Exploring open information via event network

Chen, Y, Zheng, Q, Tian, F, Liu, H, Hao, Y & Shah, N

Original citation & hyperlink:
Chen, Y, Zheng, Q, Tian, F, Liu, H, Hao, Y & Shah, N 2017, 'Exploring open information via event network' *Natural Language Engineering*, vol 24, no. 2, pp. 119-220
https://dx.doi.org/10.1017/S1351324917000390

DOI 10.1017/S1351324917000390
ISSN 1351-3249
ESSN 1469-8110

Publisher: Cambridge University Press

Exploring Open Information via Event Network

Yanping Chen*†, Qinghua Zheng†, Feng Tian†, Huan Liu†, Yazhou Hao†, Nazaraf Shah‡

E-mail: ypench@gmail.com; {qhzheng, fengtian}@mail.xjtu.edu.cn; {hliuxjtu, yazhouhao}@gmail.com; aa0699@coventry.ac.uk;

∗ *GuiZhou University, Guizhou Provincial Key Laboratory of Public Big Data, Guiyang, China*

† *Xi'an Jiaotong University, Shaanxi Province Key Lab. of Satellite and Terrestrial Network Tech. R&D, Xi'an, China*

‡ *Coventry University, the Faculty of Engineering and Computing, United Kingdom*

Abstract

It is a challenging task to discover information from a large amount of data in an open domain*. In this paper, an event network framework is proposed to address this challenge. It is in fact an empirical construct for exploring open information, composed of three steps: document event detection, event network construction and event network analysis. First, documents are clustered into document events for reducing the impact of noisy and heterogeneous resources. Secondly, linguistic units (e.g., named entities or entity relations) are extracted from each document event and combined into an event network, which enables content-oriented retrieval. Then, in the final step, techniques an be developed to analyze the event network for exploring open information.

1 Introduction

Exploring information in an open domain is a challenging task due to large volume of data involved (Brin 1998; McCallum 2005). Gigantic volumes of data spring up all over the world on a daily basis. These information is easily available to us simply by connecting to the Internet. Effectively managing this huge volume of information is beneficial for the decision-makings due to availability of more relevant information. For example, if we want to understand a person, we may mine his social relationships from relevant data. We can track the trajectory of a person by examining the correlated locations and timestamps. To explore information from an open domain, event-oriented techniques provide effective way to understand what has happened on the Internet. In information science, Information Retrieval (IR) and Information Extraction (IE) provide event-oriented techniques to cope with the flood of information.

In traditional IR systems, an event detection system outputs clustered documents, where a document cluster is considered to describe the same event. In this paper, the clustered document is referred to as a document event. The problem with IR systems is that they retrieve documents mainly based on the occurrence of terms, but the contents of documents

∗ The open domain means that the process of exploring information is domain-independent or implemented on a large body of text (Brin 1998).

cannot be identified. In an open domain, a query can return millions of documents. Even these documents are ranked by their relevancy to a query, but people still need to skim through the contents for relevant information, which results in the "Information Overload" problem. Another important issue is that, sometimes people have no knowledge of the events occurred recently. The query-based systems are not effective for finding or tracking previously unknown events.

An event in IE is defined as a template with slots to be filled, referred as a template event in this paper. Recognizing the template event is a task of extracting structured data (e.g., linguistic units) from semi-structured or unstructured data. It is expected that the extracted data can be used to populate a knowledge base directly. The main issue with an IE system is that it focuses on extracting linguistic units from the sentence level context. The context often consists of a limited number of words, which leads to sparse feature problem. In an open domain, heterogeneous, noisy and fragmental data exist, which further worsen the performance.

In this paper, an event network framework is proposed for exploring open information. The contribution of this paper includes,

1. An event network is proposed for exploring open information. It is an empirical construct for exploring open information, which can be divided into three steps: document event detection, event network construction and event network analysis.
2. To analyze event network, four analytic methods are proposed for exploring open information via event network.

The rest of this paper is organized as follows. Section 2 introduces related work. Motivations and some related definitions are presented in Section 3 and Section 4. Our implementation of the event network to explore open information is presented in Section 5. Finally Section 6 provides the conclusion of the paper.

2 Related Work

A number of systems have been developed to organize linguistic units as networks in the field of natural language processing (NLP). The related work in this area can be broadly divided into three paradigms: logic based semantic network, scalable knowledge database and semantic network constructed from open information extraction.

Logic based semantic network refers to networks constructed manually, e.g., WordNet (Miller 1995), and Cyc (Lenat 1995), etc. They mainly focus on a closed domain, for example, a "conceptual graph" represents logic as a graph representation (Sowa 1984). It supports logical operators directly and can map questions and assertions from natural language to a relational database. Logic based semantic networks are constructed by domain experts and they are used as domain-specific ontologies. These networks are mainly designed to support logical reasoning, conflicts and contradictions are not allowed. Therefore, it is difficult to construct them from an open domain.

In the second paradigm, logic based semantic network is extended to an open domain. Large knowledge databases such as Yago (Suchanek, Kasneci and Weikum 2007), Freebase (Bollacker et al. 2008) and DBpedia (Auer et al. 2007) are constructed, representing large knowledge in formalized forms. These representations merge diverse and heterogeneous

data with high scalability, providing a unified framework for organizing information. They have no canonical view of data, and a loose representation is used to support scalability and extensibility. Most of these databases are constructed by merging ontologies (e.g. WordNet, Cyc) or extracted from semi-structured database (e.g. Wikipedia). Many of these databases are constructed by using semi-supervised methods. There are research efforts for building these systems suing the state-of-the-art Natural Language Processing technologies (e.g., Xlike (Padró et al. 2014), ECKGs (Rospocher et al. 2016)).

Instead of aiming at semi-structured data, the third paradigm explores information in an open and dynamic domain. In this paradigm, weak supervision (Mintz et al. 2009; Liu et al. 2014) and bootstrapping methods (Kozareva and Hovy 2010; McIntosh et al. 2011; Weld, Hoffmann and Wu 2009; Agichtein and Gravano 2000) are employed. Scalable knowledge databases are used to guide the process, e.g., TextRunner (Banko et al. 2007), KnowItAll (Etzioni, et al. 2005; Etzioni et al. 2011), WOE (Hoffmann, Zhang and Weld 2010) and StatSnowBall (Zhu et al. 2009). There are systems focus on issues about: cross-lingual extraction (Zhang et al. 2017), integrating syntactic and semantic analysis (Zhang et al. 2017), Leveraging linguistic structure (Angeli et al. 2015), etc. In these systems nodes are named entities and edges are relationships between them. All extracted results are combined into a large network.

In the related work, the notion of event is widely used to explore information. Piskorski et al. (2011) present an on-line news event extraction system. Each event is defined as a template with slots filled with information extracted from clustered documents. Ramakrishnan et al. (2014) propose EMBERS system encoding events as templates. The system used to forecast “civil unrest” events in an open domain. TwiCal extracts open-domain events from Twitter (Ritter, Etzioni and Clark 2012), where events are identified by named entities. In the same data set, ET represents the events as clusters of keywords (Parikh and Karlapalem 2013). Kuzey et al. (2014) uses events itself as nodes of the network. They cluster documents into a hierarchical representation, where nodes are events and edges link the same event in chronological order. Angel et al. (2012) construct an entity network from social media by using “streaming edge weight” method. They mine dense subgraphs of network for identifying real-time stories. Angel et al. (2011) provide an event discovery method based on entity dynamic relation graphs, which are constructed by co-occurrences of entities in a document. NewsReader is a system that dynamically processes massive amount of news articles (Vossen et al. 2016), in which where, who, what, where, and when are extracted. EVIN defines a named event as an entity that happened at a certain time point or during a certain time period (Kuzey and Weikum 2014). EVIN extracts named events from news articles and provides a GUI that supports users to explore events in a visual manner.

The techniques used to construct these systems can be roughly divided into three categories: unsupervised, semi-supervised and supervised. Unsupervised method uses techniques such as clustering (Wang et al. 2011; Downey et al. 2007) or heuristic rules (Xu et al. 2013) to find information automatically. Unsupervised method shows an interesting perspective on open information extraction, but the output of unsupervised method is difficult to predict, and usually suffer from a worse performance. Currently, semi-supervised method is an active topic in information extraction research and various techniques were proposed, e.g., distant supervision (Mintz et al. 2009; Liu et al. 2014; Takamatsu et al. 2012), boot-

strapping algorithm (Kozareva and Hovy 2010; McIntosh et al. 2011; Weld, Hoffmann and Wu 2009) and matrix factorization (Riedel et al. 2013). Systems developed by using semi-supervised method may suffer from "semantic drift" problem, caused by error accumulation (Curran et al. 2007). In supervised, techniques, e.g., kernel method (Zhang et al. 2006; Zelenko et al. 2003; Collins and Duffy 2001), belief network (Roth and Yih 2002), maximum entropy (Kambhatla 2004) and SVM (Zhou et al. 2005) have been widely discussed. Systems developed by applying supervised approach are often trained on the manually annotated corpus and, they can achieve robust performance. However, the scalability is limited to the requirement of annotated corpus. In an open environment, heterogeneous sources are encountered , which worsens the performance of supervised method.

In this domain, many systems may generate hundreds or thousands of linguistic units dynamically. All the extracted units are combined into a complex network, and less effort is required to analyze the underlying structure of the linguistic units. Because in an open domain noisy and heterogeneous sources are often get processed, this can easily lead to a redundant and incompatible output. A chaotic result provides little help for human oriented analysis. In our work, before constructing the network, we implement the event detection and event tracking, which is useful for filtering irrelevant events or/and tracking interested events. In addition to constructing the network, several methods have been discussed to support the event network analysis. These methods have shown promising result for exploring open information.

3 Motivation

Organizing linguistic units into networks or graphs has attracted increased research interest in the NLP research area. It provides novel solutions for many NLP tasks and supports human-oriented information exploration. A network or graph representation also enables topology-based analysis developed in fields such as: social networks and complex networks.

There are automatically constructed knowledge bases to support information exploration. Many of them are constructed from semi-structured database (e.g. Wikipedia, WordNet) or with human labour (collaboratively). Therefore, a better data consistency is expected from them. They are widely used as external knowledge sources for semi-supervised methods. As semi-structured databases are often required, when a news is spreading in an open domain, automatically handling this information in real time is a difficult task. Furthermore, many systems organize extracted linguistic units into a graph-based representation. These systems generally lead to a complex network with thousands of nodes and edges (e.g., Hoffmann et al. (2010)). The information extraction systems may lead to worsen the performance. The redundant and incompatible information makes the result difficult to understand. Many systems focus on the construction of network, rare analysis was conducted to show the underlying structures of linguistic units.

In the event network, we emphasize on the analytical approaches for event network analysis. Several approaches are discussed to explore these networks. There are systems that provide linguistic processing pipelines to build semantic networks for exploring open information (e.g., Xlike (Padró et al. 2014), ECKGs (Rospocher et al. 2016)). Our event network is an empirical construct to explore open information. Instead of having a monolithic

method, the process for exploring open information is divided into three steps: document event detection, event network construction and event network analysis.

In the first step documents are clustered into document events by implementing document event detection and tracking. In this step, most of the irrelevant information is filtered out. Then, in the second step, IE techniques are used to extract linguistic units, and the result is organized into event network. In the last step, structural information between linguistic units can be used to improve the quality of constructed event networks, for example, entity disambiguation or co-reference resolution. Then, network or graph based analytic methods can be used to analyze each event. It can also be visualized to support human-oriented information exploration. By dividing the process into three steps, it is convenient to use techniques developed in information retrieval, information extraction and semantic network.

Advantages of the event network include: Firstly, after document event detection the linguistic units in each event can be independently extracted. Therefore, the impact of noise and heterogeneous data on information extraction can be reduced. Secondly, as the event network is constructed from a cluster of documents, this information enables the discovery of potential relationships between documents. Thirdly, event network provides a structured data representation, where topological information is available. Then, many topological methods (e.g., social network or complex network) can be introduced for analyzing each event.

4 Event Network

For the convenience of the discussion of the event network analysis, in this section we give definitions related to event network.

Let $\mathscr{D}$=$\{d_1, \cdots, d_L\}$ be a document set, where d_i denotes a document. A document event $\mathscr{E}_k$ is a subset of $\mathscr{D}$. For all $d_i, d_j \in \mathscr{E}_k$, the similarity between them $Similarity(d_i, d_j)$ satisfies a predefined condition (e.g. a threshold). All document events in $\mathscr{D}$ are denoted as $\mathscr{E}$=$\{\mathscr{E}_1, \cdots, \mathscr{E}_K\}$. The constraint that $\mathscr{E}$ is a partition of $\mathscr{D}$ is not necessary, because some documents in $\mathscr{D}$ can be filtered, or "fuzzy partitioning techniques" can be applied, which enables identification of a document belonging to more than one document event.

An event network on document event $\mathscr{E}_k$ is represented as a graph $\mathscr{N}_k$=$\{V_k, E_k\}$, where V_k=$\{v_{k1}, \cdots, v_{kN}\}$ and E_k=$\{e_{k1}, \cdots, e_{kM}\}$ are vertex set and edge set. Both vertices and edges are defined as templates shown as follows.

$$vertex := \{key, name, type, weight, info\}$$
$$edge := \{type, v\text{-}1, v\text{-}2, weight, info\}$$

Where *vertex* template defines nodes of event network. Slot "$name$" refers to the entity that exists in a document event. Each vertex is identified by an integer value "key" . Slot "$type$" represents categories of vertices (e.g. "Person", "Organization" and "Location"). Slot "$weight$" is used to weigh a template (e.g., the likelihood of "$type$"). This value can be given by a classifier when extracting the template. Depending on real applications, "$weight$" is used to filter an event network. An *edge* template denotes

a relation between two vertexes. Slots "v-1" and "v-2" are $keys$ of vertices in an edge, used to identify vertices linked by edges. Edge types are referred by "$type$" (e.g. Part-whole, Personal-Social). In the defined templates, slot "$info$" contains information about the templates, e.g., sentences, documents or timestamps. These information support event network analysis (e.g. co-reference resolution, statistical relational learning).

We define "nodes" and "edges" of event network as "templates" with "slots" . This definition is expected to support event network analysis. All edges in an event network are referred by a $N \times N$ matrix as $\mathbf{M}_{\mathcal{N}}$. If $\mathbf{M}_{\mathcal{N}}(i,j) \neq null$, then a relation exists between v_i and v_j. Relations between document events can be defined too, referred by a $K \times K$ matrix as $\mathbf{M}_{\mathcal{D}}(i,j)$. Various event relations can be defined. For example, hierarchical clustering methods organize document events into a tree structure. Timestamps segment document events, and same document events are linked to show the dynamic changes between them.

An example of event network is given in Figure 1.

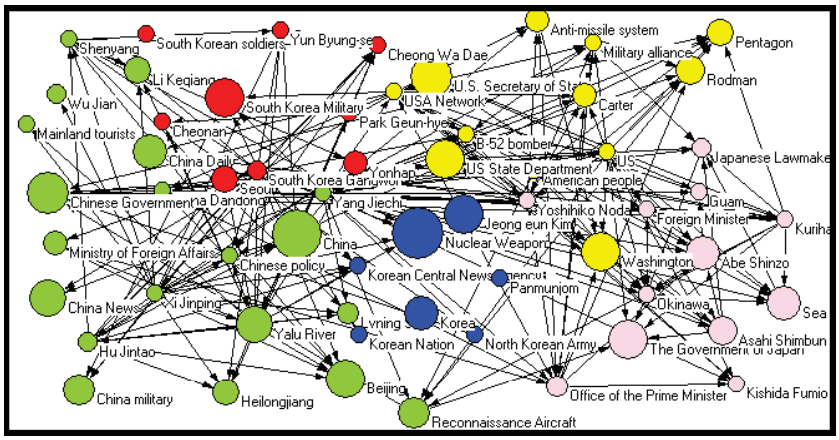

Fig. 1: Event Network

This event is about the "the Korean nuclear issue" . Nodes are named entities (e.g., persons, locations or organizations) and, edges represent the relationship between them. The color of nodes identifies the country names, where green, yellow, red and blue represent China, U.S., South Korea and North Korean respectively. The information about nationality of the entities can be given by a thesaurus. The size of nodes is set by using the $weight$. It is helpful to support human-oriented information exploration.

5 Methodology and Experiments

This section first discusses our proposed method for constructing event network. Then, four analytical approaches are discussed to show the flexibility of the event network for exploring open information.

In order to train classifiers to extract linguist units, we use the ACE 2005 Chinese corpus

†. It contains 633 documents annotated with 15,264 entities and 33,932 entity mentions ‡. Seven entity types (e.g. “person”, “organization”, etc.) and 44 entity subtypes are defined. The corpus is also annotated with 6 major relation types and 18 relation subtypes. Each relation instance has two named entities as arguments. There are 9,244 collected relation mentions§.

In order to show our method’s applicability in an open domain, we also use the Chinese Gigaword Fifth Edition corpus ¶. The Peoples Daily source is used, which contains 145,001 newswire texts covering the period from November 2006 to December 2010.

5.1 Document Event Detection

The purpose of document event detection is to cluster documents into events. Traditional, methods of event detection are mainly based on Vector Space Model (VSM), which represents documents as vectors with fixed length dimension. VSM maps a document space into a term space. Elements of the vectors correspond to the weighting of the terms in a document, e.g., Term Frequency (TF) (Luhn 1957) or Inverse Document Frequency (IDF) (Jones 1972). Documents are represented as dots scattered in a term space. Similarities between documents are computed by a measure function (e.g., Cosine or Manhattan distance). The Latent Dirichlet Allocation (LDA) model assumes a topic space (Blei et al. 2003). The topics are modelled by hidden variables, and can be defined as a distribution over a fixed vocabulary. Then, the term space is mapped into a topic space. All documents in the term space can be represented by vectors in the topic space. It is expected that the topic space can capture some semantic information (Deerwester et al. 1990). One output of LDA is the distribution of topics in the term space. A topic would have “similar” words about this topic with high probability. The idea is similar of some related work published in the Topic Detection and Tracking (TDT) evaluation (Trieschnigg and Kraaij 2004; Nallapati et al. 2004; Allan et al. 1998), which emphasizes on detection of streams of data that are topically related material in real time (e.g., newswire and broadcast news). Therefore, we use a cluster of similar documents to identify an event. In future work, other methods developed in the natural language community can be used to identify an event.

We use LDA toolkit provided by Phan et al. (2007) to implement this task. In LDA model, a corpus is first represented as a matrix, where each column refers to a document vector, and each row represents distribution of a term in the documents. Then LDA maps documents from a term space into a topic space, where topics are hidden variables.

As we focus on newswire texts, where short texts are commonly used, we use Omni-word feature proposed by Chen et al. (2014). The Omni-word feature is a subset of n-gram features filtered by a lexicon. In the pre-processing, we remove high and low frequency words ‖. Words with frequencies lower than 10 are also omitted. To train an LDA model, hyper-parameters are required. Because the Peoples Daily texts are processed, we set the number of topics according the number of columns (e.g., “military”, “sport” and

† `https://catalog.ldc.upenn.edu/LDC2006T06`
‡ An “entity mention” is a reference to an entity.
§ A relation mention is an occurrence of a relation.
¶ `https://catalog.ldc.upenn.edu/LDC2011T11`
‖ The ratio is 5% for each.

“fashion”, etc.) in a traditional news website, where it often contains 20∼30 columns. The topic number is set as 25. Other parameters use default settings.

The toolkit generates several outputs. The word-topic distributions are more favourable to us, as it gives distributions of terms in a topic space. We use topics as centroids of document clusters in a term space. This setting is helpful for tracking events and reducing the computing complexity. The idea of using centroid vector for event detection also discussed by Yang et al. (1999). When clustering documents, documents belonging to an event are identified by the nearest “Euler Distance” of the document and centroids. Because LDA is an unsupervised method, the type of events cannot be output by the LDA model directly. In order to label an event, we use the top words of topics outputted by the LDA model. The top 100 words of each topic are used to represent an event.

It is recognized that documents discussing the same event tend to be in a short period of time, and the time gap between bursts of similar documents may indicate different events (Yang et al. 1999). Therefore, timestamps are used to partition the newswire texts. In our experiment, the time step is set as 5 months. Then, the Chinese Gigaword corpus is divided into 10 parts. Each part contains 5 months of newswire texts. Hierarchical representation can give a multi-granularity view when exploring open information and reduces the travel cost. In each time step, instead of using retrospective methods to give a flat partition of documents, we organize them into a hierarchical representation. Documents in each time step are clustered into 25 events by LDA toolkit. Each event is further clustered into sub-events using the same approach. If an event contains less than ten documents , the process to find its sub-events is skipped. Therefore, in each time step, 25 events and at most 25×25 sub-events are detected.

In addition to detect events, we also implement event tracking. The difference between event tracking and detecting is that, in addition to detect an event, event tracking should link the detected event to previous event if they describe the same event. In this field, the methods for tracking event can be roughly divided into two types: unsupervised tracking and supervised tracking (Zhang and Callan 2004; Trieschnigg and Kraaij 2004). In our system, an event is linked to the nearest event in the previous time step. We compute the event distances in adjacent time steps by using cosine distances. Our event tracking is given in Table 1.

In Table 1, $\mathscr{E}_1 \sim \mathscr{E}_5$ and 0∼9 represent events and time steps respectively. Documents in each time step are clustered into 25 events independently (identified from 0∼24). Because LDA is an unsupervised method, the similar event may have different ID in different time steps. In each row of Table 1, the numbers below time steps represents the identifications of an event in different time steps. In each time step, an event is linked to the nearest event in the previous time step. For example, for $\mathscr{E}_1$ in Step 0, we compute the distances between $\mathscr{E}_1$ to all events (0∼24) in Step 1. Then, $\mathscr{E}_1$ is linked to the nearest event in Step 1 (e.g., 12 in Step 1). Top words in the event are listed in the second column.

85667045

5.2 Event Network Construction

To construct each event network from the detected document events, we extract named entities and the relationships between them as nodes and edges of the event network.

Table 1: Event Tracking

Event	Top words in events	Time Step 0	1	2	3	4	5	6	7	8	9
$\mathscr{E}_1$ (Family)	老人, 父亲, 母亲[a]	11	12	9	9	21	6	7	7	11	1
$\mathscr{E}_2$ (Law)	民法, 司法, 审判[b]	12	7	7	7	7	19	8	5	5	5
$\mathscr{E}_3$ (ENT)	明星, 篮板, 联盟[c]	17	20	12	18	8	8	19	19	16	3
$\mathscr{E}_4$ (Environ.)	生态, 森林, 气象[d]	20	9	14	2	18	3	6	0	19	15
$\mathscr{E}_5$ (Terrorism)	袭击, 北约, 威胁[e]	24	11	11	13	9	16	3	8	4	6

Translated as: (a) “The aged”, “Father” and “Mother”. (b) “Civil law”, “Judicature” and “Adjudgement”. (c) “Star”, “Backboard” and “League” (d) “Ecology”, “Forest” and “Atmosphere”. (e) “Assault”, “NATO” and “Threaten”.

5.2.1 Named Entity Recognition

The task to recognize named entity is modeled as a label tagging process. If labels of tagging units are supposed to be independent and identically distributed), a likelihood function can be used to evaluate the distribution of tagging units independently, such as Maximum Entropy (Chen et al. 2006), SVM (Hacioglu et al. 2005), TBL (Zhou et al. 2005) or Deep Belief Nets (Chen et al. 2010). If labels of tagging units are dependent, the task can be modeled as a sequence analysis. Then, sequence models (e.g. HMM (Fu and Luke 2005; Carpenter 2006), CRF (Chen et al. 2006), SVMs chunkier (Ling et al. 2003) or neural architecture (Chiu and Nichols 2015; Lample et al. 2016)) can be adopted to find an optimized label sequence. Currently, the named entity recognition is often implemented on Wikipedia (Nothman et al. 2013), tweets (Derczynski et al. 2015), biomedical documents (Tang et al. 2014). etc.

In our previous work (Chen, Zheng and Chen 2015), a Boundary Assembling (BA) method is proposed to implement the named entity recognition task. In the BA method, the beginning and last boundaries of named entities are first detected, then assembled into named entity candidates. Each candidate is further assessed by a classifier, where more features are available to distinguish them. Compared to traditional methods, the BA method has four advantages. First, tagging units to be a boundary is unambiguous, which ensures that it can be labeled precisely and detected effectively. Second, the boundary is the minimal granularity of a sentence. Recognizing it won’t depend on other analytical processes. Third, a cascading model can benefit from the flexibility that features can be used in different ways. Finally, this method is effective for recognizing nested named entities and making better use of non-local features. Based on the ACE 2005 Chinese corpus, the BA method was compared with four nested named entity recognition methods discussed in Alex et al. (2007). It outperforms existing methods in F-score by 5% in named entity recognition.

In our work, we recognize three types of named entity: “PER” (Person), “LOC”

(Location) and “ORG” (Organization). In order to filter noise, recognized named entities with Chinese characters less than two and more than six are discarded. The impact of removing sentences with too many named entities can be reduced in an open domain, because the system often benefits from the adequateness of information (Brin 1998). As shown in Figure 2, the total number of sentences in the employed corpus is 6,888,616. Only a small part (0.368884%) contains 10 or more named entities.

5.2.2 Relation Recognition

The goal of relation extraction is to detect relations between two entities from free text. Currently relation extraction is often formulated as a classification problem, where a relation takes two entities as arguments. There are two paradigms to extract relationships between entities: Open Relation Extraction (ORE) and Traditional Relation Extraction (TRE) (Banko et al. 2008). ORE often uses external sources to guide the extracting process, e.g., Freebase (Mintz et al. 2009), Patterns (Moro et al. 2013), ontology (Mohamed et al. 2011) or integrated analysis (Moro and Navigli 2013). In TRE, supervised techniques are commonly used, e.g., kernel method (Sun and Han 2014), linear programming (Roth and Yih 2007), maximum entropy (Kambhatla 2004) or deep neural network (Zeng et al. 2014).

The major challenge for relation extraction is that most relation instances occur within a sentence or clause, and usually consist of just a few words, which often leads to a sparse feature representation. To recognize relations between named entities, we adopt the method proposed in our previous work (Chen, Zheng and Zhang 2014; Chen, Zheng and Chen 2015), in which an Omni-word feature and a feature assembly method are proposed for Chinese text relation extraction.

The Chinese language has a distinct word-formation method, where the meaning of a Chinese compound word comes from the meanings of words in it. Therefore, in Chinese, fragments of a phrase are also informative. The Omni-word feature uses every potential word in a sentence as lexical features. Unlike the traditional segmentation based lexical features, which are a partition of a sentence, many Omni-word features are overlapped or nested with each other in the same sentence. Because most sentences have limited context, Omni-word features, utilizing every possible word in a sentence, is a better way to capture the sentential information.

In the feature assembly method, we make a distinction between atomic features and combined features. The atomic features refer to the traditional ``finer-grained'' features used in NLP, which are not dividable. For each employed atomic feature, an appropriate constraint condition is selected to combine them with additional information. It outputs combined features, which maps the feature space into a higher dimensional space and leads to a flexible decision boundary. The feature assembly method provides a flexible framework for capturing the local dependency of relation instances and making better utilization of atomic features. Therefore, it is helpful to reduce the problem caused by sparse feature in relation recognition.

The method was tested by using the ACE 2005 Chinese and English corpora, it was compared with three existing methods: Zhang et al.(2011), Che et al. (2005) and Kambhatla (2004)). It outperforms the compared methods about 20% in F-score respectively.

In our event network construction process, among three named entity types, five entity

relation types are recognized: “PER-SOC”, “GEN-AFF”, “ORG-AFF”, “PART-WHOLE” and “PHYS”, Corresponding to “Personal-Social”, “General-Affiliation”, “Organization-Affiliation”, “Part-Whole” and “Physical”. Sentences with more than ten entities are ignored, because extracting relations in a long sentence is error-prone.

Another problem for event network construction is co-reference resolution. It is the task to group different entity mentions according to whether they referred to the same entity or not. This task has been defined and evaluated as a separate subtask of information extraction since MUC-6 [**]. It also suffers from the problem of sparse feature the same as relation extraction. Furthermore, the co-reference resolution partitions all entity mention into mutually exclusive sets, where every mention pair belongs to the same set is coreferent. Therefore, the transitivity in co-reference resolution is important, which result in a large search space and an unbalanced dataset. In our current work, the problem of co-reference resolution is not considered. Instead, we aggregate entities with the same name into to a single node of event network. In our future work, the emphasis will be placed on this issue.

5.2.3 Merging and Visualizing

We have implemented the discussed methods on the Chinese Gigaword Fifth Edition corpus for detecting document events, recognizing named entities and relations respectively. The result is listed in Table 2.

Table 2: Information of Constructed Event Networks

Time Step	Doc.	Event	Entity	Relation
0	14,814	592	1,123,506	309,847
1	11,734	567	897,489	258,817
2	17,678	619	1,104,517	257,291
3	18,213	646	1,305,191	277,721
4	23,433	644	1,704,294	289,166
5	19,706	643	1,389,668	290,594
6	12,014	545	927,977	265,439
7	9,100	467	705,805	254,133
8	9,326	470	726,123	230,535
9	8,983	420	674,457	243,210

In the following, we conduct event network analysis by using igraph toolkit (Csardi and Nepusz 2006) to represent extracted entity mentions and relations as a graph. A network analysis Pajek toolkit (Batagelj and Mrvar 1998) is used for visualization.

5.3 Event Network Analysis

In our work, we emphasize on the analysis of event networks. After event networks were constructed, techniques such as social network, complex networks can be employed

[**] http://www-nlpir.nist.gov/relatedprojects/muc/

to analyze event networks. For example, setting a person name as a central entity, we can navigate entities around it. By collecting the person names (“PER”) and the social relationships (“PER-SOC”), we can show character relationships in an event. Using the “PART-WHOLE” relations, multi-granularity visualization can be supported. As event network has topological information about a document event, various topology-based approaches (e.g., statistical relational learning) can be used to improve the network quality. Furthermore, event networks support human-oriented analysis. When human explores open information, manual interventions can be used to modify the quality of event networks.

In this section, we choose a document event in time step 0 as an example, which contain 1,041 documents. There are 42,436 named entities and 6,272 relations. The most likely words in this document event are “袭击, 北约, 发言人, 冲突, 防御, etc.” (Assault, NATO, Spokesman, Conflict, Defence, etc.). It indicates that the domain of this event is military affairs. Extracted named entities and relations are organized in Figure 2, where there are 252 nodes and 571 edges. Nodes in “Red”, “Yellow” and “Blue” colors represent “Person”, “Organization” and “Location” respectively. Each edge is labelled with a relation type.

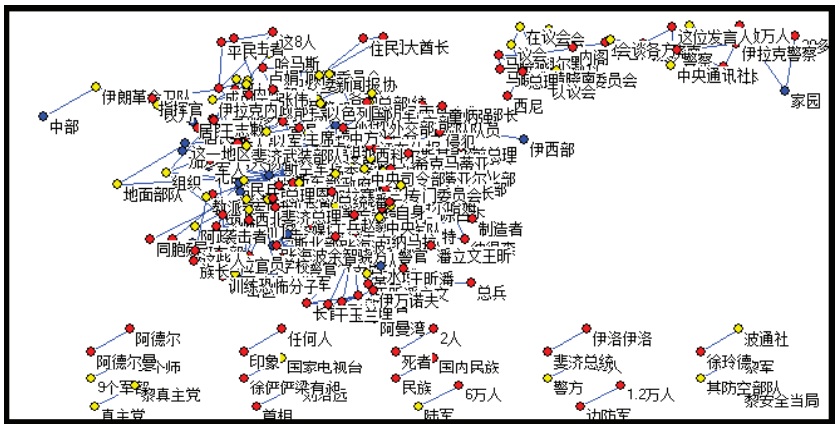

Fig. 2: Event Network

To explore open information, many systems dynamically organize linguistic units into a complex network. Heterogeneous resources and unreliable information make the network chaotic and misunderstanding. Figure 1 shows complex network which is difficult to understand. In the following, based on the event network, we give four methodologies to explore open information: Information Filtering, PLT Analysis, Action Analysis and Social Network Analysis.

5.3.1 Information Filtering

The simplest way to analyze event network is to filter our information that is irrelevant. In Figure 3, only person names and “PER-SOC” relations are left to show character relationships in an event network.

This example can be formalized as: Let $\mathcal{N}$ be an event network. The filtered event

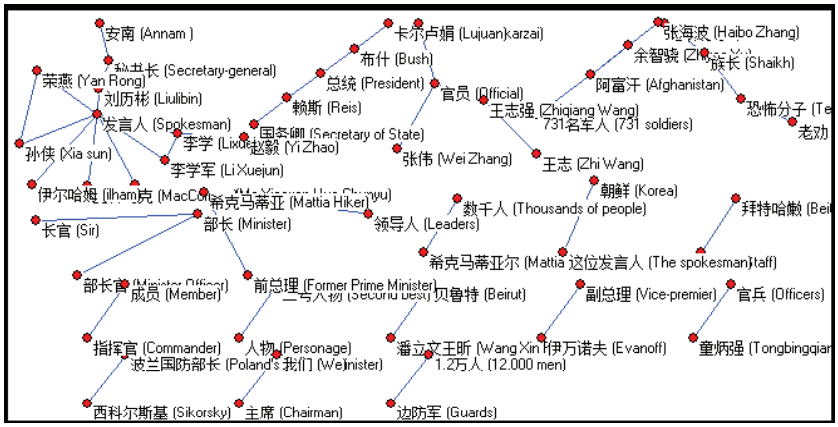

Fig. 3: Information Filtering

network $\mathscr{N}'$=$\{V', E'\}$ is a subgraph of $\mathscr{N}$, such that $\mathscr{N}' \subset \mathscr{N}$. And $\mathscr{N}'$ satisfies $\forall v \in V'(v.type = \text{PER}) \wedge \forall e \in E'(e.type = \text{PER-SOC})$.

Using information contained in $vertex$ templates and $edge$ templates, information filtering can provide effective approaches to explore open information. For example, in $vertex$ templates and $edge$ templates, we may require that the value in $weight$ slots is greater than a predefined threshold. Utilizing information in $info$ slots, we can collect named entities occurred in a specified period or area. Setting a person name as a central node, we can directly see his social relations.

5.3.2 PLT Analysis

Person-Location-Time (PLT) analysis tries to find relations between persons and locations in a period of time. It can be used to track a person, find trajectories of targeted entities. In order to implement the PLT analysis, the person name, location name and time should be recognized in advance. Techniques to extract the person name and location name are discussed above. The issue that needs to be addressed is how to get the time information.

Two kinds of time are distinguished in a document in the form of “implicit temporal information” and “explicit temporal information”. Implicit temporal information is a part of the document’s content indicating its creation, development and termination of an event. Extracting this information needs information extraction or text understanding techniques. In many applications, it is generally ignored. Generally in an open domain, all documents have explicit temporal information, which includes the creation, modification and transmission timestamps of documents. They are meta-data associated with documents. This paper focuses on newswire texts, where explicit temporal information of documents are released together with the documents. Therefore, we use the explicit temporal information for PLT analysis.

This process can be formalized by introducing an attribute $time$ in the $info$ slot. Let $\mathscr{N}$ be an event network, $time$ represents timestamps and $person$ presents a person name. $\mathscr{N}'$=$\{V', E'\}$ is the result of PLT analysis based on $\mathscr{N}$, where $\forall e \in E'(e.type = \text{PHYS} \wedge (e.v\text{-}1 = person \vee e.v\text{-}2 = person))$. In other words, all relation types in E'

is “PHYS”, and takes the same entity mention $person$ as an argument. Replacing all $person$ by corresponding $time$, we get a graph containing timestamps and locations as nodes. An example is shown in Figure 4.

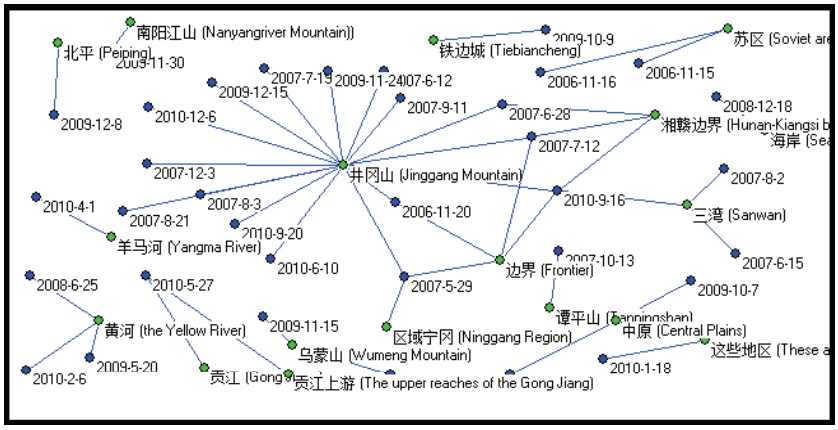

Fig. 4: PLA Analysis

In this example, we track Mao Zedong (“毛泽东”††) in the whole Gigaword corpus, collect all recognized “PHYS” relation instances which have Mao Zedong as an argument. As a result, there are 142 “PHYS” relation mentions, which take Mao Zedong (or Chairman Mao) as an argument. Then we replace Mao Zedong (or Chairman Mao) by the explicit temporal information of newswire texts. In Figure 4, nodes in green color are timestamps, and blue nodes are locations ‡‡. Each green node means that Mao Zedong occurred with the connected locations at that time. The times associated with activities of Mao Zedong mean that they are mentioned in the tracked newswire texts. The result is useful for mining public opinions, and is helpful to find important events or sensitive events.

5.3.3 Action Analysis

Recognizing an “event” under the ACE definition is difficult, where event triggers, participant roles, properties and attributes should be identified (Doddington et al. 2004). Overall, using the best learned classifiers for the various subtasks, the reported performance only achieved an ACE value score of 22.3% (Ahn 2006). In our application, we present the action analysis, where the co-occurrence information among named entities is used to understand the underlying structure of the documents. Co-occurred named entities can show potential relationships between them.

In action analysis, we focus on detecting whether or not a special action is mentioned in a sentence. Therefore, we conduct the “sentence classification” task, classifying each sentence by a classifier trained on the ACE annotated event mentions. In our application,

†† The leader of the Communist Party of Chinese.
‡‡ Because the original graph is more complex (55 nodes and 142 edges), here only a part of it is given.

we monitor the “Conflict” as ACE event type, which has two subtypes: Attack and Demonstrate (Doddington et al. 2004). The ACE corpus, which annotates 596 “Conflict” events is employed for training and testing. We have implemented the 5-fold cross validation, and the P/R/F (Precision/Recall/F-score) measurement. F-score is computed by $(2 \times P \times R)/(P + R)$. In order to perform a two-class classification, we generate negative instances by segmenting the corpus into sentences, discarding annotated ACE event mentions, and filtering sentences without event triggers of “Conflict” ACE events. Then, 1,589 sentences are collected as negative instances. We only use Omni-words features in sentences for classification. The performance is shown in row 1 of Table 3, where the performance concerning “Conflict” is listed.

Table 3: Performance of Action Analysis

No.	Precision	Recall	F-core
1	82.19	82.88	82.53
2	97.65	41.94	58.68

In an open domain with massive data, high precision is desirable. Therefore, we label an instance as a “Conflict” action only when the employed classifier (maximum entropy) outputs a predicted value closing to 1. The performance is shown in Row 2 of the Table 3. We use this setting to train a classifier and predict every sentence in document events. Entity co-occurrences in each “Conflict” sentence are calculated and the result is shown in Figure 5.

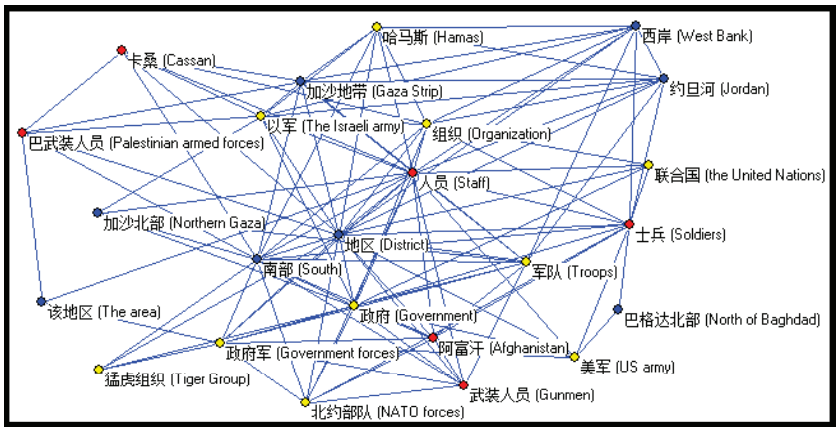

Fig. 5: Action Analysis

Figure 5 shows the result of action analysis on the discussed event. The edges represent the co-occurrence relations between entities. In this event, there are 12,076 sentences containing at least two entities, whereas 836 sentences have the “Conflict” action with value 1 outputted by the classifier. Among them, 3,221 entities co-occurred. In order to make the

result more comprehensible, edges with co-occurrence frequencies less than 12 are erased. Finally, a network with 25 entities is generated. In this example, entities (e.g., “哈马斯” (Hamas), “加沙北部” (the Gaza Strip), “阿富汗” (Afghanistan), “美军” (U.S. forces)) and the edges between them indicate meaningful information.

5.3.4 Social Network Analysis

In social network, many techniques (e.g., short path, cohesive subgroup and centre, etc.) have been proposed to discover the underlying structure of a network. To obtain a valid conclusion, these techniques are mainly implemented on networks with high quality of information. Therefore, these networks are often constructed by domain experts. However, the automatically extracted event networks may be error-prone. To get a reliable output from the event network by using social network analysis, more attentions should be paid for analyzing the output of these techniques. Figure 6 shows an example of this case in the traditional social network analysis.

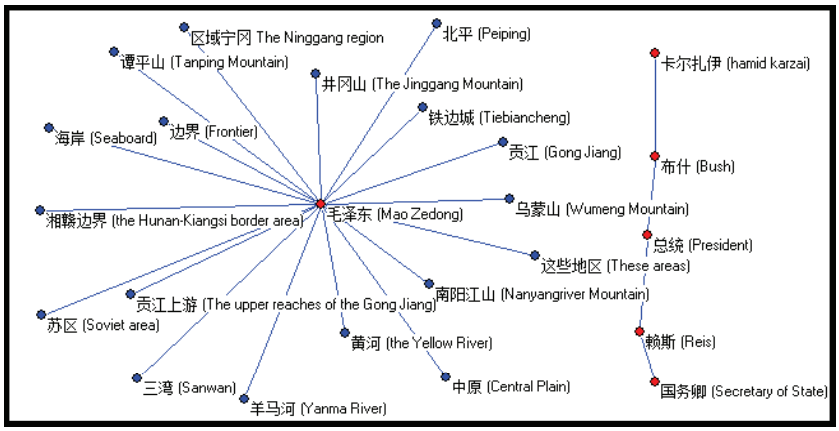

Fig. 6: Social Network Analysis

The data in this example comes from the results of “Information Filtering” and “PLT analysis”. The left hand side of Figure 5 seeks a short path between “尔扎伊” (Hamid Karzai) and “国务卿” (the Secretary of State); and they are connected by “PER-SOC” relations. On the right, “Mao Zedong” is set as the central figure to show directly collected locations, e.g., “井冈山” (Jinggangshan).

6 Conclusion and Future Work

Event network is an empirical construct for exploring open information. Techniques developed in IR, IE and linguistic networks are used to support document event detection, event network construction and event network analysis respectively. In this paper, we emphasize on the presentation of event network for exploring open information, where four methods are developed to show the flexibility of event network analysis. The traditional

systems developed by semi-supervised method may suffer from a “semantic drift” problem. In event network, by clustering documents into events, it will be helpful to reduce the influence caused by heterogeneous and noisy data. For each document event, the linguistic units are extracted independently. Then, the problem caused by “semantic drift” can be improved. The process to divide the whole documents into separate parts (“sports” or “politics”) could be beneficial for many NLP tasks (e.g., co-reference resolution or entity disambiguation). Furthermore, after event networks were constructed, topological information about an event becomes available, it also can be used to support many NLP techniques.

One limitation of our current work is that the presented methods to analyze event networks are mainly based database queries, which is not effective to show the underlying structure of the documents. Moreover, the evaluation of the extracted event for exploring open information is mainly based on human supervision. As ex-tracted from a cluster of documents, the event network contains the structural informa-tion of the underlying documents. This information is beneficial for some NLP tasks (e.g. entity disambiguation, co-reference resolution). Based on event networks, topological information can be used to support different kinds of analysis. It can also be used directly to support human oriented analysis. Therefore, in future work, techniques such as consistency judgement, statistical relational learning, entity disambiguation, co-reference resolution or manual intervention will be adopted to improve event networks. For researchers who are interested in our work, our code to implement the even network is available at (`https://github.com/YPench/EN/`).

7 Acknowledges

This research was supported in part by the National Science Foundation of China under grant 91418205, 61472315, 61540050, 61462011; The Open project NO. 2017BDKFJJ018; the Major Applied Basic Research Program of Guizhou Province NO.JZ20142001. Introduce Talents Science Projects of Guizhou University NO. 201650.

References

Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J. and others (2014) Relation Classification via Convolutional Deep Neural Network. *Proceedings of COLING'14*, pp. 2335--2344.

Sun, L. and Han, X. (2014) A Feature-Enriched Tree Kernel for Relation Extraction. *Proceedings of ACL'14*, ACL.

Mohamed, T., Hruschka Jr, E. and Mitchell, T. (2011) Discovering relations between noun categories. *Proceedings of ACL'11*, pp. 1447--1455, ACL.

Moro, A., Li, H., Krause, S., Xu, F., Navigli, R. and Uszkoreit, H. (2013) Semantic rule filtering for web-scale relation extraction. *Proceedings of ISWC'13*, pp. 347--362, Springer.

Banko, M., Etzioni, O. and Center, T. (2008) The Tradeoffs Between Open and Traditional Relation Extraction. *Proceedings of the ACL '08*, pp. 28--36, ACL.

Angeli, G., Premkumar, M. and Manning, C. (2015) Leveraging linguistic structure for open domain information extraction. *Proceedings of the ACL '15*, ACL.

Moro, A. and Navigli, R. (2013) Integrating Syntactic and Semantic Analysis into the Open Information Extraction Paradigm. *Proceedings of the IJCAI '13*, pp. 2148--2154.

Zhang, S., Duh, K. and Van Durme, B. (2017) MT/IE: Cross-lingual Open Information Extraction with Neural Sequence-to-Sequence Models. *Proceedings of the EACL '17*, pp. 64, ACL.

Kambhatla, N. (2004) Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relation. *Proceedings of the ACL '04*, pp. 22, ACL.

Che, W., Liu, T. and Li, S. (2005) Automatic entity relation extraction. *Journal of Chinese Information Processing*, pp. 1--6.

Zhang, P., Li, W., Hou, Y. and Song, D. (2011) Developing position structure-based framework for Chinese entity relation extraction. *ACM Transactions on Asian Language Information Processing*, pp.14, ACM.

Alex, B., Haddow, B. and Grover, C. (2007) Recognising nested named entities in biomedical text. *Proceedings of the BioNLP '07*, pp. 65--72, ACL.

Tang, B., Cao, H., Wang, X., Chen, Q. and Xu, H. (2014) Evaluating word representation features in biomedical named entity recognition tasks. *BioMed research international*, Hindawi.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J. and Bontcheva, K. (2015) Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, pp. 32--49, Elsevier.

Nothman, J., Ringland, N., Radford, W., Murphy, T. and Curran, J. (2013) Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence*, pp. 151--175.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K. and Dyer, C. (2016) Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Chiu, J. and Nichols, E. (2015) Named entity recognition with bidirectional LSTM-CNNs. *arXiv preprint arXiv:1511.08308*.

Ling, G., Asahara, M. and Matsumoto, Y. (2003) Chinese unknown word identification using character-based tagging and chunking. *Proceedings of the ACL '03*, pp. 197--200, ACL.

Carpenter, Bob (2006) Character language models for Chinese word segmentation and named entity recognition. *Proceedings of the SIGHAN '06*, pp. 169--172, ACL.

Chen, Y., Ouyang, Y., Li, W., Zheng, D. and Zhao, T. (2010) Using deep belief nets for Chinese named entity categorization. *Proceedings of the NEWS '10*, pp. 102--109, ACL.

Zhou, Y., Huang, C., Gao, J. and Wu, L. (2005) Transformation based Chinese entity detection and tracking. *Proceedings of the IJCNLP '05*, pp. 232--237.

Hacioglu, K., Douglas, B. and Chen, Y. (2005) Detection of entity mentions occurring in English and Chinese text. *Proceedings of the HLT-EMNLP '05*, pp. 379--386, ACL.

Chen, A., Peng, F., Shan, R. and Sun, G. (2006) Chinese named entity recognition with conditional probabilistic models. *Proceedings of the SIGHAN '06*.

Ahn, D. (2006) The stages of event extraction. *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pp. 1--8.

Fu, G. and Luke, K. (2005) Chinese named entity recognition using lexicalized HMMs. *Proceedings of the SIGKDD '05*, pp. 19--25.

Chen, W., Zhang, Y. and Isahara, H. (2006) Chinese named entity recognition with conditional random fields. *Proceedings of the SIGHAN '06*, pp. 118--121.

Kuzey, E. and Weikum, G. (2014) Evin: building a knowledge base of events. *Proceedings of the WWW '14*, pp. 103-106, IW3C2.

Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A.L., Aprosio, A.P., Rigau, G., and others (2016) NewsReader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. *Knowledge-Based Systems*, 110, pp. 60--85.

Brin, S. (1998) Extracting patterns and relations from the world wide web. *The World Wide Web and Databases*, pp. 172-183, Springer.

Allan, J., Carbonell, J., Doddington, G., Yamron, J. and Yang, Y. (1998) Topic detection and tracking pilot study final report.

Nallapati, R., Feng, A., Peng, F. and Allan, J. (2004) Event threading within news topics. *Proceedings of the CIKM '04*, pp. 446-453, ACM.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T. and Harshman, R. (1990) Indexing by latent semantic analysis. *JASIS*, 41(6), pp. 391-407.

Blei, D., Ng, A. and Jordan, M. (2003) Latent dirichlet allocation. *JMLR*, 3, pp. 993-1022.

Jones, K. (1972) A statistical interpretation of term specificity and its application in retrieval. *JDoc*, 28(1), pp. 11-21.

Luhn, H. (1957) A statistical approach to mechanized encoding and searching of literary information. *IBM J RES DEV*, 1(4), pp. 309-317.

Zhou, G., Su, J., Zhang, J. and Zhang, Min. (2005) Exploring various knowledge in relation extraction. *Proceedings of the ACL '05*, pp. 427-434, ACL.

Kambhatla, N. (2004) Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. *Proceedings of the ACL '04*, pp. 22, ACL.

Roth, D. and Yih, W. (2007) Global inference for entity and relation identification via a linear programming formulation. *Introduction to statistical relational learning*, pp. 553-580.

Roth, D. and Yih, W. (2002) Probabilistic reasoning for entity & relation recognition. *Proceedings of the COLINE '02*, pp. 1-7, ACL.

Collins, M. and Duffy, N. (2001) Convolution kernels for natural language. *Proceedings of the NIPS '01*, pp. 625-632.

Zelenko, D., Aone, C. and Richardella, A. (2003) Kernel methods for relation extraction. *JMLR*, 3, pp. 1083-1106, JMLR. org.

Zhang, M., Zhang, J., Su, J. and Zhou, G. (2006) A composite kernel to extract relations between entities with both flat and structured features. *Proceedings of the COLING-ACL '06*, pp. 825-832, ACL.

Curran, J., Murphy, T. and Scholz, B. (2007) Minimising semantic drift with mutual exclusion bootstrapping. *Proceedings of the PACL '07*, (3), ACL.

Riedel, S., Yao, L., McCallum, A. and Marlin, B. (2013) Relation extraction with matrix factorization and universal schemas.

Takamatsu, S., Sato, I. and Nakagawa, H. (2012) Reducing wrong labels in distant supervision for relation extraction. *Proceedings of the ACL '12*, pp. 721-729, ACL.

Xu, Y., Kim, M., Quinn, K., Goebel, R. and Barbosa, D. (2013) Open Information Extraction with Tree Kernels. *Proceedings of the HLT-NAACL '13*, pp. 868-877.

Downey, D., Schoenmackers, S. and Etzioni, O. (2007) Sparse information extraction: Unsupervised language models to the rescue. *DTIC Document*.

Wang, W., Besançon, R., Ferret, O. and Grau, B. (2011) Filtering and clustering relations for unsupervised information extraction in open domain. *Proceedings of the CIKM '11*, pp. 1405-1414, ACM.

Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T. and Bogaard, T. (2016) Building event-centric knowledge graphs from news. *Journal of Web Semantics*, Elsevier.

Padró, L., Agić, Ž., Carreras, X., Fortuna, B., Garcia-Cuesta, E., Li, Z., Štajner, T. and Tadić, M. (2014) Language processing infrastructure in the xlike project. *Proceedings of the LREC 2014*.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. and Ives, Z. (2007) Dbpedia: A nucleus for a web of open data. , Springer.

Lenat, D. (1995) CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11), pp. 33-38, ACM.

Miller, G. (1995) WordNet: a lexical database for English. *Communications of the ACM*, 38(11), pp. 39-41, ACM.

McCallum, A. (2005) Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9), pp. 48-57, ACM.

Zhang, Y. and Callan, J. (2004) Cmu dir supervised tracking report. *les actes de TDT*.

Trieschnigg, D. and Kraaij, W. (2004) TNO Hierarchical topic detection report at TDT 2004. *Topic Detection and Tracking Workshop Report*.

Brin, S. (1998) Extracting patterns and relations from the world wide web. *The World Wide Web and Databases* pp. 172-183, Springer.

Agichtein, E. and Gravano, L. (2000) Snowball: Extracting relations from large plain-text collections. *Proceedings of DL '00* pp. 85-94, ACM.

Angel, A., Sarkas, N., Koudas, N. and Srivastava, D. (2012) Dense subgraph maintenance under streaming edge weight updates for real-time story identification. *PVLDB* **5**: 574-585, VLDB.

Banko, M., Cafarella, M.J., Soderland, S., Broadhead, M. and Etzioni, O. (2007) Open information extraction for the web. *Proceedings of IJCAI '07* **7**: 2670-2676, IJCAI.

Batagelj, V. and Mrvar, A. (1998) Pajek-program for large network analysis. *Connections* **21**: 47-57.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T. and Taylor, J. (2008) Freebase: a collaboratively created graph database for structuring human knowledge. *Proceedings of SIGMOD '08* pp. 1247-1250, ACM.

Chen, Y., Zheng, Q. and Chen, P. (2015) A Boundary Assembling Method for Chinese Entity Mention Recognition. *IEEE INTELL SYST* **30**: 50-58, IEEE.

Chen, Y., Zheng, Q. and Chen, P. (2015) Feature assembly method for extracting relations in Chinese. *AI* **228**: 179-194, Elsevier.

Chen, Y., Zheng, Q. and Zhang, W. (2014) Omni-word Feature and Soft Constraint for Chinese Relation Extraction. *Proceedings of ACL'14* pp. 572-581, ACL.

Csardi, G. and Nepusz, T. (2006) The igraph software package for complex network research. *IJ COMP SYS* **1695**: 1-9.

Das Sarma, A., Jain, A. and Yu, C. (2011) Dynamic relationship and event discovery. *Proceedings of WSDM '11* pp. 207-216, ACM.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S. and Weischedel, R. (2004) The automatic content extraction (ACE) program--tasks, data, and evaluation. *Proceedings of LREC '04* **4**: 837-840, Citeseer.

Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S. and Yates, A. (2005) Unsupervised named-entity extraction from the web: An experimental study. *AI* **165**: 91-134, Elsevier.

Etzioni, O., Fader, A., Christensen, J., Soderland, S. and Mausam, M. (2011) Open Information Extraction: The Second Generation. *Proceedings of IJCAI '11* **11**: 3-10, IJCAI.

Hoffmann, R., Zhang, C. and Weld, D.S. (2010) Learning 5000 relational extractors. *Proceedings of ACL '10* **10**: 286-295, ACL.

Kozareva, Z. and Hovy, E. (2010) Learning arguments and supertypes of semantic relations using recursive patterns. *Proceedings of ACL'10* pp. 1482-1491, ACL.

Kuzey, E., Vreeken, J. and Weikum, G. (2014) A Fresh Look on Knowledge Bases: Distilling Named Events from News. *Proceedings of CIKM '14* pp. 1689-1698, ACM.

Leydesdorff, L. and Vaughan, L. (2006) Co-occurrence matrices and their applications in information science: Extending ACA to the Web environment. *JASIST* **57**: 1616-1628, Wiley Online Library.

Liu, M., Liu, K., Xu, L. and Zhao, J. (2014) Exploring Fine-grained Entity Type Constraints for Distantly Supervised Relation Extraction. *Proceedings of COLING'14* pp. 2107-2116, ACL.

McIntosh, T., Yencken, L., Curran, J.R. and Baldwin, T. (2011) Relation Guided Bootstrapping of Semantic Lexicons. *Proceedings of ACL '11* pp. 266-270, ACL.

Mintz, M., Bills, S., Snow, R. and Jurafsky, D. (2009) Distant supervision for relation extraction without labeled data. *Proceedings of ACL '09* pp. 1003-1011, ACL.

Parikh, R. and Karlapalem, K. (2013) Et: events from tweets. *Proceedings WWW '13* pp. 613-620, IW3C2.

Phan, X.H. and Nguyen, C.T. (2007) GibbsLDA++: AC/C++ implementation of latent Dirichlet allocation. *Technical report*.

Piskorski, J., Tanev, H., Atkinson, M., Van Der Goot, E. and Zavarella, V. (2011) Online news event extraction for global crisis surveillance. *TCCI* pp. 182-212, Springer.

Ramakrishnan, N., Butler, P., Muthiah, S., Self, N. and others (2014) 'Beating the news' with EMBERS: Forecasting Civil Unrest using Open Source Indicators. *Proceedings SIGKDD '14* pp. 1799-1808, ACM.

Ritter, A., Mausam, Etzioni, O. and Clark, S. (2012) Open domain event extraction from twitter. *Proceedings of SIGKDD '12* pp. 1104-1112, ACM.

Sowa, J.F. (1984) Conceptual structures: information processing in mind and machine. Addison-Wesley Pub.

Suchanek, F., Kasneci, G. and Weikum, G. (2007) Yago: a core of semantic knowledge. *Proceedings of WWW '07* pp. 690-706, ACM.

Weld, D.S, Hoffmann, R. and Wu, F. (2009) Using wikipedia to bootstrap open information extraction. ACM SIGMOD Record **37**: 266-270, ACM.

Yang, Y., Carbonell, J.G., Brown, R.D., Pierce, T., Archibald, B.T. and Liu, X. (1999) Learning approaches for detecting and tracking news events. *IEEE INTELL SYST* **14**: 32-43, IEEE.

Zhu, J., Nie, Z., Liu, X., Zhang, B. and Wen, J. (2009) StatSnowball: a statistical approach to extracting entity relationships. *Proceedings of WWW '09* pp. 101-110, ACM.