Open Relation Extraction for Chinese Noun Phrases

Chengyu Wang, Xiaofeng He, *Member, IEEE*, and Aoying Zhou, *Member, IEEE*

Abstract—Relation Extraction (RE) aims at harvesting relational facts from texts. A majority of existing research targets at knowledge acquisition from sentences, where subject-verb-object structures are usually treated as the signals of existence of relations. In contrast, relational facts expressed within noun phrases are highly implicit. Previous works mostly relies on human-compiled assertions and textual patterns in English to address noun phrase-based RE. For Chinese, the corresponding task is non-trivial because Chinese is a highly analytic language with flexible expressions. Additionally, noun phrases tend to be incomplete in grammatical structures, where clear mentions of predicates are often missing. In this article, we present an unsupervised Noun Phrase-based Open RE system for the Chinese language (NPORE), which employs a three-layer data-driven architecture. The system contains three components, i.e., Modifier-sensitive Phrase Segmenter, Candidate Relation Generator and Missing Relation Predicate Detector. It integrates with a graph clique mining algorithm to chunk Chinese noun phrases, considering how relations are expressed. We further propose a probabilistic method with knowledge priors and a hypergraph-based random walk process to detect missing relation predicates. Experiments over Chinese Wikipedia show NPORE outperforms state-of-the-art, capable of extracting 55.2 percent more relations than the most competitive baseline, with a comparable precision at 95.4 percent.

Index Terms—Open relation extraction, noun phrase segmentation, graph clique mining, hypergraph-based random walk

1 Introduction

1.1 Motivation

Relation Extraction (RE) is one of the core NLP tasks which aims at harvesting relational facts from free texts automatically. The extracted relations are essential for various applications such as knowledge base construction [1], [2], taxonomy learning [3], question answering [4], etc.

According to different task settings, RE can be addressed using a variety of machine learning paradigms, including supervised relation classification [5], distantly supervised RE over knowledge bases [6], pattern-based iterative relation bootstrapping [7] and the Open Information Extraction (OIE) approaches which do not require the input of a collection of pre-defined relation types [8], [9]. These methods mainly deal with RE on the *sentence* level, which determine the semantic relation between two entities within a single sentence. Recently, several approaches consider the global contexts of entities and model the global consistency of distant supervision, in order to extract relations across sentence boundaries on the *corpus* level [10], [11], [12].

A drawback of these approaches is that they pay little attention to semantic relations expressed by smaller semantic units. For example, we can extract the relation "(Donald Trump, is-decent-of, Scottish)" from "Scottish American" describing Donald Trump. These noun phrases contain rich knowledge and are regarded as fine-grained representations of entities [13], [14]. However, harvesting such knowledge from noun phrases is non-trivial because they are extremely incomplete of syntactic structures. Relational expressions within noun phrases are highly implicit [15]. While the relation predicate "is-decent-of" in the previous case can be easily inferred by humans, this predicate is omitted in texts and is difficult for machines to generate. To deal with this problem, noun phrase-based OIE systems are proposed to extract relations from noun compounds [16], [17], [18]. These systems require a large number of human-compiled assertions and lexical patterns to identify relations. For instance, pattern "the [...] of [...]" can be used to extract "(Donald Trump, is-president-of, United States)" from "the President of the United States, Donald Trump".

Although there has been significant success for English, harvesting such relations from Chinese noun phrases still faces several challenges and is an emerging task for NLP. This is because Chinese is a highly *analytic* language, lacking explicit expressions to convey grammatical relations [19]. There are no word spaces, explicit tenses and voices, or singular/plural distinctions in Chinese. Circumstances of how semantic relations are expressed in Chinese noun phrases are more complicated. Additionally, based on linguistic research, properties of entities are more likely to be expressed by noun phrases rather than verbal clauses [20]. To achieve more intuitive understanding, we illustrate relations extracted from two noun phrases describing Donald Trump:

Example 1. (American entrepreneur born in 1946)[1]

- *C. Wang is with the School of Software Engineering, East China Normal University, Shanghai 200062, China. E-mail: chywang2013@gmail.com.*
- *X. He is with the School of Computer Science and Technology, East China Normal University, Shanghai 200062, China. E-mail: hexf@cs.ecnu.edu.cn.*
- *A. Zhou is with the School of Data Science and Engineering, East China Normal University, Shanghai 200062, China. E-mail: ayzhou@dase.ecnu.edu.cn.*

Manuscript received 28 Dec. 2018; revised 31 Oct. 2019; accepted 11 Nov. 2019. Date of publication 18 Nov. 2019; date of current version 29 Apr. 2021. (Corresponding author: Xiaofeng He.)
Recommended for acceptance by J. Tang.
Digital Object Identifier no. 10.1109/TKDE.2019.2953839

[1] The Chinese noun phrases below are printed after the Chinese word segmentation process [21]. English words directly under Chinese characters refer to the literal translation. "的(*de*)" is a Chinese auxiliary word, usually put at the end of a modifier.

1946年	出生	的	美国	企业家
Year of 1946	*Born*	*de*	*America*	*Entrepreneur*
	(Modifier 1)		*(Modifier 2)*	*(Head)*

Extracted relations (English translation):
(Donald Trump, born-in, 1946)
(Donald Trump, has-nationality, American)

Example 2. (People originated from Queens, New York)

纽约市	皇后区	出身	人物
New York City	*Queens District*	*Origin*	*Person*
	(Modifier 1)		*(Head)*

Extracted relation (English translation):
(Donald Trump, originated-from, Queens District of New York)

As seen, such Chinese noun phrases are usually in the form of *one/many modifier(s) + head word*, with prepositions omitted in a large proportion. While *head words* are typically considered as *hypernyms* or *topics* of entities [22], [23], *modifiers* express non-taxonomic semantic relations of entities, either explicitly or implicitly. Different from traditional RE approaches, we observe that three unique challenges should be addressed for accurate RE from Chinese noun phrases:

Challenge 1. (Difficult to segment Chinese noun phrases into modifiers and head words) For English, boundaries between modifiers and head words can be identified by patterns [24], [25]. In contrast, there are no natural boundaries in Chinese noun phrases. Chinese word segmentation and NLP chunking methods (e.g., [21], [26]) can not be applied to this task directly. A modifier (or even a complicated entity, see Example 2) may consist of multiple words. There is no standard, effective solution in NLP to solve this problem, without large amount of manual work.

Challenge 2. (Unclear mappings from modifiers to semantic relations) Due to the lack of prepositions and attributive clauses in Chinese, a modifier is usually a combination of nouns and other words. It is unclear how to extract the relation predicate and the object from the modifier to generate relation triples.

Challenge 3. (Missing relation predicates in noun phrases) In many cases, relation predicates are non-existent in modifiers. In Example 1, the noun phrase does not explicitly express the relation predicate between America and Donald Trump. Humans can easily infer the relation predicate as “has-nationality” based on commonsense knowledge. In contrast, a specific mechanism should be designed for machines to learn the predicates automatically. In the literature, it is similar to the task of noun phrase interpretation in NLP [27]. However, our task is more challenging due to the complicated linguistic nature of the Chinese language.

1.2 Summary of Our Approach

We present an unsupervised Noun Phrase-based Open RE system for the Chinese language (NPORE). The input is a collection of entity-noun phrase pairs, where noun phrases are semantically related to the corresponding entities. The system generates knowledge in the form of relation triples, describing facts about entities explicitly. A topically related corpus is also provided, treated as the background knowledge source. To avoid tedious human labeling, the NPORE system employs a three-layer data-driven architecture. The three major components are summarized as follows:[2]

Step 1. Modifier-sensitive Phrase Segmenter (MPS): It segments a Chinese noun phrase into one/many modifier(s) and one head word. To avoid the time-consuming human labeling process and to be self-adaptive to any domains, we propose an unsupervised graph clique mining algorithm to segment the noun phrases based on statistical measures and word embeddings. Especially, we propose two graph pruning strategies and an approximate algorithm for efficient graph clique detection.

For example, after the process of MPS, the segmentation results of the two noun phrases are shown as follows:

Example 1	1946年 出生 的	美国	企业家
	Born in 1946	*America*	*Entrepreneur*
	(Modifier 1)	*(Modifier 2)*	*(Head)*

Example 2	纽约市 皇后区 出身	人物
	Originated from Queens, New York	*Person*
	(Modifier 1)	*(Head)*

Step 2. Candidate Relation Generator (CRG): This component generates full relations (subject-predicate-object triples) and partial relations (subject-object pairs with predicates missing) based on the results of MPS and syntactic structures of noun phrases.

The sample outputs of CRG are shown in below:

Example 1	*(Donald Trump, born-in, 1946)* *(Donald Trump, ?, American)*	*[full relation]* *[partial relation]*
Example 2	*(Donald Trump, originated-from, Queens, New York)*	*[full relation]*

Step 3. Missing Relation Predicate Detector (MRPD): For partial relations, a probabilistic predicate detection approach is proposed. Especially, we employ Bayesian knowledge priors and a hypergraph-based random walk process to encode both contextual signals derived from the background text corpus and the commonsense knowledge of humans into the model.

In the experiments, we evaluate the NPORE system over datasets generated from Chinese Wikipedia categories. Generally, the number of extracted relation triples are 155.2 percent as many as the most competitive baseline and has a comparable precision of 95.4 percent. We also evaluate various aspects of the system to make the convincing conclusion.

1.3 Contributions and Paper Organization

In summary, we make the following major contributions:

- We introduce an unsupervised RE system named Noun Phrase based Open RE. It employs a three-layer

[2] Head words of noun phrases may express *is-a* or *topic-of* relations between entities and the noun phrases. This issue has been addressed via the hypernymy predication task in abundant papers (e.g., [22], [23]) and summarized in [28]. Hence, it is not the focus of this work.

data-driven architecture to extract various relations from Chinese noun phrases.

- We propose a graph-based algorithm to chunk Chinese noun phrases into modifiers and head words. A probabilistic method (integrated with Bayesian priors and a hypergraph-based random walk process) is presented to detect missing relation predicates.
- We conduct extensive experiments over Chinese Wikipedia categories to evaluate NPORE. The results show that it outperforms state-of-the-art approaches.

The rest of this paper is organized as follows. Section 2 summarizes the related work. The detailed techniques of NPORE are described in Section 3. Experiments are presented in Section 4, with the conclusion drawn in Section 5.

2 Related Work

In this section, we briefly overview recent advances on RE. Besides the research on the general RE task, we specifically focus on noun phrase-based RE and commonsense RE. This is because our goal is to extract relations from Chinese noun phrases, which also requires commonsense reasoning to detect missing predicates (especially for spatial and temporal commonsense relations). Additionally, we discuss some special considerations for RE over Chinese texts.

2.1 General Relation Extraction

The task of RE has been extensively studied in the NLP community, aiming at harvesting relational facts from free texts automatically. A typical paradigm of RE is supervised relation classification, which classifies entity pairs into a finite, pre-defined set of relation types based on contextual information [5]. To reduce human labeling efforts, distant supervision has been proposed to use relational facts in knowledge bases as training data [6]. One disadvantage of these approaches is that relation types of the RE systems need to be defined by humans in advance. OIE expands the RE research into open domains, which automatically identifies relation types and their corresponding relation triples in sentences [8], [29], [30]. Recently, deep learning benefits RE at a large extent by introducing techniques mostly in the following aspects: i) deep reinforcement learning models optimize long-term rewards of the quality of extracted relations by treating relation extractors as agents [31]; adversarial training techniques impose additional regularization effects on relation classification by training discriminators and relation extractors at the same time [32]; attention mechanisms [33] improve the process of contextual feature extraction from sentences, etc. For OIE, the encoder-decoder architecture has been deeply exploited [34]. Because the general RE task is not the major focus of this work, we do not elaborate here.

To improve the recall of RE, several works harvest relations beyond the single sentence level. The intuition is that relations can be identified by considering more complicated sentence structures and the global contexts of entities, rather than a single sentence [10], [11], [12]. For example, Han and Sun [10] propose a global distant supervision model, which reduces the uncertainty of traditional distant supervision approaches by considering the global consistency of RE. Su et al. [11] learn the textual relation embeddings for distantly supervised RE. This work deals with the wrong labeling problem of distant supervision and models the global statistics of relations. For OIE systems, Zhu et al. [35] leverage global information in documents by adding global structure constraints to the relation extractors of OIE. These methods harvest general semantic relations effectively but are unable to deal with relations hidden in non-sentences, especially for relations expressed by noun phrases.

2.2 Noun Phrase-Based Relation Extraction

A recent advance on OIE is to learn noun phrase-based relations. For example, Xavier and de Lima [16] harvest relations expressed in noun compounds based on noun phrase interpretation. RELNOUN [18] extends the RENOUN system [17], which considers demonyms and relational compound nouns to improve noun-based OIE. Among all noun phrases, *user generated categories* (especially Wikipedia categories) are highly informative, providing rich knowledge to characterize entities. To discover relations in Wikipedia categories, Nastase and Strube [36] propose a hybrid approach based on preposition patterns. Pasca [37] studies how to decompose names of Wikipedia categories into attribute-value pairs, using lexical patterns in English.

Another similar task to extract relations from noun phrases is called *noun phrase interpretation*, which uses verbal relations to interpret the meanings of noun phrases. This task is closely related to MRPD in our system for finding the missing predicates. For example, a verbal relation “made-from” should be generated from the noun phrase “olive oil”, because “olive oil” is a kind of “oils” *made from* “olives”. In the literature, this is usually formulated as supervised classification, where a noun phrase is classified into an abstract verbal relation from a manually-defined, fixed inventory [38]. However, a finite set of relations (or verbs) are insufficient to represent the semantics of noun phrases. For finer-grained relation representations, several works (e.g., [39], [40]) consider multiple paraphrases to express the semantics of noun phrases. In SemEval-2013 Task 4 [41], participants are allowed to use free paraphrases to represent the relations within noun phrases.

Compared to English, RE from Chinese noun phrases does not yield comparable performance due to the complicated linguistic nature. ZORE [42] is a recent sentence-based OIE system for Chinese, using verb-based syntactic patterns to extract relations. Similar OIE systems include [43], [44], etc. However, there is limited success in RE from Chinese short texts. Our previous work [14], [45] proposes to learn multiple types of relations over Chinese Wikipedia categories, which relies on human work to define relation types and only deals with most frequent patterns. This work improves previous research by enabling unsupervised open-domain commonsense RE from Chinese noun phrases.

2.3 Commonsense Relation Extraction

Commonsense RE is fundamentally different from general RE, because commonsense relations are rarely expressed in texts. In the early age of artificial intelligence, commonsense knowledge bases are mostly constructed by experts or Web-scale crowd-sourcing, such as the CYC project [46], ConceptNet [47], etc. The automatic acquisition of commonsense

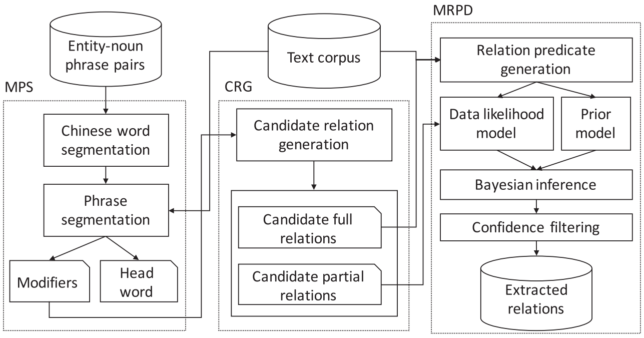


Fig. 1. The general framework of the NPORE system.

relations is primarily based on pattern-based approaches. In the literature, WebChild [15] employs textual patterns to extract several types of commonsense relations from Web texts, including has-shape, has-taste, evokes-emotion, etc. The research of Narisawa et al. [48] focuses on harvesting numerical commonsense facts. They extract numerical expressions and their contexts from the Web, and propose distributional and pattern-based models to predict whether a given value is large, small, or normal based on the context. The extraction of spatial commonsense relations is presented in [49], based on implicit spatial templates. Xu et al. [50] specifically focus on the locate-near commonsense relations. They propose a sentence-level relation classifier to predict whether two entities are close to each other, and aggregate the scores of entity pairs from a large corpus.

As seen, all above works on short-text RE pay attention to one or a few types of relations only. It remains a challenge to extract a large number of relations from Chinese noun phrases, without pre-defined relation types. Our work aims at solving this problem, with minimal human supervision.

3 The NPORE System

In this section, we begin with a high-level overview of the NPORE system, with important notations and concepts introduced. Next, we elaborate the algorithms and techniques of the three components of NPORE in detail.

3.1 Overview of NPORE Components

The input of NPORE is a collection of entity-noun phrase pairs, denoted as $\{(e,p)\}$ where the noun phrase p describes the entity e. For example, we have $e=$“唐纳德・特朗普 (Donald Trump)” and $p=$“1946年出生的美国企业家(American entrepreneur born in 1946)”. A topically related text corpus D is also provided as the background knowledge source. The three modules of NPORE are introduced as follows. The general framework of the NPORE system is illustrated in Fig. 1.

3.1.1 Modifier-Sensitive Phrase Segmenter (MPS)

We first perform Chinese word segmentation over the noun phrase p. The result is denoted as an ordered list: $ws(p)=\{w_1,w_2,\ldots,w_{|ws(p)|}\}$ where $w_i\in ws(p)$ is a segmented Chinese word in p. The goal of MPS is to generate the modifier-sensitive phrase segmentation of p, i.e., $ps(p)=\{q_1,q_2,\ldots,q_{|ps(p)|}\}$ where q_i is a modifier/head word in p, consisting of one or several words in $ws(p)$.

Based on the observation discussed in the introduction and previous research [51], we follow the assumption that a noun phrase consists of one/many modifiers and a head word. We treat $q_{|ps(p)|}$ as the head word of p and q_i $(1\leq i\leq|ps(p)|-1)$ as a modifier of p. In order to generate the segmentation results, we construct an N-gram Segmentation Graph (NSG) G_p for each noun phrase p and generate the result $ps(p)$ based on the structure of G_p.

3.1.2 Candidate Relation Generator (CRG)

After MPS, we generate entity-modifier pairs $\{(e,q_i)\}$ where each $q_i\in ws(p)$ $(1\leq i\leq|ps(p)|-1)$. Denote $R(p)$ as the collection of extracted candidate relations from the pair (e,p). For each entity-modifier pair (e,q_i), if a candidate predicate can be detected from q_i or the entire sequence $ws(p)$, a *candidate full relation* $r(e,q_i)$ is extracted and added to $R(p)$. Let $r_v(e,q_i)$ and $r_o(e,q_i)$ be the relation predicate and object w.r.t. the relation $r(e,q_i)$, respectively. If the relation predicate is missing or can not be detected, a *candidate partial relation* $\tilde{r}(e,q_i)$ is derived and added to $R(p)$. The relation object is denoted as $\tilde{r}_o(e,q_i)$.

3.1.3 Missing Relation Predicate Detector (MRPD)

In this step, we employ a probabilistic predicate detection algorithm with knowledge priors and a hypergraph-based random walk process to detect the proper relation predicates. A collection of relation predicates $\mathcal{V}$ is first generated. After that, the prior model $\Pr(v)$ and the data likelihood model $\Pr(\tilde{r}(e,q_i)|v)$ are trained in an unsupervised manner, where $v\in\mathcal{V}$. Especially, we construct a Predicate-based Hypergraph Network (PHN) $H(\mathcal{R},\mathcal{V})$ to approximate $\Pr(\tilde{r}(e,q_i)|v)$ based on a random walk process. The most possible relation predicate $\tilde{r}_{v^*}(e,q_i)$ w.r.t. the partial relation $\tilde{r}(e,q_i)$ is generated via Bayesian inference over $\Pr(v)$ and $\Pr(\tilde{r}(e,q_i)|v)$ for all $v\in\mathcal{V}$.

Finally, for both full relations and partial relations with the predicate detected, we compute confidence scores $conf(r(e,q_i))$ or $conf(\tilde{r}(e,q_i))$ to quantify the possibility that the relation triples are correct. Relation triples with low confidence scores are filtered.

Important notations are summarized in Table 1.

3.2 Modifier-Sensitive Phrase Segmenter

In this section, we introduce the graph mining-based approach for MPS. To ensure that our system is unsupervised and can be adapted to any domains, this algorithm is fully data-driven and does not require any human-labeled data.

3.2.1 Graph Construction

The first step of MPS is Chinese word segmentation, separating a noun phrase p into a sequence of words, i.e., $ws(p)=\{w_1,w_2,\ldots,w_{|ws(p)|}\}$. In this work, we treat Chinese word segmentation and modifier-sensitive phrase segmentation as two separate tasks for two reasons: i) Chinese word segmentation techniques have relatively high performance [21]; ii) the separation of two tasks lowers the computational complexity of MPS.

A natural gap between Chinese word segmentation and MPS is that the boundaries of modifiers and head words in Chinese are semantically implicit. Consider the following noun phrase in both English and Chinese:

TABLE 1
Important Notations

Notation	Description
(e, p)	An entity-noun phrase pair such that the noun phrase p describes the entity e
D	A large background text corpus
$ws(p)$	Chinese word segmentation result of phrase p
$ps(p)$	Modifier-sensitive segmentation result of phrase p
(e, q_i)	An entity-modifier pair where $q_i \in ps(p)$
$G_p(M, E, W)$	An N-gram Segmentation Graph (NSG) w.r.t. phrase p
$\text{vec}(m_i)$	The embedding vector of n-gram m_i
$\mathcal{P}_i / \mathcal{N}_i$	A positive/negative constraint over (w_i, w_{i+1})
M^*	The maximum edge weight clique in NSG G_p
$R(e, p)$	The collection of candidate relations generated from the pair (e, p)
$r(e, q_i)$	A full relation generated from the pair (e, q_i)
$\tilde{r}(e, q_i)$	A partial relation generated from the pair (e, q_i)
$r_v(e, q_i)$	The relation predicate of $r(e, q_i)$
$r_o(e, q_i)$	The relation object of $r(e, q_i)$
$\mathcal{V}$	A collection of relation predicates
$\Pr(v)$	The prior model of relation predicates where $v \in \mathcal{V}$
$\Pr(\tilde{r}(e, q_i) \mid v)$	The data likelihood model where $v \in \mathcal{V}$
$H(\mathcal{R}, \mathcal{V})$	The Predicate-based Hypergraph Network (PHN)
$conf(r(e, q_i))$	The confidence score of the full relation $r(e, q_i)$
$conf(\tilde{r}(e, q_i))$	The confidence score of the partial relation $\tilde{r}(e, q_i)$

Example 3. (American entrepreneur in the 21st century)

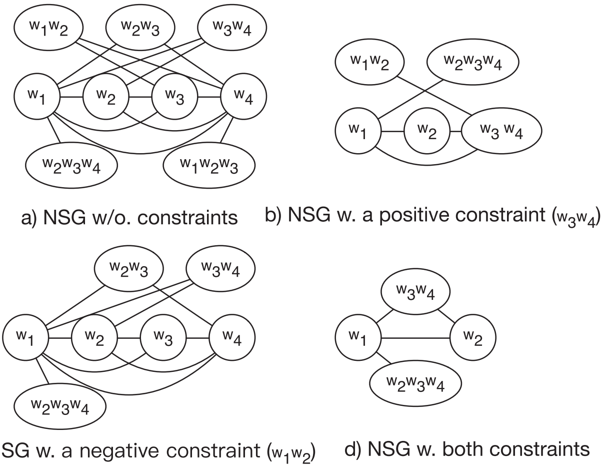
English	American	entrepreneur	in the 21st century
	Pre-modifier	Head word	Post-modifier
	(Adjective)	(Noun)	(Prepositional phrase)
Chinese	21 世纪	美国	企业家
	21st century	America	entrepreneur
	Modifier	Modifier	Head word
	(Two Nouns)	(Noun)	(Noun)

As seen, the modifiers in English noun phrases can be easily separated by POS-based rules (which is a common practice in the literature such as [24]). For Chinese, there is no clear separation between modifiers and head words. The lack of variations of lexical units results in the fact that multiple consecutive nouns can be used to modify the head words, further increasing the difficulty of MPS.

In this work, we propose a graph-based approach to address this problem. For each segmented noun phrase $ws(p)$, we construct a graph model to represent all possible configurations of phrase segmentation, and select the best configuration as the results of MPS. Define n as an n-gram factor, typically set to a small, positive integer. We introduce the concept of N-gram Segmentation Graph:

Definition 1 (N-gram Segmentation Graph). *An NSG $G_p(M, E, W)$ w.r.t. noun phrase p is an undirected graph with edge weights, where M and E denote collections of nodes and edges, respectively. W is an $|E|$-dimensional weight vector that assigns a weight $\alpha_{i,j}$ to each $(m_i, m_j) \in E$, in the range of [0,1].*

In the graph $G_p(M, E, W)$, each node $m \in M$ is a word sequence derived from $ws(p)$. The word sequences include uni-grams, bi-grams, to n-grams. In Fig. 2a, given $ws(p) = \{w_1, w_2, w_3, w_4\}$ and $n = 3$, we have $M = \{w_1, w_2, w_3, w_4, w_1w_2, w_2w_3, w_3w_4, w_1w_2w_3, w_2w_3w_4\}$. For each $m_i, m_j \in M$, we constrain that $(m_i, m_j) \in E$ iff $m_i \cap m_j = \emptyset$.[3] This is because in a segmented noun phrases, any two elements must be mutually excluded. We can see that each maximal clique in the NSG $G_p(M, E, W)$ represents a configuration of phrase segmentation of $ws(p)$. For instance, if the maximal clique $\{w_1, w_2w_3, w_4\}$ is selected, the phrase segmentation of $ws(p)$ is $m_1 = w_1, m_2 = w_2w_3, m_3 = w_4$. For rigorousness, we prove this claim as follows:



a) NSG w/o. constraints b) NSG w. a positive constraint (w_3w_4)

c) NSG w. a negative constraint (w_1w_2) d) NSG w. both constraints

Fig. 2. The graph structure of NSG w. and w/o. positive and negative constraints. For simplicity, edge weights of all graphs are omitted. (In this example, we have $ws(p) = \{w_1, w_2, w_3, w_4\}$ and $n = 3$.).

Theorem 1. *A maximal clique in G_p is equivalent to a configuration of the phrase segmentation of p.*

Proof Sketch. Based on the definition of the modifier-sensitive phrase segmentation, a valid phrase segmentation of p (i.e.,$ps(p) = \{q_1, q_2, \ldots, q_{|ps(p)|}\}$) forms a partition of $ws(p)$. This requires two conditions: i) $\forall q_i, q_j \in ps(p), q_i \cap q_j = \emptyset$ and ii) $\bigcup_{q_i \in ps(p)} = ws(p)$.

Consider a maximal clique $M^{'}$ in G_p. Because $\forall m_i, m_j \in M, m_i \cap m_j = \emptyset$ and $M^{'} \subseteq M$, we have $\forall m_i, m_j \in M^{'}$, $m_i \cap m_j = \emptyset$. By mapping each $m_i \in M^{'}$ to its corresponding segment $q_i \in ps(p)$, we have $\forall q_i, q_j \in ps(p), q_i \cap q_j = \emptyset$. Next, we prove the satisfaction of Condition ii) by contradiction. Assume the maximal clique $M^{'}$ does not ensure $\bigcup_{q_i \in ps(p)} = ws(p)$. There must be one node m_i^* not in $M^{'}$ such that $m_i^* \notin \bigcup_{m_i \in M^{'}}$. This is contradictory to the definition of the maximal clique because adding m_i^* to $M^{'}$ also forms a clique. Therefore, the assumption is not valid.

For the definition of the weights W, we propose a hybrid approach to encode both statistical and distributional knowledge into the model. If m_i and m_j are two consecutive n-grams in $ws(p)$ (e.g., $m_i = w_1$, $m_j = w_2w_3$), the statistical score $w_s(i, j)$ and the distributional score $w_d(i, j)$ are defined as follows. $w_s(i, j)$ is a variant of Normalized Pointwise Mutual Information (NPMI), in the range of [0,1]

[3] Without ambiguity, we also use the notation m_i to represent words of corresponding n-grams. Hence, $m_i \cap m_j = \emptyset$ means corresponding n-grams of m_i and m_j do not share overlapping sequences.

$$w_s(i,j) = \frac{1}{2} - \frac{\mathrm{PMI}(i;j)}{2h(i,j)} = -\frac{\log\,\Pr(m_i)\,\Pr(m_j)}{2\log\,\Pr(m_i,m_j)},$$

where $\mathrm{PMI}(i;j)$ and $h(i,j)$ are the PMI scores and self-information of n-grams m_i and m_j, respectively. $\Pr(m_i)$, $\Pr(m_j)$ and $\Pr(m_i,m_j)$ are probabilities estimated using any language models.

The distributional score $w_d(i,j)$ is inspired by the compositionality analysis in computational linguistics. We define $w_d(i,j)$ based on a variant of the measure in [52]

$$w_d(i,j) = \frac{1}{2}(1 - \cos(\mathrm{vec}(m_i m_j), \mathrm{vec}(m_i + m_j))),$$

where $\mathrm{vec}(m_i m_j)$ is the compound embedding of m_i and m_j, and $\mathrm{vec}(m_i + m_j)$ is the normalized sum of the word embeddings of m_i and m_j separately, i.e., $\mathrm{vec}(m_i + m_j) = \frac{\mathrm{vec}(m_i)}{\|\mathrm{vec}(m_i)\|} + \frac{\mathrm{vec}(m_j)}{\|\mathrm{vec}(m_j)\|}$. If m_i and m_j are highly indecomposable, the individual contexts of m_i and m_j should be significantly different from the context of the $m_i m_j$ compound. Hence, $\mathrm{vec}(m_i m_j)$ and $\mathrm{vec}(m_i + m_j)$ are dis-similar. We employ the compositionality score in this work because it leverages the low-dimensional representations of terms.

As seen, if m_i and m_j are highly decomposable, $w_s(i,j)$ and $w_d(i,j)$ will be close to 1. This is a strong signal that m_i and m_j should be groped into different modifiers/head words. $\alpha_{i,j}$ is computed by combining the two scores

$$\alpha_{i,j} = \gamma w_s(i,j) + (1-\gamma) w_d(i,j), \tag{1}$$

where $\gamma \in (0,1)$ is a pre-defined hyper-parameter.

Remarks. For simplicity, let $k = |ws(p)|$. Recall that n is the n-gram factor ($n \leq k$). It is trivial to see that at least $\lceil \frac{k}{n} \rceil$ times of segmentation are required. Hence, the total number of possible segmentation configurations Δ is derived as

$$\Delta = \sum_{i=\lceil \frac{k}{n} \rceil}^{k-1} \binom{k-1}{i} = 2^{k-1} - \sum_{i=0}^{\lceil \frac{k}{n} \rceil - 1} \binom{k-1}{i}$$

where $ki = \frac{k!}{i!(k-i)!}$. Therefore, in the worst cases, it takes $\mathcal{O}(2^k)$ time to find the best segmentation by brute-force search of all maximal cliques.

Although in real applications, n and k are small integers, finding the optimal segmentation result could be computationally expensive. In this work, we propose two techniques to minimize the computation cost: i) two graph pruning strategies to reduce the graph size; and ii) an approximate algorithm to detect the proper maximal clique (i.e., the segmentation result).

3.2.2 Graph Pruning Strategies

We introduce two types of constraints based on linguistic rules to reduce of the NSG size and improve the accuracy of MPS. The concept of positive constraint is defined as:

Definition 2 (Positive Constraint). *A positive constraint $\mathcal{P}_i$ is defined over a consecutive word pair (w_i, w_{i+1}) ($w_i \in ws(p)$, $w_{i+1} \in ws(p)$) such that two words w_i and w_{i+1} must be segmented into the same phrase.*

Fig. 2b illustrates the NSG structure by adding a positive constraint over (w_3, w_4). The other type of constraints is the negative constraint, defined as follows:

Definition 3 (Negative Constraint). *A negative constraint $\mathcal{N}_i$ is defined over a consecutive word pair (w_i, w_{i+1}) ($w_i \in ws(p)$, $w_{i+1} \in ws(p)$) such that two words w_i and w_{i+1} must not be segmented into the same phrase.*

The NSG structure with a negative constraint over (w_1, w_2) is shown in Fig. 2c. In this example, compared with the original graph, all nodes containing the bi-gram $w_1 w_2$ are removed. The combination of both constraints is performed by calculating the intersection of the two graphs, as illustrated in Fig. 2d.

Remarks. We analyze to what degree the two types of constraints can reduce the graph size. For the original graph, it is trivial to see that: $|M| = k + (k-1) + \cdots + (k-n+1) = nk - \frac{1}{2}n^2$.

Define Φ as the collection of all words associated with at least one positive constraint. For example, we have $\Phi = \{w_1, w_2, w_3, w_5, w_6\}$ if (w_1, w_2), (w_2, w_3) and (w_5, w_6) match the constraints. The usage of positive constraints reduce the number of nodes to $|M| - |\Phi|$ by removing the corresponding $|\Phi|$ uni-grams. The situations of negative constraints are more complicated, depending on the positions of words and the values of n and k. For a negative constraint $\mathcal{N}_i = (w_i, w_{i+1})$, if $i = 1$ or $i+1 = k$, it can reduce $n-1$ nodes in the graph, which is the worst case. The best case can be satisfied if $i+1 \geq n$ and $i > k-n$, with the reduction number of nodes as $\sum_{j=1}^{n-1} j = \frac{1}{2}n(n-1)$. Let ψ be the number of negative constraints used in this work. The number of nodes in the NSG is loosely bounded by: $[nk - \frac{1}{2}n^2 - |\Phi| - \frac{\psi}{2}(n-1), nk - \frac{1}{2}n^2 - |\Phi| - \frac{\psi}{2}n(n-1)]$. Tighter bounds can be achieved by discussing all cases in details, which are beyond the scope of this paper. As seen in Fig. 2, the number of nodes of the NSG reduces from 9 to 4 by applying only two constraints. Correspondingly, the number of edges reduces from 14 to 4.

Algorithm 1. NSG Construction with Pruning Strategies

```
Input: Word segmentation result $ws(p)$ of phrase $p$, n-gram factor $n$, positive constraints $\{\mathcal{P}_i\}$, negative constraints $\{\mathcal{N}_i\}$.
Output: Pruned NSG $G_p(M, E, W)$.
Initialize an empty NSG $G_p(M, E, W)$;
//Handling positive constraints
Construct the word collection $\Phi$ w.r.t. positive constraints $\{\mathcal{P}_i\}$;
for each $w_i \in ws(p)$ ($i < |ws(p)|$) do
  if $w_i \notin \Phi$ then
    Add the node $w_i$ to $M$;
  end if
end for
//Handling negative constraints
for $j = 2$ to $n$ do
  for each $w_i \in ws(p)$ ($i < |ws(p)| - j + 1$) do
    if word pairs in $w_i w_{i+1} \cdots w_{i+j-1}$ does not violate any negative constraints $\{\mathcal{N}_i\}$ then
      Add the node $w_i w_{i+1} \cdots w_{i+j-1}$ to $M$;
      Add corresponding edges w.r.t. $w_i w_{i+1} \cdots w_{i+j-1}$ to $E$;
      Compute the weights of the edges by Eq. (1);
    end if
  end for
end for
return Pruned NSG $G_p(M, E, W)$.
```

TABLE 2
Positive and Negative Constraints That we Designed for the Chinese Language

Positive Constraints w.r.t. w_i and w_{i+1}
Condition 1: POS(w_i)=VERB and POS(w_{i+1})=PREP Condition 2: POS(w_i)=CONJ or POS(w_{i+1})=CONJ Condition 3: w_{i+1}="的(*de*)"
Negative Constraint w.r.t. w_i and w_{i+1}
Condition 1: w_i="的(*de*)"

POS(w_i) is the Part-of-Speech tag of word w_i.

Algorithm 1 presents the detailed procedure to construct the NSG, with pruning strategies applied. Before the algorithm adds all uni-grams $w_i \in ws(p)$ to the NSG G_p, the word collection set Φ is constructed by checking all the positive constraints. If $w_i \in \Phi$, w_i does not need to be added to the graph. The negative constraints take effect when bi-grams, tri-grams to n-grams in $ws(p)$ are added as nodes. If any pair of consecutive words in $w_i w_{i+1} \cdots w_{i+j-1}$ violate one negative constraint ($j > 1$), the node $w_i w_{i+1} \cdots w_{i+j-1}$ should be pruned in advance. As seen, by applying the two pruning strategies during the graph construction process, the system does not need to construct the original NSG fully, which reduces computational resources.

Because our work specifically focuses on the Chinese language, we design three positive constraints and one negative constraint specifically for the Chinese language, shown in Table 2. For example, the auxiliary word "的(*de*)" is an important signal, indicating the end of a modifier. Hence, the noun phrase should be segmented after the word "的(*de*)". Our method is flexible that can be extended to other languages by designing language-specific rules.

3.2.3 Approximate Algorithm for MEWC

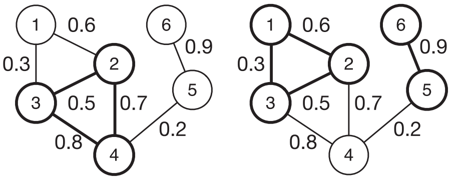
We select the optimal maximal clique as the best segmentation result over the pruned NSG $G_p = (M, E, W)$. In this work, this problem is modeled as detecting the Maximum Edge Weight Clique (MEWC) M' among all maximal cliques in G ($M' \subseteq M$). Denote $G_p'(M', E')$ as the subgraph of G_p w.r.t. the clique M'. Formally, MEWC is defined as:

Definition 4 (MEWC Problem). *The optimization objective of the Maximum Edge Weight Clique problem is:*

$$\max \sum_{(m_i,m_j)\in E'} \alpha_{i,j}$$

$$\text{s.t. } E' \subseteq E, \forall m_i \in M', \forall m_j \in M', (m_i, m_j) \in E'.$$

This problem proves to be NP-hard by Alidaee et al. [53]. In our previous work, we present an Monte Carlo-based approximate algorithm to solve this problem, suitable for detecting MEWCs in word similarity graphs [14], [45]. This algorithm is highly efficient for dense, complete graphs. However, NSGs, especially after constraints-based pruning, tend to be sparse in structure. Directly applying this method to these graphs may lead to detection of multiple cliques in a graph, rather than one clique with the largest sum of edge weights. This is because the algorithm greedily selects edges with large weights without checking whether the selected edges can form a single clique. An example of MEWC and its counter-example are shown in Fig. 3.



a) Detected MEWC b) A counter-example

Fig. 3. The detected MEWC and an counter-example. The cliques and their corresponding edges are printed in bold.

In this work, we improve the algorithm of MEWC for sparse graphs, as shown in Algorithm 2. It starts with the selection of an edge $(m_i, m_j) \in E$ with probability $\propto \alpha_{i,j}$. Let $G_p' = (M', E')$ be an initial graph where $M' = \{m_i, m_j\}$ and $E' = \{(m_i, m_j)\}$. $N(M')$ is the neighboring node set of M' in G_p. For each $m_i \in N(M')$, the algorithm checks whether $M' \cup \{m_i\}$ forms a clique and denotes nodes that satisfy the criteria as the candidate node set $Can(M')$. An iterative process samples m_i from $Can(M')$ with probability $\propto \sum_{m_j \in M'} \alpha_{i,j}$, and add m_i and its corresponding edges to G_p'. The node collections $N(M')$ and $Can(M')$ are updated in each iteration. The algorithm continues until no edges can be selected, resulting in M' as the MEWC of the NSG G_p.

Algorithm 2. Improved Algorithm for MEWC

```
Input: Pruned NSG $G_p = (M, E, W)$.
Output: MEWC $V'$.
Sample an edge $(m_i, m_j)$ from $E$ with prob. $\propto \alpha_{i,j}$;
Initialize $G' = (M', E')$ with $M' = \{m_i, m_j\}, E' = \{(m_i, m_j)\}$;
Compute neighbor node set $N(M')$;
Generate candidate node set $Can(M') \subseteq N(M')$;
while $Can(M') \neq \emptyset$ do
    Sample $m_i$ from $Can(M')$ with prob. $\propto \sum_{m_j \in M'} \alpha_{i,j}$;
    Add $m_i$ and corresponding edges to $G'$;
    Update $N(M')$ and $Can(M')$;
end while
return MEWC $M'$.
```

The worst-case runtime complexity of this algorithm is $\mathcal{O}(|M|^2|E|)$ using hash-maps as graph implementation, slightly larger than [14] (i.e., $\mathcal{O}(|E|^2)$). However, the increase of complexity does not affect efficiency much because in most cases, a pruned NSG usually contains fewer than ten nodes. We run this algorithm multiple times and denote the collection of detected cliques as $\mathcal{C} = \{M'\}$. The clique $M^* \in \mathcal{C}$ is selected to form the final segmentation result

$$M^* = \operatorname*{argmax}_{M' \in \mathcal{C}} \frac{\sum_{(m_i,m_j)\in M'} \alpha_{i,j}}{\log(1 + \beta|M'|)}, \tag{2}$$

where $\beta > 0$ is a scaling factor. This technique favors smaller cliques. As smaller cliques contain a fewer number of segments, this technique avoids segmenting noun phrases into too many short, semantically incomplete phrases.

For better understanding of MPS, we summarize the high-level procedure, shown in Algorithm 3.

Cases	Segmented Noun Phrase	Extracted Predicate	Extracted Object
i)	DEP 1947年 建立 的 行政区划 Year of 1947 Establish *de* Administrative region (Administrative region established in 1947)	建立 (Established-in)	1947年 (1947)
ii)	DEP 罹患 肺结核 逝世者 Suffer from Pulmonary tuberculosis The dead (Dead people suffered from Pulmonary tuberculosis)	罹患 (Suffered-from)	肺结核 (Pulmonary tuberculosis)
	DEP 福克斯电视台 播放 的 电视连续剧 FOX Television Broadcast *de* TV series (TV series broadcast by FOX Television)	播放 (Broadcast-by)	福克斯电视台 (FOX Television)
iii)	意大利 作曲家 Italy Composer (Italian composer)	?	意大利 (Italy)

Fig. 4. Examples of three cases in CRG. In a full/partial relation triple $r(e, q_i)$ or $\tilde{r}(e, q_i)$, we only list the extracted relation predicate $r_v(e, q_i)$ and the object $r_o(e, q_i)$ or $\tilde{r}_o(e, q_i)$ in the table, with the name of the entity e omitted. The notation $\xrightarrow{DEP}$ refers to the case where the object depends on the relation predicate in the dependency parsing tree.

Algorithm 3. High-Level Algorithm of MPS

```
Input: Chinese noun phrase $p$, max iteration number $max$.
Output: Modifier-sensitive phrase segmentation result $ps(p)$.
Generate word segmentation result $ws(p)$ of phrase $p$ via Chinese word segmentation;
Construct a pruned NSG $G_p(M, E, W)$ based on $ws(p)$ by Algorithm 1;
Initialize the collection of cliques $\mathcal{C} = \emptyset$;
for each iteration $i = 1$ to $max$ do
    Detect the MEWC $M^{'}$ by Algorithm 2;
    Update $\mathcal{C} = \mathcal{C} \cup \{M^{'}\}$;
end for
Select the best clique $M^*$ from $\mathcal{C}$ by Eq. (2);
Generate the segmentation result $ps(p)$ based on $M^*$;
return Modifier-sensitive phrase segmentation result $ps(p)$.
```

3.3 Candidate Relation Generation

As discussed earlier, MPS is able to extract modifiers from Chinese noun phrases, which provide useful information about entities. However, it is still unclear how modifiers can be transformed into relation predicates and objects.

In the CRG component, for each entity-segmented noun phrase pair $(e, ps(p))$, let $R(p)$ be the collection of candidate relations derived based on modifiers in $ps(p)$. For each modifier q_i w.r.t. an entity e $(i < |ps(p)|)$, if q_i does not contain a specific named entity other than e, it is likely that this modifier does not express a relational fact. Hence, we simply discard it. Otherwise, it is probable that a relational fact can be derived from the entity e and the modifier q_i.

To generate candidate relations, it is vital to identify the relation predicates and objects from the modifiers. For example, given "1946年出生的(Born in 1946)" w.r.t. Donald Trump, CRG aims at detecting "出生(Born in)" as the relation predicate and "1946年(1946)" as the object. Hence, the full relation "(Donald Trump, born-in, 1946)" can be extracted. In Chinese, relation expressions are more irregular than English. Based on the syntactic structure of q_i, the operations of CRG can be divided into three cases, elaborated as follows, with four examples shown in Fig. 4:

Case i). If q_i is a verbal clause, the verb and the object in q_i can be directly extracted as the relation predicate and object (i.e., $r_v(e, q_i)$ and $r_o(e, q_i)$) of the full relation $r(e, q_i)$. The corresponding objects of verbs are determined by dependency parsing [54].

Case ii). In a few cases, the verb and the object are not clustered into one phrase q_i. This is because MPS is fully unsupervised with no pre-defined number of modifiers. Here, we propose a cross-modifier relation generation technique. If a named entity exists in q_i but no verbs are found, we further search q_{i-1} and q_{i+1}. If q_{i-1} or q_{i+1} is also not a complete verbal clause and contains a verb for the entity based on dependency parsing, the relation triple $r(e, q_i)$ can also be extracted, together with the predicate $r_v(e, q_i)$ and the object $r_o(e, q_i)$.

Case iii). If no verbs are detected for the entity, it means i) the verbal relation is expressed implicitly or ii) an error occurs in MPS. In this case, we extract a partial relation, denoted as $\tilde{r}(e, q_i)$ where q_i contains an entity as the relation object $\tilde{r}_o(e, q_i)$.

It should be further noted that not all candidate relations (especially partial relations) are correct in semantics. Consider the following segmented phrase w.r.t. Donald Trump:

Example 4. (Political figure in the United States)

美国	政治	人物
America	*Politics*	*Person*
(Modifier 1)	*(Modifier 2)*	*(Head)*

Extracted partial relations (English translation):
(Donald Trump, ?, America)
(Donald Trump, ?, Politics)

Although "Donald Trump" is related to "Politics", no explicit relation can be established. This relation triple can be automatically filtered by confidence assessment in MRPD.

3.4 Missing Relation Predicate Detector

After MPS and CRG, we continue to detect missing predicates for partial relations. Here, we introduce the probabilistic predicate detection algorithm with knowledge priors and a hypergraph-based random walk process to detect missing relation predicates for partial relations generated by CRG.

3.4.1 Model Formulation

Denote $\mathcal{V}$ as the collection of all possible relation predicates. Based on our previous research [14], the pattern-based approach for Chinese relation predicate extraction from noun phrases has very low accuracy (around 14 percent). Additionally, the detection of predicates from texts also suffers from low accuracy due to flexible expressions and the existence of light verb constructions [42]. Hence, in the NPORE system, we restrict our focus to only two knowledge sources in order to create the collection $\mathcal{V}$: i) all the relation predicates generated by CRG, due to the explicit verbal structures in noun phrases; and ii) relation predicates defined manually based on human common sense.

Given a partial relation $\tilde{r}(e, q_i)$, a basic model is the discriminative model $\Pr(v|\tilde{r}(e, q_i))$, which directly models the conditional probability of all relation predicates $v \in \mathcal{V}$, given a partial relation $\tilde{r}(e, q_i)$ as input. However, this model would suffer from the data sparsity problem due to the huge number of combinations of relation subject-object pair

TABLE 3
Examples of Spatial and Temporal Commonsense Relations

Type	Entity	Noun phrase
Spatial	复旦大学 Fudan University	上海高等院校 University in **Shanghai**
	故宫博物院 Palace Museum	北京博物馆 Museum in **Beijing**
Temporal	诺曼底战役 Battle of Normandy	**1944**年欧洲战场战役 Battle of Europe in **1944**
	安史之乱 An Lushan Rebellion	**8**世纪中国战争 Chinese War in **8th Century**

Locations and temporal expressions are printed in bold.

$(e, \tilde{r}_o(e, q_i))$ and the predicate v. Besides, as our system is fully unsupervised, learning parameters of $\Pr(v|\tilde{r}(e, q_i))$ would be highly challenging.

In this work, inspired by the text generation method [55], we model the problem of probabilistic predicate detection as a generative model $\Pr(v, \tilde{r}(e, q_i)) = \Pr(v)\Pr(\tilde{r}(e, q_i)|v)$ where $\Pr(v)$ and $\Pr(\tilde{r}(e, q_i)|v)$ are the prior and data likelihood models, respectively. For model prediction, based on the Bayesian rule, we have $\Pr(v|\tilde{r}(e, q_i)) = \frac{\Pr(v)\Pr(\tilde{r}(e,q_i)|v)}{\Pr(\tilde{r}(e,q_i))}$, where $\Pr(\tilde{r}(e, q_i))$ is treated as the normalization terms. The verb $\tilde{r}_{v^*}(e, q_i)$ is then selected by the following formula as the relation predicate between e and $\tilde{r}_o(e, q_i)$ in $\tilde{r}(e, q_i)$:

$$\tilde{r}_{v^*}(e, q_i) = \underset{v^{'} \in \mathcal{V}}{\operatorname{argmax}} \Pr(v^{'})\Pr(\tilde{r}(e, q_i)|v^{'}). \tag{3}$$

In the following, we introduce the definitions of the two models $\Pr(v)$ and $\Pr(\tilde{r}(e, q_i)|v)$ in detail.

3.4.2 The Prior Model

The prior model $\Pr(v)$ integrates both knowledge learned from previously extracted fully relations and human common sense. The first part of model $\Pr(v)$ is formulated based on Maximum Likelihood Estimation (MLE), i.e., $\Pr(v)^{MLE} = \frac{N_v}{N}$, where N and N_v denote the numbers of extracted full relation triples by CRG and a subset of these full relation triples with relation predicates as v.

According to our data-centric analysis, the majority of cases with missing relation predicates are due to the existence of implicit commonsense relations [15], [48], [49]. As a preliminary experiment, we randomly sample 300 partial relation triples from the experiment dataset, and observe that *spatial* and *temporal* commonsense relations are the two most frequent relation types with missing predicates. Here, spatial commonsense relations refer to the *located-in* relations between locations, while temporal commonsense relations refer to the *happened-in* relations between events and temporal expressions. In Chinese, the prepositions of spatial and temporal expressions are usually omitted (i.e., the counterpart of the preposition "in" in English). Examples of both types of commonsense relations are shown in Table 3. Therefore, we need to derive such relation triples by commonsense reasoning.[4]

Addition to MLE, we propose the commonsense probability distribution $\Pr(v)^{CS}$ to encode this observation. As an approximate estimation, let N_s, N_t and N_p be the numbers of locations, temporal expressions and all objects among all the candidate relations generated by CRG. The model $\Pr(v)^{CS}$ is defined as follows:

$$\Pr(v)^{CS} = \begin{cases} \frac{N_s}{N_p}, & v \text{ is spatial relation} \\ \frac{N_t}{N_p}, & v \text{ is temporal relation} \\ \frac{1}{|\mathcal{V}|-2}(1 - \frac{N_s+N_t}{N_p}), & \text{Otherwise} \end{cases}.$$

Combining the two probabilistic distributions $\Pr(v)^{MLE}$ and $\Pr(v)^{CS}$, we derive the full model of $\Pr(v)$

$$\Pr(v) = \lambda_1 \Pr(v)^{MLE} + \lambda_2 \Pr(v)^{CS} + (1 - \lambda_1 - \lambda_2)\frac{1}{|\mathcal{V}|}, \tag{4}$$

where λ_1 and λ_2 are balancing hyper-parameters with $0 < \lambda_1 < 1$, $0 < \lambda_2 < 1$ and $\lambda_1 + \lambda_2 < 1$. $(1 - \lambda_1 - \lambda_2)\frac{1}{|\mathcal{V}|}$ gives a smoothing effect on the verb distribution $\Pr(v)$ based on the Jelinek-Mercer smoothing technique [56].

3.4.3 The Data Likelihood Model

The data likelihood model estimates $\Pr(\tilde{r}(e, q_i)|v)$ in an unsupervised manner based on a hypergraph-based random walk process. We first introduce two scores over the graph to define the random walk process.

Predicate Coherence Score. The predicate coherence score is defined between a predicate $v \in \mathcal{V}$ and a partial relation $\tilde{r}(e, q_i)$, denoted as $w_p(v, \tilde{r}(e, q_i))$. It measures whether a predicate v is suitable to describe the relation between the subject e and the object $\tilde{r}_o(e, q_i)$ of the partial relation $\tilde{r}(e, q_i)$. To speed up the process of verb retrieval, we construct a sentence-level inverted index over the background text corpus D using Apache Lucene.[5] The query "e AND $\tilde{r}_o(e, q_i)$" is used to retrieve a collection of sentences S. For each sentence $s \in S$, we extract contextual verbs from s which may indicate the relations between e and $\tilde{r}_o(e, q_i)$. Inspired by [29], we regard a verb to be contextual if it is in the dependency chain between e and $\tilde{r}_o(e, q_i)$. Let $V(e, q_i)$ be the contextual verb collection, $c(v)$ be the count of v extracted from S. The score $w_p(v, \tilde{r}(e, q_i))$ is defined as

$$w_p(v, \tilde{r}(e, q_i)) = \frac{1}{Z}\sum_{v^{'} \in V(e,q_i)} c(v^{'})\cos(\text{vec}(v), \text{vec}(v^{'})),$$

where $Z = \sum_{v^{'} \in V(e,q_i)} c(v^{'})$ is the normalization factor.

Relation Similarity Score. The relation similarity score $w_r(r(e, q_i), r(e^{'}, q_i^{'}))$ is defined over two relations $r(e, q_i)$ and $r(e^{'}, q_i^{'})$. It computes the degree that the two relations may have the same predicate, based on the similarity of word embeddings of their subjects and objects:[6]

$$\begin{aligned} w_r(r(e, q_i), r(e^{'}, q_i^{'})) = & \frac{1}{2}(\cos(\text{vec}(e), \text{vec}(r_o(e, q_i))) \\ & + \cos(\text{vec}(e^{'}), \text{vec}(r_o(e^{'}, q_i^{'})))). \end{aligned} \tag{5}$$

[4] Note that a majority of existing research works focus harvesting of spatial/temporal relations [48], [49]. This practice is also applied in the YAGO2 knowledge base [25]. Harvesting more types of commonsense knowledge automatically is left as future research.

[5] http://lucene.apache.org

[6] For simplicity, we do not distinguish full or partial relations here, and denote them as $r(e, q_i)$ and $r(e^{'}, q_i^{'})$ uniformly.

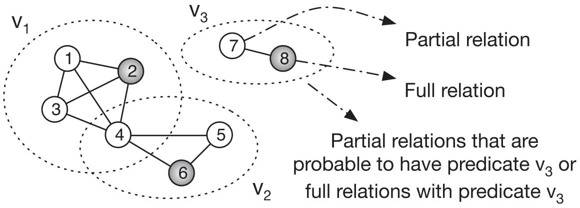

v1
v3
7
8
1
2
3
4
5
6
v2
Partial relation
Full relation
Partial relations that are probable to have predicate v3 or full relations with predicate v3

Fig. 5. A toy example of the graph structure of PHN. Small circles 1-8 refer to nodes (i.e., full or partial relations, depending on the color). Large circles v_1, v_2 and v_3 refer to hyper-edges (i.e., relation predicates). Lines refer to all possible routes that random walkers can travel.

Based on the two scores, we construct the hypergraph for the random walk process. We introduce the concept of Predicate-based Hypergraph Network $H(\mathcal{R},\mathcal{V})$:

Definition 5 (Predicate-based Hypergraph Network). *A PHN $H(\mathcal{R},\mathcal{V})$ is a hypergraph model where $\mathcal{R}$ is the node collection, corresponding to all full and partial relations generated by CRG and $\mathcal{V}$ is the hyper-edge collection, corresponding to all predicates.*

In PHM $H(\mathcal{R},\mathcal{V})$, each hyper-edge (i.e., relation predicate) $v \in \mathcal{V}$ is associated with a collection of full and partial relations. We require that i) a full relation $r(e,q_i)$ is in the hyper-edge v iff $r_v(e,q_i) = v$; and ii) a partial relation $\tilde{r}(e,q_i)$ is in the hyper-edge v iff the score $w_p(v,\tilde{r}(e,q_i)) > \tau_1$ where $\tau_1 \in (0,1)$ is a predefined hyper-parameter. Hence, we can see that all nodes in the hyper-edge v are either full relations with predicate v or partial relations that are highly probable to have the predicate v. Note that it is possible for a partial relation to be in more than one hyper-edge. Refer to a toy example in Fig. 5.

Hypergraph-Based Random Walk Process. We further define the concept of neighborhood of any nodes in PHN:

Definition 6 (Neighborhood of PHN). *The neighborhood $Nb(r(e,q_i))$ of a node $r(e,q_i)$ in PHN $H(\mathcal{R},\mathcal{V})$ is a collection of nodes such that a node $r(e^{'},q_i^{'}) \in Nb(r(e,q_i))$ iff there exists a hyper-edge $v \in \mathcal{V}$ with $(r(e,q_i), r(e^{'},q_i^{'})) \in v$.*

Consider the example in Fig. 5. The neighborhoods of Nodes 1 and 4 are {2,3,4} and {1,2,3,5,6}, respectively.

The hypergraph-based random walk process is as follows. Let $\mathcal{R}_v$ be the node collection in hyper-edge $v \in \mathcal{V}$ that all correspond to full relations. For each $v \in \mathcal{V}$, a separate random walker starts from each $r(e,q_i) \in \mathcal{R}_v$, and goes to a neighbor node $r(e^{'},q_i^{'}) \in Nb(r(e,q_i))$ with probability $\propto w_r(r(e,q_i), r(e^{'},q_i^{'}))$. The process iterates after a sufficient number of walks. Finally, each partial relation $\tilde{r}(e,q_i)$ receives a score $s_v(\tilde{r}(e,q_i))$, indicating the number of visits of all random walkers. $\Pr(\tilde{r}(e,q_i)|v)$ is approximated by

$$\Pr(\tilde{r}(e,q_i)|v) = \frac{s_v(\tilde{r}(e,q_i))}{\sum_{\tilde{r}(e^{'},q_i^{'})\in\mathcal{R}} s_v(\tilde{r}(e^{'},q_i^{'}))}. \qquad (6)$$

It is noteworthy that it is highly possible for a random walker starting from a hyper-edge to go to another hyper-edge. For example, in Fig. 5, the random walker may go from Node 1 to Node 5 (from hyper-edge v_1 to v_2). This setting assigns a part of the probability to nodes outside the candidate sets, which addresses the problem where the candidate generation technique does not yield 100 percent recall. Readers can also refer to a summarization of the probabilistic predicate detection process in Algorithm 4.

Algorithm 4. Missing Predicate Detection

```
Generate the predicate collection $\mathcal{V}$;
for each predicate $v \in \mathcal{V}$ do
    Estimate prior probability $\Pr(v)$ by Eq. (4);
    Generate full relations associated with $v$ as $\mathcal{R}_v$;
end for
Construct the PHN $H(\mathcal{R},\mathcal{V})$;
for each predicate $v \in \mathcal{V}$ do
    Run the random walk process based on Eq. (5);
    for each partial relation $\tilde{r}(e,q_i)$ do
        Compute $\Pr(\tilde{r}(e,q_i)|v)$ by Eq. (6);
        Predict the relation predicate $\tilde{r}_{v^*}(e,q_i)$ by Eq. (3);
    end for
end for
```

3.4.4 Confidence Assessment

Finally, we filter out noisy and meaningless extractions. We observe that most extraction errors occur when the algorithm extracts meaningless "relation predicates". This phenomenon is also consistent with our previous research [14], [45] and classical OIE research [29], [30]. Let $\tilde{c}(v)$ be the number of extracted full and partial relations with predicates predicted as v. The confidence score of each full relation $r(e,q_i)$ is defined as: $conf(r(e,q_i)) = \tilde{c}(r_v(e,q_i))$.

For each partial relation $\tilde{r}(e,q_i)$, we add another factor to measure whether the prediction of the relation predicate $\tilde{r}_{v^*}(e,q_i)$ by Eq. (3) is confident. From a probabilistic perspective, if the predication is confident, the score $\max_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e,\ q_i)|v)$ should be larger than $\mathrm{secmax}_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e, q_i)|v)$ by a large margin, where $\mathrm{secmax}_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e,q_i)|v)$ refers to the second largest value among all $\Pr(v) \Pr(\tilde{r}(e,q_i)|v)$ $(v \in \mathcal{V})$. Hence, the confidence score $conf(\tilde{r}(e,q_i))$ is defined as

$$conf(\tilde{r}(e,q_i)) = \tilde{c}(r_{v^*}(e,q_i)) \cdot \frac{\max_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e,q_i)|v)}{\max_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e,q_i)|v) + \mathrm{secmax}_{v\in\mathcal{V}} \Pr(v) \Pr(\tilde{r}(e,q_i)|v)}.$$

In the NPORE system, we employ a pre-defined threshold τ_2 to filter out relations if $conf(r(e,q_i)) < \tau_2$ for full relations or $conf(\tilde{r}(e,q_i)) < \tau_2$ for partial relations.

4 Experiments

In this section, we conduct experiments to evaluate NPORE in various aspects. We also compare it with state-of-the-art to make the convincing conclusion.

4.1 Data Source and Experimental Settings

The collection of entity-noun phrase pairs is taken from the Chinese Wikipedia category system of version January 20th, 2017.[7] We follow the common practice in [14], [36], [37] to extract the pairs. In Wikipedia, the titles of the pages are

[7] http://download.wikipedia.com/zhwiki/20170120/

treated as names of entities and the categories are treated as the noun phrases related to these entities. Because several Wikipedia pages are not about entities, after filtering of indirect, template, stub and disambiguation pages, we obtain 0.6M entities and 2.4M entity-noun phrase pairs.

In this paper, we use the FudanNLP toolkit [54] for basic Chinese NLP analysis, such as Chinese word segmentation, POS and NER. To improve the recall of NER, we also add the names of Chinese Wikipedia entities as a dictionary in FudanNLP. Because the size of the Chinese Wikipedia corpus is relatively small, we also crawl 1.3M articles from *Baidu Baike* (a large Chinese online encyclopedia) to train language models based on [57]. In total, we have a large Chinese corpus, consisting of 2M articles as the background text corpus. The dimension of word embeddings is 100.

In the implementation of the NPORE system, the default hyper-parameter settings are shown as follows: $n=3$, $\gamma=0.3$, $\beta=5$, $\lambda_1=0.6$, $\lambda_2=0.3$, $\tau_1=0.7$ and $\tau_2=20$. The MEWC algorithm is run for 3 times in MPS to generate the clique collection $\mathcal{C}$. For the hypergraph-based random walk process, we send out ten random walkers from each starting points in the PHM and run for 500 steps. We also study how different values of these hyper-parameters can affect the performance in the experiments. All the algorithms of the NPORE system are implemented in JAVA and run in a single PC machine with 2.9 GHz CPU and 16 GB memory.

4.2 Baselines

Although there are abundant RE approaches, most of them can not be taken as baselines due to the difference between these works and ours. Because our system works in open domains, we compare our method against several OIE systems, especially noun phrase-based OIE systems. We also employ knowledge extraction methods for Wikipedia categories as baselines, summarized as follows:

Classical Sentence-Based OIE. Because there are significant linguistic differences between English and Chinese, we regard the state-of-the-art Chinese OIE system ZORE [42] as a strong baseline of classical OIE systems. To accommodate noun phrase-based RE, we use entity-noun phrase pairs as queries to search for sentences in the background text corpus D. For each entity-noun phrase pair, we take top-5 sentences returned by the Apache Lucene search engine as the input sentences of ZORE. The implementation and detailed parameter settings of ZORE are taken from the authors' original source.[8]

Neural Sentence-Based OIE. The encoder-decoder network is the state-of-the art neural network architecture for OIE. We employ the neural network proposed in [34] to extract relations from the same sentences as the inputs of ZORE [42]. In this model, we use three-layer BiLSTMs as the encoder and the decoder (as the default settings reported in [34]). The word embeddings are trained over our corpus D by ourselves with the dimensionality set to 100.

Classical Noun Phrase-Based OIE. Because we consider RE from noun phrases, we take a recent noun phrase-based OIE system RELNOUN [18] as a baseline. As it considers English patterns only, we manually translate such noun phrase-based patterns into Chinese and and implement a variant of CN-RELNOUN to extract the relations.

Neural Noun Phrase-Based OIE. To apply cutting-edge deep learning techniques for noun phrase-based OIE, we implement a variant of Soares et al. [58] as a baseline. In the implementation, we use our MPS and CRG modules as the first two steps. As our task is unsupervised, we take the relation pairwise similarity model in [58] to approximate $\Pr(\tilde{r}(e,q_i)|v)$ in MPRD. The random walker travels from one node $r(e,q_i)$ to another $r(e^{'},q_i^{'})$ with probability $\propto s_n(r(e,q_i),r(e^{'},q_i^{'}))$ where $s_n(\cdot,\cdot)$ is the predicted relation similarity score. The relation statements used in [58] are the same as what we used for ZORE [42]. The underlying Chinese BERT model [59] is downloaded from GitHub.[9]

Wikipedia-Specific Methods. We also employ two methods designed for RE from Wikipedia categories as baselines. The first method is Nastase and Strube [36], which employ prepositions in Wikipedia category patterns to discover relations. The second method takes from our previous work [14], [45], which mines frequent textual patterns by graph-based mining and achieves the highest performance previously. Because the method of Nastase and Strube [36] focuses on the English language only, we take the variant for Chinese (i.e., CN-WikiRe [45]) as our baseline. The implementation details are described in [45].

In a few cases, the Chinese Wikipedia categories are verbal clauses, such as "1946年出生 (Born in 1946)" for Donald Trump. The extraction of relations from these categories has been addressed in several baselines [14], [18], [36]. To make NPORE comparable with these baselines, we add an additional step to the system. If the input category is verbal, we regard all the segmented elements generated by MPS as modifiers, rather than modifiers plus one head word.

4.3 Evaluation Metrics

As a variant of RE systems, Precision, Recall and F1 score would be the first choices to evaluate NPORE. However, this evaluation method is infeasible. Re-consider Example 3. Given the modifier "美国(America)" in "美国政治人物 (Political figure in the United States)" and "Donald Trump" as input, four possible extracted relation triples are:

Possible extracted relations (English translation):
(Donald Trump, has-nationality, American)
(Donald Trump, born-in, America)
(Donald Trump, works-in, America)
(Donald Trump, is-leader-of, America)

All four relation predicates are valid. Hence, the "gold-standard" for computing Recall and F1 score for our system can not be established. The difficulty of evaluating such systems is also an open research challenge in OIE [9]. To address this issue, Mausam et al. [30] propose to use the Yield score to evaluate OIE systems. The score is calculated by multiplying the number of extractions (i.e., relation triples) by their precision scores. Hence, it is equal to the (estimated) number of correct extractions. In this paper, we employ three metrics to compare our system against all the baselines, i.e., #Relations (total number of extracted relation triples), Precision (the ratio

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

[9] https://github.com/google-research/bert/

TABLE 4
General Performance Comparison of Different Methods Over Four Subsets of Chinese Wikipedia Categories

Method	#Rel.	Pre.	Yield	#Rel.	Pre.	Yield
Domain		**General**			**Politics**	
Nastase and Strube [36]	87	41.7%	41	84	57.1%	48
Pal and Mausam [18]	31	93.5%	29	35	88.6%	31
Qiu and Zhang [42]	28	75.0%	21	34	76.4%	26
Wang et al. [14]	193	**94.3%**	182	193	**95.9%**	185
Cui et al. [34]	52	51.9%	27	51	43.1%	22
Soares et al. [58]	213	75.6%	161	154	70.1%	108
NPORE	**289**	92.7%	**268**	**314**	93.9%	**295**
Domain		**Entertainment**			**Military**	
Nastase and Strube [36]	102	39.2%	40	76	53.9%	41
Pal and Mausam [18]	42	88.1%	37	34	82.3%	28
Qiu and Zhang [42]	21	76.2%	16	32	81.2%	26
Wang et al. [14]	204	**95.1%**	194	188	**96.3%**	181
Cui et al. [34]	54	48.1%	26	44	56.8%	25
Soares et al. [58]	163	60.1%	98	201	69.2%	139
NPORE	**324**	92.3%	**299**	**274**	94.2%	**258**

of the numbers of extracted corrected relation triples and all extracted relation triples), and and Yield score (the product of #Relations and Precision).

4.4 Overall Performance Comparison

We report the overall performance of the NPORE system and compare it with baselines. After we run all the systems over the entire Wikipedia entity-category dataset, we sample four subsets to evaluate NPORE and baselines in terms of #Relations, Precision and Yield score. Each subset contains 300 entities and their corresponding categories. The first subset is uniformly sampled from the entire dataset, denoted as “General”. The remaining three subsets are domain-specific datasets, which are related to the three domains: Politics, Entertainment and Military. In this work, we employ heuristic rules to extract domain-specific entities. For example, we regard an entity is in the entertainment domain if there exists one category that ends in the following words: 歌手(singer), 音乐(music), 电影(movie), 娱乐(entertainment), etc. The example entities that belong to the three domains are shown in Table 5. Due to space limitation, we omit the details of the extraction process here.

Table 4 illustrates the experimental results of the overall performance of all the systems. Generally, the metric scores of all methods are consistent over all four subsets. As seen, sentence-based OIE systems [34], [42] (either classical or neural-based) do not yield satisfactory results. This is because sentence-based OIE systems extract relations primarily based on subject-verb-object structures, which inevitably suffers from data noise and low recall. In our datasets, a lot of relations within Chinese noun phrases are expressed more concisely and implicitly. Hence, sentence-based methods are not suitable for noun phrase-based RE. Classical noun phrase-based OIE approaches (e.g., [18]) have relatively high precision but low coverage, leading to the low Yield score. For English, due to the frequent usage of prepositions, the patterns “[...] is [...] of [...]” and “[...] is [...] from [...]” are highly effective in RENOUN [18]. The lack of such expressions in Chinese result in low recall. In our approach, we do not rely on fixed patterns to extract relations. Instead, our three-layer framework can be viewed as a “divide-and-conquer” strategy to harvest relations. As for Soares et al. [58], although deep language models such as BERT [59] are employed to learn relation similarities, this method is not suitable for Chinese short texts. This is because the encoding process of Soares et al. [58] still requires the detection of relation statements in the corpus, which are extremely sparse.

TABLE 5
Examples of Domain-Specific Entities

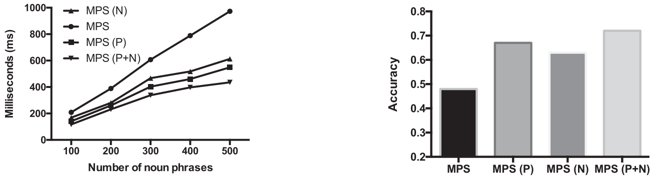
Domain	Entities
Politics	唐纳德・特朗普 (Donald Trump), 罗纳德・里根 (Ronald Reagan), 英国议会 (Parliament of the UK)
Entertainment	王菲 (Faye Wong), 肖申克的救赎 (The Shawshank Redemption), 奥斯卡金像奖 (Academy Award)
Military	航空母舰 (Aircraft Carrier), 氢弹 (Thermonuclear weapon), 中途岛海战 (Battle of Midway)

TABLE 6
Summary of Extraction Performance Overall All Wikipedia Categories

Method	#Relations	Precision	Yield score
		(Estimated)	(Estimated)
Nastase and Strube [36]	165K	58.6%	96.7K
Pal and Mausam [18]	65K	92.8%	60.3K
Qiu and Zhang [42]	42K	82.3%	34.6K
Wang et al. [14]	357K	**97.4%**	347.7K
Cui et al. [34]	89K	51.2%	45.6K
Soares et al. [58]	420K	72.3%	303.7K
NPORE	**554K**	95.4%	**528.5K**

Compared to Wikipedia-based baselines, for the general domain, NPORE extracts 149.7 percent as many as relations compared to the strongest baseline [14] and has a comparable precision of 92.7 percent. Overall, the NPORE system achieves a 47.3 percent higher Yield score than [14], indicating the effectiveness of the proposed approach. Regarding the three specific domains, the trends of performance are similar to the general domain. The improvement of NPORE is mostly due to the data-driven designs of our system. To elaborate, previous approaches (e.g., [14], [36]) either use manually-defined or automatically-mined patterns for RE, which may only consider a small portion of circumstances. Our system imposes few hypotheses on the input data and is fully data-driven, capable of extracting more relations. In summary, the superiority of the NPORE system can be easily proven through the four sets of experiments and our analysis.

We further estimate the general extraction performance over the entire dataset. We randomly sample 500 relations from the complete relation collections generated from all the approaches and ask human annotators to evaluate the precision manually. Based on the estimated precision and the numbers of extracted relations, the total Yield scores can be also estimated. The results are summarized in Table 6. Overall, NPORE harvests 554K relations, at the precision of 95.4 percent. It outperforms state-of-the-art [14] by



(a) Efficiency study

(b) Effectiveness study

Fig. 6. Efficiency and effectiveness study of MPS.

55.2 percent of #Relations and 52.0 percent of the Yield score, without sacrificing too much in precision.

4.5 Detailed Analysis of NPORE

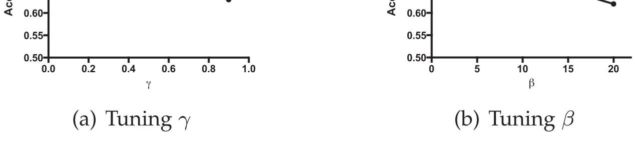
We tune the hyper-parameters of NPORE and analyze the performance of the NPORE components in detail.

Efficiency and Effectiveness of Clique Detection. A preliminary experiment shows that over 90 percent of the modifiers contain fewer than four segmented Chinese words (not Chinese characters). Therefore, we set the n-gram factor as $n = 3$. One can also set a larger value for n but we suggest that such practice increases the computational complexity of MPS and makes the perplexity of language models larger.

Next, we focus on the efficiency and effectiveness of the MEWC algorithm and linguistic constraints. We set γ and β as their default values and conduct two sets of experiments, considering four settings: "MPS" (the MEWC algorithm without any constraints), "MPS (P)" (with positive constraints), "MPS (N)" (with negative constraints) and "MPS (P+N)" (with both types of constraints).

For efficiency study, we randomly sample 100$\sim$500 noun phrases, perform MPS in four settings and record the execution time. For effectiveness study, we ask human annotators to label the correctness of phrase segmentation results over the "General" subset. The results are shown in Fig. 6. As seen, the proposed approach with both types of constraints are highly efficient. Generally speaking, the usage of both constraints reduces the running time to approximately 50 percent of the original time. It can be further noted that the constraints can also improve the accuracy of phrase segmentation by considering the linguistic characteristics of Chinese noun phrases. Additionally, we tune the values of two parameters γ and β, with results illustrated in Fig. 7. The experimental results show that the distributional score $w_d(i, j)$ contributes more than the statistical score $w_s(i, j)$. The highest performance can be achieved when $\beta = 5$.

Study of Candidate Relation Generation. In this component, the cross-modifier relation generation technique is proposed to improve the ratio of full relations with detected predicates. To verify this hypothesis, we report the percentages of such relations under two settings: with and without the cross-modifier relation generation technique. The results are shown in Table 7. As seen, the percentages have varying degrees of improvement, from 2.4 to 4.6 percent.



(a) Tuning γ

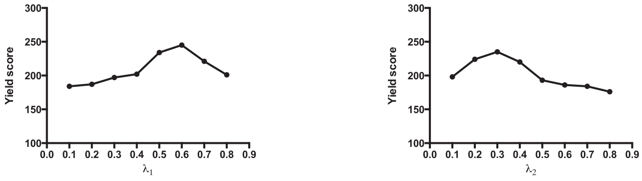
(b) Tuning β

Fig. 7. Parameter analysis w.r.t. γ and β of the MPS component.

TABLE 7
Percentage of Generated Relations With Relation Predicates

Setting	General	Politics	Entertainment	Military
w/o. CMRG	12.2%	11.0%	15.4%	13.3%
w. CMRG	**16.8%**	**14.6%**	**17.8%**	**15.8%**
Improvement	**+4.6%**	**+3.6%**	**+2.4%**	**+2.5%**

CMRG refers to "cross-modifier relation generation".



(a) Tuning λ_1

(b) Tuning λ_2

Fig. 8. Parameter analysis w.r.t. λ_1 and λ_2 of the MRPD component.

TABLE 8
Examples of Relation Predicates With High and Low $\tilde{c}(v)$ Scores

Relation Predicate	$\tilde{c}(v)$	Relation predicate	$\tilde{c}(v)$
位于(located-in)	124K	警告 (warn)	19
发生 (happened-in)	53K	民变 (civil commotion)	16
毕业 (graduated-from)	44K	冷藏 (refrigerate)	14
建立 (established-in)	23K	集会 (assembly)	8

Study of Missing Relation Predicate Detection. For the MRPD component, we tune the parameters λ_1 and λ_2. The performance is evaluated based on the Yield score of the "General" subset. Fig. 8 illustrates the change of Yield scores when the two parameters vary. In each experiment, we fix one parameter as 0.1 and tune the value of the other. We can see that the changes of λ_1 and λ_2 reflect the relative importance of three prior probabilistic distributions. As for the hyper-graph based random walk process, we set $\tau_1 = 0.7$ because we observe that when $\tau_1 \geq 0.7$, the subject-object pairs usually share the same relation predicate. We further tune the value of τ_2 for confidence-based filtering. In Table 8, we list examples of extracted relation predicates with high and low $\tilde{c}(v)$ scores. We can see that words with low $\tilde{c}(v)$ scores are usually not relation predicates due to POS and parsing errors. We set $\tau_2 = 20$ because most words with $\tau_2 < 20$ are not proper relation predicates.

4.6 Error Analysis and Case Studies

To indicate how our method can be improved in the future, we analyze errors in extracted relations. 300 extracted errors are re-presented to the human annotators to distinguish the types of errors. The examples are shown in Table 9. The first type of errors can be summarized as *incomplete object extraction* (IOE, about 32 percent), which means the extracted objects in relation triples are not semantically complete. For example, "Seoul Special City" is the name of the complete entity, but MPS separates "Seoul" and "Special City", leading to the error. The relation "(Wang Jiaji, has-position, president)" is

TABLE 9
Two Major Types of Extraction Errors and Their Examples

Error Type	Extracted Relations	Corrections
IOE	(王家骥,职务,校长) (Wang Jiaji, has-position, president)	(王家骥,职务,国立台东大学校长) (Wang Jiaji, has-position, president of National Taitung University)
	(梁耀燮,出身,特别市) (Yang Yo-seop, originated-from, special city)	(梁耀燮,出身,首尔特别市) (Yang Yo-seop, originated-from, Seoul Special City)
EPD	(第65届戛纳电影节,担任,戛纳) (65th Cannes Film Festival, work-as, Cannes)	(第65届戛纳电影节,位于,戛纳) (65th Cannes Film Festival, located-in, Cannes)
	(台北101,生于,台湾) (Taipei 101, born-in, Taiwan)	(台北101,位于,台湾) (Taipei 101, located-in, Taiwan)

correct in syntax but it is not much meaningful due to its semantic incompleteness. The complete relation should be "(Wang Jiaji, has-position, president of National Taitung University)". The incomplete extraction problem also contributes to a large proportion of errors in other OIE systems [8], [29], [30]. This problem is more difficult for our system due to the flexible expressions of Chinese. The remaining errors occurs when the detection of missing relation predicates is incorrect. This type is called *error predicate detection*, abbreviated as EPD. For example, the NPORE system predicts there is a *located-in* relation between an event entity and another entity tagged as a location. However, a few person names in our datasets are tagged as locations by NER errors. Hence, the *located-in* relations do not hold. Additionally, during the hypergraph-based random walk process, incorrect relation predicates may be predicted due to NLP parsing errors and noises, with examples illustrated in Table 9.

A more important problem to be addressed is the missing extraction problem. Table 10 illustrates two Wikipedia categories w.r.t. Donald Trump that contain relational facts un-extracted by NPORE. The missing relations should be "(Donald Trump, survived, assassination attempt)" and "(Donald Trump, worked-in, real estate)". Based on our research and the survey [9], this issue is even difficult to be evaluated, not to mention solving it completely.

4.7 Discussion on Further Research

Although we have achieved some success, knowledge extraction from Chinese noun phrases still faces challenges. The key barriers lie in two aspects: i) the lack of (relatively) fixed syntactic/lexical expressions in Chinese and ii) the large amount of commonsense knowledge left unexpressed inside noun phrases. In the future, our work can be extended by addressing the following issues: i) improving our work by using more conceptual and commonsense knowledge such as [47], [60]; ii) extending this system to the Web scale, which automatically extracts descriptive noun phrases and entities from free texts and extracts the relations from them; iii) developing a comprehensive framework to evaluate noun phrase-based OIE; and iv) studying how neural networks can be applied to RE from noun phrases. While neural networks are suitable for learning implicit, high-dimensional representations, it is still a challenge for neural networks be used for short-text knowledge extraction that requires explicit reasoning and human common sense.

TABLE 10
Categories w.r.t. Donald Trump With Relations Un-Extracted by NPORE

Wikipedia Category	English Translation
暗杀未遂幸存者	Attempted assassination survivor
美国房地产商	US real estate developer

5 Conclusion

In this work, we present a fully unsupervised, open-domain Noun Phrase based Open RE system for RE from Chinese noun phrases. NPORE contains three major components: Modifier-sensitive Phrase Segmenter, Candidate Relation Generator and Missing Relation Predicate Detector. Especially, the system integrates with a graph clique mining algorithm to chunk Chinese noun phrases into modifiers and head words, which are utilized to generate candidate relation triples. We further propose a probabilistic predicate detection algorithm with Bayesian knowledge priors and a hypergraph-based random walk process to detect missing relation predicates. Experimental results over Chinese Wikipedia show that NPORE outperforms state-of-the-art.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (No. 2016YFB1000904).

References

[1] X. Ren *et al.*, "CoType: Joint extraction of typed entities and relations with knowledge bases," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 1015–1024.

[2] R. Lu, X. Jin, S. Zhang, M. Qiu, and X. Wu, "A study on big knowledge and its engineering issues," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 9, pp. 1630–1644, Sep. 2019.

[3] J. Shen *et al.*, "HiExpan: Task-guided taxonomy construction by hierarchical tree expansion," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2018, pp. 2180–2189.

[4] M. Yu, W. Yin, K. S. Hasan, C. N. dos Santos, B. Xiang, and B. Zhou, "Improved neural relation detection for knowledge base question answering," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 571–581.

[5] S. T. Hsu, C. Moon, P. Jones, and N. F. Samatova, "An interpretable generative adversarial approach to classification of latent entity relations in unstructured sentences," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 5181–5188.

[6] S. Su, N. Jia, X. Cheng, S. Zhu, and R. Li, "Exploring encoder-decoder model for distant supervised relation extraction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 4389–4395.

[7] P. Gupta, B. Roth, and H. Schütze, "Joint bootstrapping machines for high confidence relation extraction," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 26–36.

[8] K. Gashteovski, R. Gemulla, and L. D. Corro, "MinIE: Minimizing facts in open information extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2630–2640.

[9] C. Niklaus, M. Cetto, A. Freitas, and S. Handschuh, "A survey on open information extraction," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 3866–3878.

[10] X. Han and L. Sun, "Global distant supervision for relation extraction," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 2950–2956.

[11] Y. Su, H. Liu, S. Yavuz, I. Gur, H. Sun, and X. Yan, "Global relation embedding for relation extraction," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 820–830.

[12] S. Saha and Mausam, "Open information extraction from conjunctive sentences," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 2288–2299.

[13] B. Xu, C. Xie, Y. Zhang, Y. Xiao, H. Wang, and W. Wang, "Learning defining features for categories," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 3924–3930.

[14] C. Wang, Y. Fan, X. He, and A. Zhou, "Learning fine-grained relations from chinese user generated categories," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2577–2587.

[15] N. Tandon, G. de Melo, F. M. Suchanek, and G. Weikum, "WebChild: Harvesting and organizing commonsense knowledge from the web," in *Proc. 7th ACM Int. Conf. Web Search Data Mining*, 2014, pp. 523–532.

[16] C. C. Xavier and V. L. S. de Lima, "Boosting open information extraction with noun-based relations," in *Proc. 9th Int. Conf. Lang. Resou. Eval.*, 2014, pp. 96–100.

[17] M. Yahya, S. Whang, R. Gupta, and A. Y. Halevy, "ReNoun: Fact extraction for nominal attributes," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 325–335.

[18] H. Pal and Mausam, "Demonyms and compound relational nouns in nominal open IE," in *Proc. 5th Workshop Autom. Knowl. Base Construction*, 2016, pp. 35–39.

[19] C. N. Li and S. A. Thompson, "Mandarin chinese: A functional reference grammar," *J. Asian Stud.*, vol. 42, no. 3, pp. 10–12,1989.

[20] C.-T. J. Huang, *Logical Relations in Chinese and the Theory of Grammar*. New York, NY, USA: Taylor & Francis, 1998.

[21] X. Chen, Z. Shi, X. Qiu, and X. Huang, "Adversarial multi-criteria learning for chinese word segmentation," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 1193–1203.

[22] R. Fu, J. Guo, B. Qin, W. Che, H. Wang, and T. Liu, "Learning semantic hierarchies via word embeddings," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, 2014, pp. 1199–1209.

[23] C. Wang, J. Yan, A. Zhou, and X. He, "Transductive non-linear learning for chinese hypernym prediction," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 1394–1404.

[24] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 697–706.

[25] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum, "YAGO2: A spatially and temporally enhanced knowledge base from wikipedia," *Artif. Intell.*, vol. 194, pp. 28–61, 2013.

[26] C. Lyu, Y. Zhang, and D. Ji, "Joint word segmentation, POS-tagging and syntactic chunking," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 3007–3014.

[27] I. Dagan and V. Shwartz, "Paraphrase to explicate: Revealing implicit noun-compound relations," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1200–1211.

[28] C. Wang, X. He, and A. Zhou, "A short survey on taxonomy learning from text corpora: Issues, resources and recent advances," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1190–1203.

[29] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007, pp. 2670–2676.

[30] Mausam, M. Schmitz, S. Soderland, R. Bart, and O. Etzioni, "Open language learning for information extraction," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.*, 2012, pp. 523–534.

[31] X. Zeng, S. He, K. Liu, and J. Zhao, "Large scaled relation extraction with reinforcement learning," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 5658–5665.

[32] W. Y. Wang, W. Xu, and P. Qin, "DSGAN: Generative adversarial training for distant supervision relation extraction," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 496–505.

[33] Y. Lin, Z. Liu, and M. Sun, "Neural relation extraction with multilingual attention," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 34–43.

[34] L. Cui, F. Wei, and M. Zhou, "Neural open information extraction," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 407–413.

[35] Q. Zhu, X. Ren, J. Shang, Y. Zhang, F. F. Xu, and J. Han, "Open information extraction with global structure constraints," in *Companion Proc. Web Conf.*, 2018, pp. 57–58.

[36] V. Nastase and M. Strube, "Decoding wikipedia categories for knowledge acquisition," in *Proc. 23rd Nat. Conf. Artif. Intell.*, 2008, pp. 1219–1224.

[37] M. Pasca, "German typographers vs. german grammar: Decomposition of wikipedia category labels into attribute-value pairs," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, 2017, pp. 315–324.

[38] V. Shwartz and C. Waterson, "Olive oil is made of olives, baby oil is made for babies: Interpreting noun compounds using paraphrases in a neural model," in *Proc. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2018, pp. 218–224.

[39] P. Nakov and M. A. Hearst, "Using verbs to characterize noun-noun relations," in *Proc. Int. Conf. Artif. Intell.: Methodology Syst. Appl.*, 2006, pp. 233–244.

[40] A. Grycner and G. Weikum, "POLY: Mining relational paraphrases from multilingual sentences," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 2183–2192.

[41] I. Hendrickx, Z. Kozareva, P. Nakov, D. Ó. Séaghdha, S. Szpakowicz, and T. Veale, "SemEval-2013 task 4: Free paraphrases of noun compounds," in *Proc. 2nd Joint Conf. Lexical Comput. Semantics: Proc. 7th Int. Workshop Semantic Eval.*, 2013, pp. 138–143.

[42] L. Qiu and Y. Zhang, "ZORE: A syntax-based system for chinese open relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1870–1880.

[43] Y. Tseng, L. Lee, S. Lin, B. Liao, M. Liu, H. Chen, O. Etzioni, and A. Fader, "Chinese open relation extraction for knowledge acquisition," in *Proc. 14th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2014, pp. 12–16.

[44] S. Jia, S. E, M. Li, and Y. Xiang, "Chinese open relation extraction and knowledge base establishment," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 17, no. 3, pp. 15:1–15:22, 2018.

[45] C. Wang, Y. Fan, X. He, and A. Zhou, "Decoding chinese user generated categories for fine-grained knowledge harvesting," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 8, pp. 1491–1505, Aug. 2019.

[46] D. B. Lenat, "CYC: A large-scale investment in knowledge infrastructure," *Commun. ACM*, vol. 38, no. 11, pp. 32–38, 1995.

[47] R. Speer and C. Havasi, "Representing general relational knowledge in ConceptNet 5," in *Proc. 8th Int. Conf. Lang. Resources Eval.*, 2012, pp. 3679–3686.

[48] K. Narisawa, Y. Watanabe, J. Mizuno, N. Okazaki, and K. Inui, "Is a 204 cm man tall or small ? Acquisition of numerical common sense from the web," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, 2013, pp. 382–391.

[49] G. Collell, L. V. Gool, and M. Moens, "Acquiring common sense spatial knowledge through implicit spatial templates," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 6765–6772.

[50] F. F. Xu, B. Y. Lin, and K. Q. Zhu, "Automatic extraction of commonsense LocatedNear knowledge," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 96–101.

[51] M. Pasca, "Interpreting compound noun phrases using web search queries," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2015, pp. 335–344.

[52] S. Cordeiro, C. Ramisch, M. Idiart, and A. Villavicencio, "Predicting the compositionality of nominal compounds: Giving word embeddings a hard time," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 1986–1997.

[53] B. Alidaee, F. Glover, G. A. Kochenberger, and H. Wang, "Solving the maximum edge weight clique problem via unconstrained quadratic programming," *Eur. J. Oper. Res.*, vol. 181, no. 2, pp. 592–597, 2007.

[54] X. Qiu, Q. Zhang, and X. Huang, "FudanNLP: A toolkit for chinese natural language processing," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, 2013, pp. 49–54.

[55] D. Zhang, J. Yuan, X. Wang, and A. Foster, "Probabilistic verb selection for data-to-text generation," *Trans. Assoc. Comput. Linguistics*, vol. 6, pp. 511–527, 2018.

[56] C. Zhai and J. D. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," *ACM SIGIR Forum*, vol. 51, no. 2, pp. 268–276, 2017.

[57] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. 1st Int. Conf. Learn. Representations*, 2013.

[58] L. B. Soares, N. FitzGerald, J. Ling, and T. Kwiatkowski, "Matching the blanks: Distributional similarity for relation learning," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 2895–2905.

[59] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 4171–4186.

[60] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2012, pp. 481–492.

Chengyu Wang received the BE degree in software engineering from East China Normal University, in 2015. He is currently working toward the PhD degree in the School of Software Engineering, East China Normal University (ECNU), China. His research interests include Web data mining, information extraction, and natural language processing. He is working on the construction and application of large-scale knowledge graphs.

Xiaofeng He received the PhD degree from Pennsylvania State University, Pennsylvania. He is a professor of computer science at the School of Computer Science and Technology, East China Normal University, China. His research interests include machine learning, data mining, and information retrieval. Prior to joining ECNU, he worked at Microsoft, Yahoo Labs, and Lawrence Berkeley National Laboratory. He is a member of the IEEE.

Aoying Zhou is a professor with East China Normal University. He is now acting as a vice director of the ACM SIGMOD China and Database Technology Committee of the China Computer Federation. He is serving as a member of the editorial boards of the *VLDB Journal*, *WWW Journal*, etc. His research interests include data management for data-intensive computing, and memory cluster computing. He is a member of the IEEE.