one on the left and one on the right of the predicate, and the resulting binary relation can be either *(x, y, Pred)*, *(Pred, x, y)* and *(x, Pred, y)*, depending on the sentence. This makes the detection of relation phrases more complicated.

Definition 4 (syntactic pattern) A ***syntactic pattern*** is the syntactic abstraction of a relation. A relation can be generalized into the combination of words, POS-tags and syntactic dependency labels (Nakashole et al., 2012). For instance, the syntactic pattern of the sentence in Figure 1 is {*nsubj-NR(A) Pred[毕业] prep-于 pobj-NN(A)*}. It consists of four sub-patterns. The first, *nsubj-NR(A)*, denotes that the current phrase acts as the subject of the predicate phrase with the POS-tag *NR* (proper nouns). Here, "*(A)*" means that the phrase is an argument of the extracted relation. The second sub-pattern denotes that the predicate phrase of the example is "毕业 *(graduate)*". Note that the words between the predicate and arguments (e.g., *prep-于*) are included into the pattern directly (Nakashole et al., 2012; Mausam et al., 2012).

Definition 5 (semantic signature) The ***semantic signature*** of a relation consists of the semantic categories of the arguments. The semantic signature of Figure 1 is *(Af, Di)*, where *Af* and *Di* denotes *human* and *institute*, respectively.

Definition 6 (semantic pattern) A ***semantic pattern*** is the semantic abstraction of a relation. It is the combination of a syntactic pattern and a semantic signature. For instance, the syntactic pattern {*nsubj-NR(A) Pred[毕业] prep-于 pobj-NN(A)*}, combined with the semantic signature *(Af, Di)*, results in the semantic pattern {*nsubj-NR(Af) Pred[毕业] prep-于 pobj-NN(Di)*}.

[1]https://sourceforge.net/projects/zore/

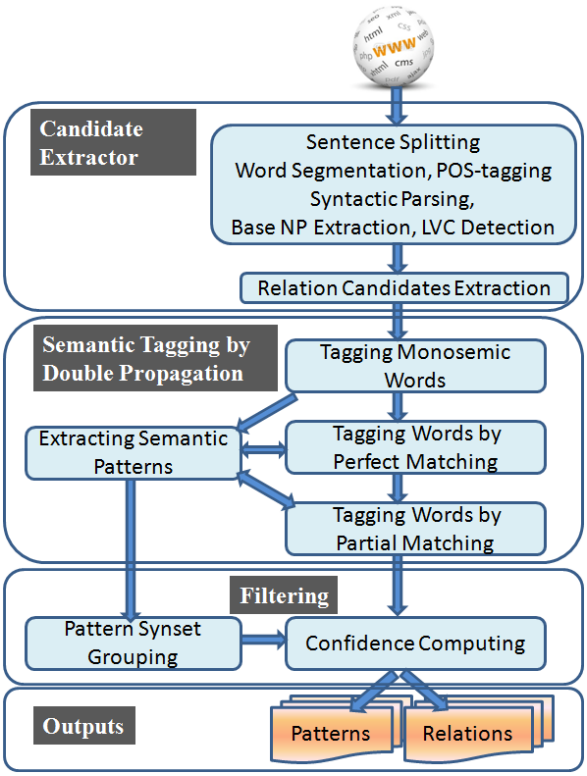

Figure 2: Architecture of ZORE.

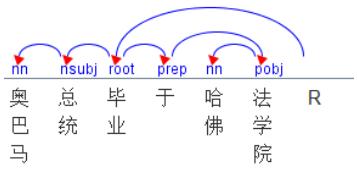

Figure 3: Parsing result of the example sentence in Figure 1, in Stanford dependencies.

3 ZORE

The architecture of ZORE is shown in Figure 2. It consists of three components. The first is a relation candidate extractor, which consumes input text and performs sentence segmentation, word segmentation, POS tagging, syntactic parsing, base NP extraction, light verb structure (LVC) detection and relation candidate extraction. The output is a set of relation candidates. The second component tags relations and extracts semantic patterns by a double propagation algorithm. In the third component, extracted patterns are grouped into synsets, and relations are filtered by confidence scores.

3.1 Extracting Relation Candidates

3.1.1 Parsing and Base NP Extraction

ZORE analyzes the syntactic structures of input texts by applying a pipeline of NLP tools. Each sentence is segmented into a list of words by using the Stanford segmenter (Chang et al., 2008), and parsed by using ZPar (Zhang and Clark, 2011), with POS tags and constituent structures by the CTB standard (Xue et al., 2005). The resulting constituent trees are transformed into projective trees with Stanford dependencies by using the Stanford parser (Chang et al., 2009). Figure 4 shows the parse tree of the sentence in Figure 1.

Next, base noun phrases (NPs) are extracted from the dependency tree. Here a base NP is a maximum phrase whose words can only have POS from the first row of Table 1. The head word of a base NP can be either a noun, a pronoun, a number or a measure word (the second row of Table 1). The dependency labels within a base NP can only be from the third row of Table 1. Obviously, a base NP does not contain other base NPs, and is also not contained by any other base NP.

3.1.2 Detecting Light Verb Constructions

In linguistics, a *light verb* is a verb that has little semantic content of its own, and typically forms a predicate with a noun (Butt, 2003). Example predicates by light verb constructions (LVC) include "*is a capital of*" and "*claim responsibility for*", where "is" and "claim" are light verbs. Improper handling of LVC can cause a significant problem by uninformative extractions (Etzioni et al., 2011). For example, if "is" and "claim" are extracted as predicates, the resulting relations (such as (*Hamas, claimed, responsibility*) from the sentence "Hamas claimed responsibility for the Gaza attack") might not bare useful information. ReVerb (Etzioni et al., 2011) handles this problem by hard syntactic constraints, taking the noun phrase (e.g., responsibility) between a verb phrase (e.g., "claim") and a preposition (e.g., "for") as a part of the predicate phrase rather than an argument, leading to the relation *(Hamas, claimed responsibility for, the Gaza attack)*.

In Chinese, LVCs are highly frequent and should be handled properly in order to ensure that the extracted relations are informative. However, the syntactic constraints in ReVerb can not be transferred to Chinese directly, because the word orders of English and Chinese are quite different.

	Labels
Base NP modifier	*NN (common noun), M (measure word), CD (cardinal number), OD (ordinal number), PN (pronoun), NR (proper noun), NT (temporal noun), JJ (other noun-modifier), or PU (punctuation)*
Base NP head	*NN (common noun), M (measure word), CD (cardinal number), OD (ordinal number), PN (pronoun), NR (proper noun), NT (temporal noun)*
Labels in base NPs	*nn (noun compound modifier), conj (conjunct), nummod (number modifier), cc (coordinating conjunction), clf (classifier modifier), det (determiner), ordmod (ordinal number modifier), punct (punctuation), dep (other dependencies), or amod (adjectival modifier)*
Labels from base NPs to predicate phrase	*nsubj (nominal subject), conj (conjunct), dobj (direct object), advmod (adverbial modifier), prep (prepositional modifier), pobj (prepositional object), lobj (localizer object), range (dative object that is a quantifier phrase), tmod (temporal modifier), plmod (localizer modifier of a preposition), attr (attributive), loc (localizer), top (topic), xsubj (controlling subject), ba ("ba" construction), nsubjpass (nominal passive subject)*

Table 1: Constraints on POS-tags and dependency labels. Labels in the top three rows are used for base NP extraction, while labels in the last row for traversing from a base NP to the predicate phrase.

In Chinese, prepositions acting as the modifier of a verb can be on both the left and right of the verb. For instance, the sentence "奥巴马 *(Obama)* 总统 *(President)* 于 *(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 毕业 *(graduate)*" is a paraphrase of the sentence in Figure 1, with the preposition 于 *(from)* on the left of the predicate phrase.

Chinese LVCs can be classified into two types, which we refer to as dummy-LVCs and common LVCs, respectively. For the first type, the predicate is a dummy verb such as "进行 *(do)*" and "予以 *(give)*", which has a noun phrase as its object. Since dummy verbs in Chinese are a closed set, we detect this type of LVCs (such as *"进行 (do) 会谈 (talk)"*) by finding the dummy verb from a lexicon. For the second type of LVCs, the predicate is a common verb, which has a nominalized structure or a common noun as its object. For instance, "展开 *(launch)* 调查 *(investigation)*" belongs to this type of construction.

Common LVCs are more difficult to detect than dummy-LVCs. We detect common LVCs by the context. Besides the NPs in the LVC itself, a common LVC typically governs two NPs, with the latter being connected to the predicate phrase by an LVC-related preposition such as "对 (for), 对于 (for), 针对 (for), 向 (to), 同 (with), 与 (with), 和 (with) ". Based on the observation, a basic idea of identifying common LVCs is to find verb-object structures that frequently co-occur with a LVC-related preposition in a large-scale corpus parsed automatically. For a given verb-object v, let f^v and f^p denote the frequency of v and the frequency of v co-occurring with an LVC-related preposition, respectively. We define the statistical strength of v to be an LVC as the ratio f^p/f^v. If the statistical strength of v exceeds a threshold t^{lvc}, we identify v as a LVC. Table 2 illustrates some high-frequency LVCs extracted by the method automatically.

3.1.3 Extracting Relation Candidates

ZORE tries to extract relation candidates from sentences that contain two or more base NPs. Given two base NPs, we traverse the dependency tree to obtain the shortest path that connects them. The path can contain only dependency labels in the fourth row of Table 1, and should contain at least one of the labels from "nsubj" and "dobj" to ensure that a predicate phrase is included in the path. If such a path is acquired, other base NPs governed by the same predicate phrase are included into the target relation, resulting in a n-ary relation candidates with each base NP being an argument. According to the predicate phrase, relation candidates can be classified into the following classes.

Common and dummy LVC relations. In this type of relations, the predicate phrase of the path is an LVC (e.g., a light verb and a nominal object). The two base NPs can be the subject or prepositional object of the light verb. For instance, in the sentence "霍迪尼 *(Houdini)* 对 *(to)* 我的 *(my)* 事业 *(career)* 有 *(have)* 很大 *(big)* 影响 *(influence)*", "有 *(have)*"and "影响 *(influence)*" are combined into a common LVC and taken as the predicate phrase, resulting the relation *(霍迪尼 (Houdini), Pred[有 (have) 影响 (influence)], 我的 (my) 事业 (career))*. In the corresponding English sentence, the predicate phrase "*be a big influence in*" is also an LVC structure.

Verb relations. In this type of relations, a verb acts as the predicate phrase. For instance, the relation *(奥巴马 (Obama) 总统 (President), Pred[毕业 (graduate)], 哈佛 (Harvard) 法学院 (Law School))* extracted from the sentence in Figure 1 is a typical verb relation.

Relative-clause relations. In an relative-clause

Verb	Noun
进行 (do) (*)	发行 (distribution), 分析 (analysis), 收集 (collection), 修改 (modification), 访问 (visit), 处罚 (punishment)
有 (have) (*)	影响(effect), 贡献 (contribution), 兴趣 (interest), 帮助 (help), 认识 (understanding), 期望 (expectation)
产生 (generate) (**)	影响 (effect), 兴趣 (interest), 怀疑(doubt), 冲击 (shock), 好感 (good feeling), 恐惧 (fear)
造成 (cause) (**)	影响 (effect), 破坏(destruction), 伤害 (harm), 威胁 (threat), 压力 (pressure), 干扰 (distraction)
表示 (express) (**)	满意 (satisfaction), 欢迎 (welcome), 尊重 (respect), 担忧 (worry), 哀悼 (mourning), 感谢 (gratitude)
展开 (launch) (**)	调查 (investigation), 攻击 (attack),攻势 (offensive), 批评 (criticism), 批判 (negotiation), 诉讼 (lawsuit)

Table 2: Instances of dummy-LVCs (*) and common LVCs (**). A verb in the left column is combined with a noun in the right column to form an LVC, which serves as the predicate phrase.

relation, the head word is a noun, modified by an relative clause, but acting as an argument of the predicate of the relative clause semantically. The sentence "毕业 *(graduate)* 于 *(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 的 *(de, an auxiliary word)* 奥巴马 *(Obama)* 总统 *(president)*" is a paraphrase of the sentence in Figure 1, with the same predicate phrase and arguments. However, the relation extracted from this phrase is an relative-clause relation *(Pred[*毕业 *(graduate)],* 哈佛 *(Harvard)* 法学院 *(Law School),* 奥巴马 *(Obama)* 总统 *(president))*, which belongs to the same pattern synset as the relation of Figure 1.

3.2 Semantic Tagging by Double Propagation

The basic idea of our approach is to identify relations and patterns iteratively through semantically tagging the head words of arguments in relation candidates. Given a set of relation candidates and a semantic taxonomy, the propagation consists of three steps. In Step 1, monosemic arguments in candidate relations are tagged with a semantic category, such as *Af* and *Di*, to obtain semantic patterns. In Step 2 and Step 3, untagged ambiguous and unknown words are tagged by perfect matching and partial matching, respectively. In the end of each step, semantic patterns are generalized from extracted and tagged relations, and then used to help relation tagging in the next step. Because of the two-way information exchange, we call this method *double propagation*. The method can also be treated as similar to bootstrapping (Yangarber et al., 2000; Qiu et al., 2009).

3.2.1 Step 1: Tagging Monosemic Arguments

Each argument in a relation candidate is a base NP. Since base NPs are endocentric, we can take the semantic category of the head word of a base NP as the semantic category of the base NP. In a taxonomy, each word is associated with one or more semantic categories. In this step, however, only monosemic words are tagged, while both ambiguous words and unknown words are left untagged.

Most named entities are not included in the taxonomy. However, after POS-tagging, most of them are detected as NR (proper noun). As a result, they are taken as ambiguous words that can be person names, organization names or location names. The named entities that are not included in the taxonomy are tagged in Steps 2 and 3.

After this step, all the arguments in some relation candidates have been tagged with semantic categories. We refer to these relation candidates as *tagged relation candidates*, and the remaining relation candidates as *untagged relation candidates*. Tagged relation candidate are generalized into semantic patterns, consisting of syntactic patterns and semantic signatures, as illustrated in Figure 1 and Section 2. We call the set of resulting semantic patterns Set^{SemPat}.

3.2.2 Step 2: Tagging by Perfect Pattern Matching

In this step, the arguments in the untagged relation candidates are tagged by semantic pattern matching. Given an untagged relation candidate *r*, we acquire a set of *possible* semantic categories for each argument with an *ambiguous* head word. For the arguments with *unknown* head words, we acquire a set of *possible* semantic categories according to their characters. Qiu et al. (2011) demonstrate that 98% Chinese words have at least one synonym, which shares at least one character. For Chinese nouns, the set of synonyms usually shares the last one or two characters. According to this, our strategy for acquiring possible semantic categories for an unknown word is as follows.

Given an unknown word w^u, if we find a known word w^k that shares the last two character with w^u, the semantic categories of w^k will be used as the possible semantic categories of w^u. Otherwise, if we find a known word w^k that share the last one character with w^u, the semantic categories of w^k will be used as the possible categories of w^u.

We then acquire possible semantic signatures of untagged relation candidates, of which all the arguments are tagged with possible semantic categories. As in Step 1, we generalize relation *r* into a syntactic pattern pat^{syn}, and then combine pat^{syn} with each *possible* semantic signature of *r* to generate *possible* semantic patterns. In case one or more possible semantic patterns of *r* exist in Set^{SemPat}, if the highest frequency of these patterns is above a threshold t^{sem}, the corresponding pattern will be taken as the semantic pattern of *r*, from which we infer the semantic signature for *r* and then the semantic category for the head word of each argument of *r*. After this step, the frequency of each semantic pattern in Set^{SemPat} is updated according to the newly tagged relation candidates.

3.2.3 Step 3: Tagging by Partial Pattern Matching

In this step, we tag the ambiguous and unknown words by partial matching rather than perfect matching of the whole semantic pattern. This can be treated as a back-off of the last step.

We first split *n*-ary semantic patterns in Set^{SemPat} into binary semantic patterns, and calculate their frequencies. Second, we split each untagged relation candidate *r* into several binary sub-relations and then search for the corresponding semantic patterns as in Step 2 — for each binary sub-relation, we obtain a binary semantic signature with the highest frequency. By combining the binary semantic signatures, we obtain one *n*-ary semantic signature for *r*, based on which all the unknown and ambiguous words can be tagged with a semantic categories. If all the arguments of a relation candidate *r* are tagged, *r* is treated as tagged. Finally, according to the newly tagged relations, statistics in Set^{SemPat} are updated.

3.3 Grouping Patterns into Synsets

In this step, we group semantic patterns from Set^{SemPat} into *pattern synsets*, based on a single-pass clustering process (Papka and Allan, 1998). Given two semantic patterns $SemPat_i$ and $SemPat_j$, we refer to their corresponding syntactic pattern, semantic signature and predicate phrases as $SynPat_i$ and $SynPat_j$, $SemSig_i$ and $SemSig_j$, $Pred_i$ and $Pred_j$, respectively. Not taking the predicate phrase into account, $SynPat_i$ and $SynPat_j$ are identical, and we call them *loosely identical* ($\approx$).

The algorithm in Figure 4 is used to group

if $Pred_i$ = "是 (is)" **or** $Pred_j$ = "是 (is)": **return** *false*.
else if ARGCOUNT($SemPat_i$) = 2 **and** ARGCOUNT($SemPat_j$) = 2 **and** SEMCAT(arg_1) = SEMCAT(arg_2):
  if $SynPat_i \approx SynPat_j$ **and** ISSONONYM ($Pred_i$, $Pred_j$): **return** *true*.
  else if $Pred_i = Pred_j$: **return** *true*.
  else: **return** *false*.
else if $Pred_i = Pred_j$ **and** $SemSig_i = SemSig_j$: **return** *true*.
else if ISSYNONYM($Pred_i$, $Pred_j$) **and** $SemSig_i = SemSig_j$ **and** $SynPat_i \approx SynPat_j$: **return** *true*.
else: **return** *false*.

Figure 4: Algorithm for pattern synset grouping.

Type	Feature	Weight
Base	*r* covers all words in *c*	0.96
Base	There are commas within *r*	-0.47
Base	LENGTH(*r*)<10 words	0.35
Base	10 words≤LENGTH(*r*)<20 words	0.11
Base	20 words≤LENGTH(*r*)	-1.06
Base	COUNT(*arguments*)=2	0.14
Base	COUNT(*arguments*)=3	0.33
Base	COUNT(*arguments*)=4	-0.60
Base	COUNT(*arguments*)> 4	-0.46
SemPat	Being tagged in Step 3	0.87
SemPat	Being tagged before Step 3	0.75
SemPat	50≤SIZE(*SemPat*) and untagged	-0.05
SemPat	50≤SIZE(*SemPat*) and tagged	0.65
SemPat	10≤SIZE(*SemPat*)<50 and untagged	-0.16
SemPat	10≤SIZE(*SemPat*)<50 and tagged	0.39
SemPat	5≤SIZE(*SemPat*)<10 and untagged	-0.22
SemPat	5≤SIZE(*SemPat*)<10 and tagged	0.36
SemPat	SIZE(*SemPat*) <5 and untagged	-0.92
SemPat	SIZE(*SemPat*) <5 and tagged	-0.64

Table 3: Features of the logistic regression classifier with weights trained on Wiki-500 dataset.

patterns, where ARGCOUNT($SynPat_j$) denotes the number of arguments in $SemPat_i$, SEMCAT(arg_1) indicates the semantic category of the first argument, and ISSYNONYM ($Pred_i$, $Pred_j$) returns whether two predicates are synonyms. In similarity-based single-pass clustering, the topic excursion problem is common (Papka and Allan, 1998). But since our similarity measure is symmetric, we do not suffer from this problem.

3.4 Computing the Confidence for Relations

Without filtering, the extraction algorithm in the previous sections may yield false relations. Following previous ORE systems, we make a balance between recall and precision by using a confidence threshold (Fader et al., 2011). A logistic regression classifier is used to give a confidence score to each relation, with features shown in Table 3. In the table, *c*, *r*, *arguments* and *SemPat* denote

Dataset	Source	#Sen	#Rel
Wiki-500	Chinese Wikipedia	500	561
Sina-500	Sina News	500	707

Table 4: Annotated relation datasets.

clause, relation, arguments in a relation, and semantic pattern, respectively. LENGTH(r), COUNT(*arguments*) and SIZE(*SemPat*) indicate the number of words in r, the number of arguments in r, and the number of relations that belong to the same semantic pattern *SemPat* as r. Because semantic patterns from the double propagation algorithm are used as features in the classifier, they participate in relation extraction also. Their effect on relation extraction can directly demonstrate the effectiveness of double propagation.

4 Experiments

4.1 Experimental Setup

We run ZORE on two difference corpora: the Chinese edition of Wikipedia (Wiki), which contains 4.3 million sentences (as of March 29, 2014), and a corpus from the Sina News archive (Sina News), which includes 6.1 million sentences from January 2013 to May 2013. The sentences that do not end with punctuations are filtered. The Chinese taxonomy *Extended Cilin*[2] (*Cilin*) (Che et al., 2010) is used to give semantic categories for each word. *Cilin* contains 77,492 Chinese words, organized into a five-level hierarchy. There are 12 categories in the top level, 94 in the second and 1492 in the third. In this paper, the second level is used for semantic categories. We create two test sets, containing 500 sentences from Wiki and 500 sentences from Sina News, respectively (see Table 4), annotated by two independent annotators using the annotation strategy of Fader et al. (2011). The thresholds t^{lvc} and t^{sem} for pattern matching are set as 0.4 and 5, tuned on 100 sentences from Wiki-500 dataset, respectively.

4.2 Evaluation of Relation Extraction

First, we compare ZORE with a baseline system to illustrate the effectiveness of the double propagation algorithm. The baseline system does not have the double propagation tagging component in Figure 2, using the logistic regression classifier in Section 3.4.1 with the 9 base features to filter extracted relation candidates. It is similar to the architecture of ReVerb (Fader et al., 2011). We

[2]http://ir.hit.edu.cn/demo/ltp/Sharing_Plan.htm

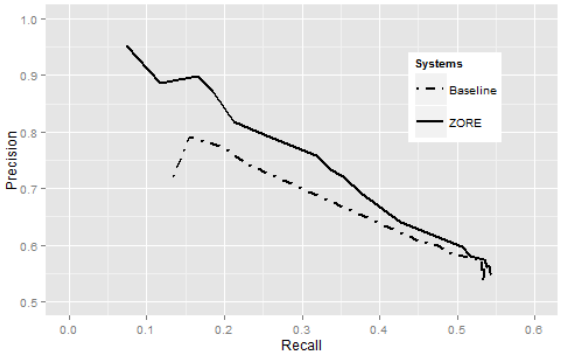

Figure 5: Performance on Wiki.

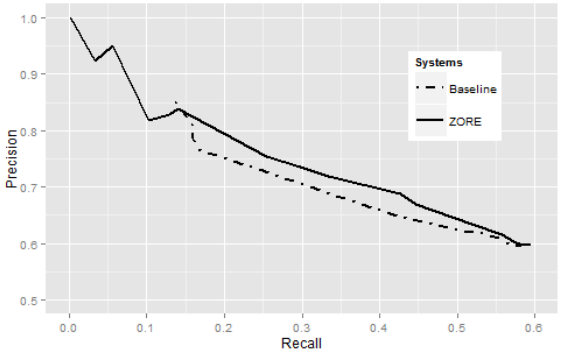

Figure 6: Performance on Sina News.

measure the precision and recall of the extracted relations. An extracted relation is considered correct only when the predicate phrase and all the arguments match the the gold set. On each data set, we perform 5-fold cross-validation test and take the average as the final precision and recall.

Figures 5 and 6 show the comparison of the two systems on Wiki and Sina News, respectively. On Wiki, ZORE has higher precision than the baseline at all levels of recall. When the recall is 0.3, the precision of ZORE is 0.77, 0.11 higher than the baseline. The result on Sina News is similar. The second column of Table 3 shows the weights of all features trained on the Wiki data set, which indicates that the semantic pattern features can give a positive effect on relation filtering.

Second, we compare the intermediate results at Steps 1, 2, and 3 in Section 3.2, respectively. The precision, recall and F1 of the three steps with different numbers of Wiki sentences (from 10K to 5M sentences) are shown in Table 5. This figure shows that Step 2 achieves higher precision than Step 1 at all levels of recall, indicating that the word sense tagging method in step 2 is useful for

Sentences	Step 1			Step 2			Step 3		
	P	R	F1	P	R	F1	P	R	F1
10K	0.947	0.032	0.062	**0.960**	0.043	0.082	0.933	0.075	0.139
50K	0.894	0.075	0.138	0.922	0.105	0.189	0.907	0.139	0.241
100K	0.897	0.093	0.169	0.924	0.130	0.228	0.909	0.160	0.272
200K	0.901	0.114	0.202	0.926	0.157	0.268	0.892	0.191	0.315
500K	0.891	0.146	0.251	0.909	0.196	0.322	0.860	0.230	0.363
1M	0.860	0.164	0.275	0.885	0.219	0.351	0.842	0.248	0.383
2M	0.797	0.182	0.296	0.819	0.250	0.383	0.788	0.278	0.411
3M	0.784	0.187	0.302	0.802	0.253	0.385	0.778	0.282	0.414
4M	0.739	0.178	0.287	0.801	0.258	0.390	0.778	0.287	0.419
5M	0.779	0.189	0.304	0.798	0.260	0.392	**0.768**	**0.289**	**0.420**

Table 5: Accuracies on different numbers Wiki sentences.

a significant boost of recall, together with a little improvement in precision. In particular, Step 2 can extract about 20% relations with relatively high precision (about 90%). The result of Step 3 is better to that of Step 2 in terms of F1-measure, with the highest F1-measure achieved by this step.

4.3 Evaluation of Patterns

ZORE acquires 122K and 222K patterns from Wiki and Sina News, clustered into 59K and 118K pattern synsets, respectively. The frequency distribution of the Wiki patterns is shown in Figure 7, which conforms to Zipf's law.

To assess the accuracy of pattern extraction, we rank the extracted patterns by the size, and evaluated the precision of the top 100 and a random set of 100 pattern synsets. Two annotators were shown a pattern synset with its semantic signature and a few example relations, and then asked to judge whether it indicates a valid semantic relation or not. The results are shown in Table 6. The averaged precision is 92% for the top 100 set, and 85% for the random 100 set.

The patterns in a pattern synset can be taken as paraphrases (Barzilay and Lee, 2003). We observe that two synonymous patterns might differ in three aspects. First, two patterns can differ by the predicates, which are synonyms. For instance, the verbs "担任, 当, 任, 出任, 为, 做" are synonyms, meaning "to hold the appointment of". Second, two patterns in the same synset can belong to different syntactic patterns, and therefore are paraphrases in the syntactic level. For instance, the semantic patterns of the two sentences "毕业 *(graduate)* 于 *(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 的 *(de, an auxiliary word)* 奥巴马 *(Obama)* 总统 *(president)*" and "奥巴马 *(Obama)* 总统 *(president)* 从*(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 毕业 *(graduate)*" are both synonymous to that of the sentence in Figure 1; all the three patterns are found in the same synset obtained by ZORE. Third, two patterns can differ only by the POS-tag. For instance, "奥巴马 *(Obama)* 从*(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 毕业 *(graduate)*" and "那个 *(That)* 律师 *(attorney)* 从*(from)* 哈佛 *(Harvard)* 法学院 *(Law School)* 毕业 *(graduate)*" are synonyms with different POS-tags for the first argument (i.e. N-R and NN). According to the grouping algorithm in Section 3.3, all the three types of paraphrases are grouped in a pattern synset, which makes some synsets very large. The largest synset contains 110 patterns, while the top 100 synsets contain more than 20 patterns.

4.4 Error analysis

We analyze the incorrect extractions (precision loss) and missed correct relations (recall loss) returned by Step 2, running on 500K sentences. Table 7 summarizes the types of correct relations that are missed by ZORE. 40% missed relations are due to the minimum frequency constraint on semantic patterns, which is used for a balance between precision and recall. Another main source of failure is the incorrect identification of the predicate phrase due to parsing errors, which account for 37% of the total errors. Other sources of failures include redundant arguments and segmentation errors. Most redundant arguments are related to prepositions such as "按照 (according to)" and "根据 (on the basis of)". For instance, in the sentence "按照 (according to) 这个 (the) 观点 (point of view) ， (,) 根本 (fundamental) 问题 (problem) 是 (is)", an incorrect binary relation *(这个 (the) 观点 (point of view), 根本 (fundamental) 问题 (problem), Pred[是 (is)])* is extracted, because the prepositional object "这个 (the) 观点 (point of view)" is tagged as an argument of the predicate phrase "是 (is)".

Table 8 summarizes the major types of incor-

Corpus	Patterns	Synsets	Top100	Random100
Wiki	122,723	59,298	0.93	0.87
Sina	222,773	118,923	0.91	0.83

Table 6: Precision of pattern synsets.

40%	Relations filtered by semantic pattern constraint
37%	Could not identify correct predicates because of preprocessing errors
12%	Too many arguments because of parsing errors
11%	Segmentation and POS tagging errors

Table 7: Relations missed by ZORE.

rect relations, 56% of which were caused by parsing errors, and 34% of which were due to word segmentation and POS tagging errors. Although many errors have been filtered by ZORE, the biggest source of errors is still syntactic analysis, which is very important for high quality of ORE.

5 Related Work

English has been the major language on which ORE research has been conducted. Previous work on English ORE has evolved from shallow-syntactic (Banko et al., 2007; Fader et al., 2011; Merhav et al., 2012) to full-syntactic (Nakashole et al., 2012; Mausam et al., 2012; Moro and Navigli, 2013; Xu et al., 2013) and semantic (Johansson and Nugues, 2008) systems.

It has been shown that a full-syntactic system based on dependency grammar can give significantly better results than shallow syntactic systems based on surface POS-patterns, yet enjoy higher efficiency compared with semantic systems (Mesquita et al., 2013). Our investigation on Chinese ORE takes root in full dependency syntax and is hence able to identify patterns that involve long-range dependencies. Considering the characteristics of the Chinese language, such as the lack of morphology and function words, and the high segmentation and word sense ambiguities, we incorporate semantic ontology information into the design of the system to improve the output quality without sacrificing efficiency.

The state-of-the-art systems most closely related to our approach are PATTY (Nakashole et al., 2012) and the system of Moro and Navigli (2013). Both, however, extract relations first, and then defines patterns based on extracted relations. This paper differs in that patterns and relations are extracted in a simultaneous process and so they can improve each other. Previous studies show that pattern generalization benefit from relation extrac-

56%	Parsing errors
17%	Segmentation errors
17%	POS tagging errors
6%	Redundant arguments
6%	Other, including base NP extraction errors

Table 8: Incorrect extractions by ZORE.

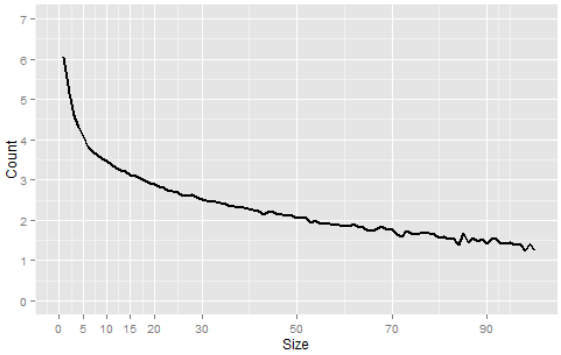

Figure 7: The frequency distribution of patterns extracted from Wiki. Size and Count denote the number of relations that belong to a semantic pattern and the logarithmic number of semantic patterns that have the same size, respectively.

tion (Nakashole et al., 2012; Moro and Navigli, 2013), and relation extraction can benefit from pattern generalization (Mausam et al., 2012). By using double propagation, not only can we make relation and pattern extraction benefit from each other, but we can also tag relations and patterns with semantic categories in a joint process.

There has been a line of research on Chinese relation extraction, where both feature-based (Zhou et al., 2005; Li et al., 2008) and kernel-based (Zhang et al., 2006; Che et al., 2005) methods have been applied. In addition, semantic ontologies such as *Extended Cilin* have been shown useful for Chinese relation extraction (Liu et al., 2013). However, these studies have focused on traditional IE, with pre-defined relations. In contrast, we investigate ORE for Chinese, finding that semantic ontologies useful for this task also. Tseng et al. (2014) is the only previous research focusing on Chinese ORE. Their system can be considered as a pipeline of word segmentation, POS-tagging and parsing, while our work gives semantic interpretation and explicitly deals with statistical errors in parsing by a novel double propagation algorithm between patterns and relations.

6 Conclusion and Future Work

We presented a Chinese ORE system that integrates relation extraction with semantic pattern generalization by double propagation. Experimental results on two datasets demonstrated the effectiveness of the proposed algorithm. We make the ZORE system, together with the large scale relations and pattern synsets extracted by ZORE, freely available at (`https://sourceforge.net/projects/zore/`). Another version of ZORE (ZORE-PMT), which is based on the dependency tagset from PMT1.0 (Qiu et al., 2014), is also provided.

Our error analysis demonstrates that the quality of syntactic parsing is crucial to the accuracy of syntax-based Chinese ORE. Improvements to syntactic analysis is likely to lead to improved ORE. In addition, the idea of double propagation can be generalized into information propagation between relation extraction and syntactic analysis. We plan to investigate the use of ORE in improving syntactic analysis in future work.

Acknowledgments

We gratefully acknowledge the invaluable assistance of Ji Ma, Wanxiang Che and Yijia Liu. We also thank the anonymous reviewers for their constructive comments, and gratefully acknowledge the support of the Singapore Ministry of Education (MOE) AcRF Tier 2 grant T2MOE201301, the start-up grant SRG ISTD 2012 038 from Singapore University of Technology and Design, the National Natural Science Foundation of China (No. 61103089), National High Technology Research and Development Program of China (863 Program) (No. 2012AA011101), Major National Social Science Fund of China (No. 12&ZD227), Scientific Research Foundation of Shandong Province Outstanding Young Scientist Award (No. BS2013DX020) and Humanities and Social Science Projects of Ludong University (No. WY2013003).

References

Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction for the web. In *IJCAI*, volume 7, pages 2670–2676.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 16–23. Association for Computational Linguistics.

Miriam Butt. 2003. The light verb jungle. In *Workshop on Multi-Verb Constructions*, pages 1–28.

Pi-Chuan Chang, Michel Galley, and Christopher D Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 224–232. Association for Computational Linguistics.

Pi-Chuan Chang, Huihsin Tseng, Dan Jurafsky, and Christopher D Manning. 2009. Discriminative reordering with Chinese grammatical relations features. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation*, pages 51–59. Association for Computational Linguistics.

WX Che, Jianmin Jiang, Zhong Su, Yue Pan, and Ting Liu. 2005. Improved-edit-distance kernel for Chinese relation extraction. In *IJCNLP*, pages 132–137.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. Ltp: A Chinese language technology platform. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, pages 13–16. Association for Computational Linguistics.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam Mausam. 2011. Open information extraction: The second generation. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume One*, pages 3–10. AAAI Press.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Richard Johansson and Pierre Nugues. 2008. Dependency-based semantic role labeling of propbank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 69–78. Association for Computational Linguistics.

Jun-Tae Kim and Dan I Moldovan. 1993. Acquisition of semantic patterns for information extraction from corpora. In *Artificial Intelligence for Applications, 1993. Proceedings., Ninth Conference on*, pages 171–176. IEEE.

Wenjie Li, Peng Zhang, Furu Wei, Yuexian Hou, and Qin Lu. 2008. A novel feature-based approach to Chinese entity relation extraction. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 89–92. Association for Computational Linguistics.

Dandan Liu, Zhiwei Zhao, Yanan Hu, and Longhua Qian. 2013. Incorporating lexical semantic similarity to tree kernel-based Chinese relation extraction. In *Chinese Lexical Semantics*, pages 11–21. Springer.

Michael Schmitz Mausam, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. pages 523–534.

Yuval Merhav, Filipe Mesquita, Denilson Barbosa, Wai Gen Yee, and Ophir Frieder. 2012. Extracting information networks from the blogosphere. *ACM Transactions on the Web (TWEB)*, 6(3):11.

Filipe Mesquita, Jordan Schmidek, and Denilson Barbosa. 2013. Effectiveness and efficiency of open relation extraction. *New York Times*, 500:150.

Andrea Moro and Roberto Navigli. 2012. Wisenet: Building a wikipedia-based semantic network with ontologized relations. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1672–1676. ACM.

Andrea Moro and Roberto Navigli. 2013. Integrating syntactic and semantic analysis into the open information extraction paradigm. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2148–2154. AAAI Press.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: a taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145. Association for Computational Linguistics.

Ron Papka and James Allan. 1998. On-line new event detection using single pass clustering. *University of Massachusetts, Amherst*.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2009. Expanding domain sentiment lexicon through double propagation. In *IJCAI*, volume 9, pages 1199–1204.

Likun Qiu, Yunfang Wu, and Yanqiu Shao. 2011. Combining contextual and structural information for supersense tagging of Chinese unknown words. In *Computational Linguistics and Intelligent Text Processing*, pages 15–28. Springer.

Likun Qiu, Yue Zhang, Peng Jin, and Houfeng Wang. 2014. Multi-view Chinese treebanking. In *Proceedings of COLING 2014*, pages 257–268.

Alan Ritter, Oren Etzioni, et al. 2010. A latent dirichlet allocation method for selectional preferences. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 424–434. Association for Computational Linguistics.

Yuen-Hsien Tseng, Lung-Hao Lee, Shu-Yen Lin, Bo-Shun Liao, Mei-Jun Liu, Hsin-Hsi Chen, Oren Etzioni, and Anthony Fader. 2014. Chinese open relation extraction for knowledge acquisition. In *EACL 2014*, pages 12–16.

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.

Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. 2013. Open information extraction with tree kernels. In *Proceedings of NAACL-HLT*, pages 868–877.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The penn Chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering*, 11(2):207–238.

Roman Yangarber, Ralph Grishman, Pasi Tapanainen, and Silja Huttunen. 2000. Automatic acquisition of domain knowledge for information extraction. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 940–946. Association for Computational Linguistics.

Yue Zhang and Stephen Clark. 2011. Syntactic processing using the generalized perceptron and beam search. *Computational Linguistics*, 37(1):105–151.

Min Zhang, Jie Zhang, Jian Su, and Guodong Zhou. 2006. A composite kernel to extract relations between entities with both flat and structured features. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 825–832. Association for Computational Linguistics.

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