Zhou XG, Gong RB, Shi FG *et al.* PetroKG: Construction and application of knowledge graph in upstream area of PetroChina. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 35(2): 368–378 Mar. 2020. DOI 10.1007/s11390-020-9966-7

PetroKG: Construction and Application of Knowledge Graph in Upstream Area of PetroChina

Xiang-Guang Zhou[1], Ren-Bin Gong[1], Fu-Geng Shi[1], and Zhe-Feng Wang[2]

[1] *PetroChina Research Institute of Petroleum Exploration and Development, Beijing 100083, China*
[2] *Huawei Technologies, Hangzhou 310007, China*

E-mail: {zhouxg69, gongrb, sfg}@petrochina.com.cn; wangzhefeng@huawei.com

Received August 20, 2019; revised December 31, 2019.

Abstract There is a large amount of heterogeneous data distributed in various sources in the upstream of PetroChina. These data can be valuable assets if we can fully use them. Meanwhile, the knowledge graph, as a new emerging technique, provides a way to integrate multi-source heterogeneous data. In this paper, we present one application of the knowledge graph in the upstream of PetroChina. Specifically, we first construct a knowledge graph from both structured and unstructured data with multiple NLP (natural language progressing) methods. Then, we introduce two typical knowledge graph powered applications and show the benefit that the knowledge graph brings to these applications: compared with the traditional machine learning approach, the well log interpretation method powered by knowledge graph shows more than 7.69% improvement of accuracy.

Keywords knowledge graph, natural language processing, oil and gas industry

1 Introduction

Exploration and production (E&P), also known as the upstream sector of the oil and gas industry, is the process of searching for oil and gas deposits and taking measures to extract these resources from the earth for commercial sale. During the process of E&P, PetroChina accumulates a large amount of data, such as raw data, production data, and results data. These data are viewed as one of the most important assets in the upstream area of PetroChina. Consequently, how to manage the data asset such that they can be fully utilized becomes an urgent challenge.

In recent years, the knowledge graph has received much attention from both academics and industry due to its wide application in many industry domains. As a way of organizing the information, knowledge graphs provide a unified way to represent and store different kinds of heterogeneous data. Therefore, knowledge graphs are a good choice to integrate multi-source heterogeneous data in the E&P area and make full use of the data asset.

To this end, we construct an E&P knowledge graph in the upstream area of PetroChina and develop several applications based on the knowledge graph. Specifically, we make the following contributions.

First, we build an E&P knowledge graph from the production data of the upstream of PetroChina. We first integrate all the heterogeneous data from various sources into a unified format and then map them into triplets with respect to a pre-defined ontology. We also apply the same process flow to the online encyclopedia of the petroleum industry, which provides a high-quality source of semi-structured oil knowledge.

Second, to further enrich the knowledge graph, we try to extract triplets directly from the petroleum documents crawled from online websites. Since the documents are unstructured, we use multiple NLP methods to achieve the goal.

Last, we develop two applications based on the E&P knowledge graph: semantic search and well log interpretation. We show that the knowledge graph plays a key role in both applications.

The rest of the paper is organized as follows. We

Regular Paper
Special Section on Learning and Mining in Dynamic Environments
©Institute of Computing Technology, Chinese Academy of Sciences 2020

review related work in Section 2. In Section 3, we introduce the process of constructing the knowledge graph. In Section 4, we demonstrate two typical applications based on the knowledge graph. In Section 5, we conclude the paper.

2 Related Work

2.1 Information Extraction

Information extraction (IE) is a fundamental component in any knowledge graph construction pipeline. The goal of an IE system is to extract useful information from raw data, usually text or Web pages.

2.1.1 Named Entity Recognition

Named entity recognition (NER) is the task of locating and classifying named entities in texts. It is usually one of the first processing steps in IE. Traditional NER systems require a large amount of specific knowledge and hand-crafted features, which are expensive to construct and maintain [1–3]. Recently, many systems have been introduced by research studies that outperform traditional NER systems. These novel systems combine traditional methods with neural network architectures such that feature engineering is not necessary any more. For instance, Limsopatham and Collier [4] used a bidirectional LSTM neural network to leverage orthographic features and achieve the first place on WNUT-2016 shared task①. A new architecture based on bidirectional LSTM and conditional random fields (CRF) was proposed by [5]. Their system obtains the state-of-the-art performance without relying on hand-crafted features. Ma and Hovy [6] proposed a neural network architecture that benefits from both word- and character-level representations automatically, by using a combination of bidirectional LSTM, convolutional neural networks (CNN) and CRF. This system is truly end-to-end, i.e., it requires neither feature engineering nor data preprocessing. In our task, we choose to use BERT [7] for NER because BERT currently defines the state of the art.

2.1.2 Relation Extraction

Most researchers construct knowledge bases by extracting triplets information from massive unstructured data. Early researches mainly used a dependency parser to analyze the semantic information in sentences. For example, the OpenIE [8] system from Stanford University provides a dependency parsing method to extract relation triplets from plain text without any labeled data. Meanwhile, the research team from Carnegie Mellon University used their NELL [9, 10] system to extract over 50 million beliefs, which only uses a small amount of labeled data as seeds to extract information from Web pages. Christensen *et al.* [11, 12] applied semantic role labeling on open information extraction and achieved good results.

With the rise of deep learning and the proposal of large-scale datasets, many deep learning based methods have been proposed. Santos *et al.* [13] and Wang *et al.* [14] considered the relation extraction process as a pipeline which consists of two steps: entity extraction and relation classification. Due to the lack of labeled data, Zeng *et al.* [15] applied distant supervision method to relation extraction, which uses relation triples from an existing knowledge base to obtain labeled data. To reduce the impact of error accumulation in a pipeline, joint learning based methods were proposed. Miwa and Bansal [16] and Zheng *et al.* [17] combined the entity extraction model and the relation extraction model by jointly training the two models simultaneously, while [18] proposes a novel tagging scheme to extract entities and relations simultaneously. Since different methods have their unique advantages, we choose to merge the results from dependency parsing and semantic role labeling as our triplet candidates.

2.2 Knowledge Graph Embedding

In the recent years, a variety of methods of representation learning for knowledge graphs have been introduced, many of which encode both entities and relations into a continuous low-dimensional vector space. TransE [19] projects both entities and relations into the same continuous low-dimensional vector space. Here, relations are considered to be translation operations between head and tail entities. The energy function is defined as follows:

$$E(h, r, t) = ||h + r - t||,$$

which indicates that the tail embedding t should be the nearest neighbour of $h + t$. TransE is both effective and efficient when tackling 1-to-1 relations, but modeling more complicated entities and relations may lead to problems. To address this issue, TransH [20] attempts to interpret relations as translation operations on relation-specific hyperplanes, allowing entities to play different roles in different relationships. TransR [21] models entities and relations in separate entity and relation spaces.

① http://noisy-text.github.io/2016/ner-shared-task.html, Dec. 2019.

It uses relation-specific matrices to project entities from entity spaces to relation spaces. TransD [22] further considers the diversities of both entities and relations. It uses a dynamic mapping matrix for multiple representations of entities. In our application, we choose to use TransR because of its efficiency and effectiveness on N-to-N relations.

2.3 AI in Petroleum Industry

Recent years have witnessed the rapid development of machine learning and AI techniques, which also benefit many industries, such as educational industry [23, 24], transportation industry [25, 26] and energy industry [27, 28]. Along this line, more and more researchers leverage machine learning methods in the petroleum industry, such as for the inspection of oil pipelines [28, 29], and drilling report mining with natural language processing (NLP) techniques [30–32]. One of the most important research fields that has just been emerged is petroleum knowledge management. Data in petroleum industries is complex in nature and often poorly organized, duplicated, and heterogeneous. Without a good organization, knowledge can hardly be extracted, understood or applied. Thus, the search for a method of petroleum data and knowledge management is an urgently vital task. Early studies like [33] presented a petroleum exploration domain ontology-based knowledge integration and a sharing system framework. This framework minimizes the complexity of heterogeneous data, and enhances the power of knowledge integration and information sharing among different operational units. As far as we know, we are among the first group who leverages a knowledge graph for petroleum knowledge management on a large-scale real-world dataset.

3 Knowledge Graph Construction

In this section, we introduce how we construct the knowledge graph (i.e., PetroKG) with the data in the upstream (E&P) of PetroChina. We firstly design the concept architecture of PetroKG. Specifically, PetroKG contains five concept categories, i.e., Geology, Top Design, Activity, Document and Material, and each concept encompasses definitions of the entities, properties and relations between the entities. In PetroKG, there are 876 well entities, 47 438 stratum entities and 15 787 concept entities with all the related properties and relations. We collect the data mainly from three sources. The first is production data from the upstream of PetroChina, which is structured but heterogeneous. The second source is an online encyclopedia of the petroleum industry, which is a collection of semi-structured web pages. For these two kinds of data, we first integrate them into a unified format and then map them into formal triplets. The last data source is the literature, which is composed of unstructured petroleum documents. We directly extract triplets from these documents with information extraction techniques. The data sources are summarized in Fig.1.

3.1 Structure Knowledge Extraction

Most of the available production data is structured data in our case. Although the data is structured, they are in different formats and distributed in multiple sources. The integration of these multi-source heterogeneous data is a huge challenge for us.

3.1.1 *Data Integration*

As mentioned above, to construct our domain knowledge graph, we first integrate the multi-source structured production and concept data. As shown in Fig.2, our structure data is mainly from four external sources: well data, stratum data, block data, and PetroWiki data. Specifically, the well data contains all the well log information which is exported as CSV files by the well log processing tool. The stratum data contains all the description of the geologic stratums. The block

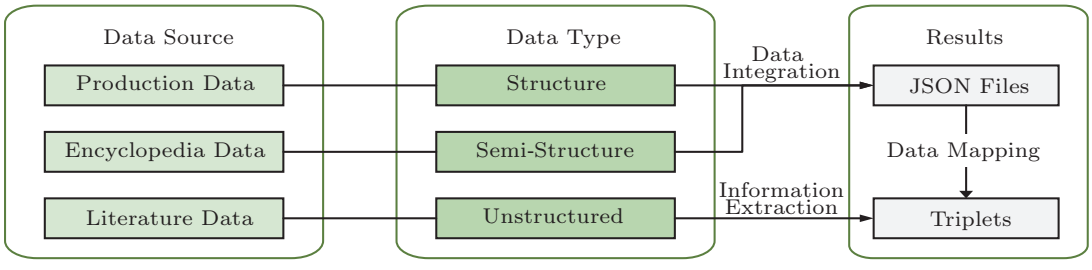


Fig.1. Data sources for knowledge graph.

data contains information about an area, such as wells and reservoirs in an area. Besides, the PetroWiki data is a special source from the online encyclopedia② of the petroleum domain and this data is semi-structured.

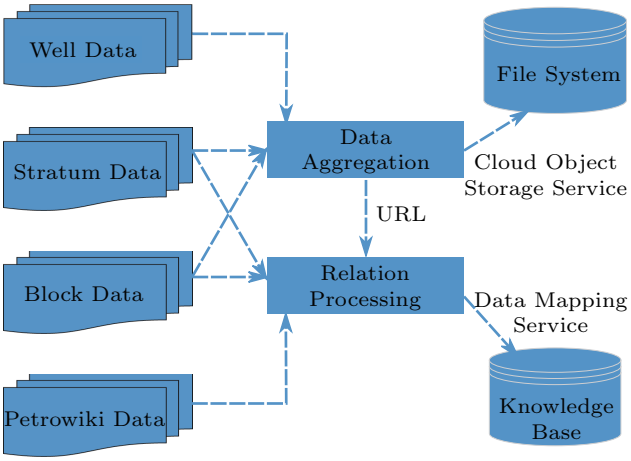


Fig.2. Data integration for knowledge graph.

Considering the different sources of the data, we design different processing and storage procedures. For all the production data, i.e., well data, stratum data, and block data, we first take all the production data files from different external tools. Then, we use our data aggregation module to merge the multi-source data into a unified format. This format contains the log data of the specified wells with related geologist stratum and block information. At last, we use a cloud object storage service to save the whole data and send the data URL to the next step, i.e., relation processing module.

In the relation processing module, we mainly extract the relations from the structured production and concept data. In this module, we do not use the well data in the production source. This is because the relationships between wells, such as well-adjacent relations and stratum-sharing relations, are contained in stratum and block data already. For these structured production data, we conflate all the formal relations and convert the knowledge into a unified JSON file. The storage path of the whole data is added as an URL for the specified well. At the same time, we also extract knowledge from semi-structured encyclopedia data. We transform the concepts to the pre-defined JSON format. So far, we have conflated all the structured multi-source data and converted them into a unified format.

3.1.2 Data Mapping

After we integrate all the heterogeneous data in a unified JSON format, we convert the JSON files into triplets with pre-defined schema and artificial rules. The procedure of data mapping mainly consists of two steps.

First, we extract original triplets from the JSON files. We create extraction rules according to the advice from oil and gas experts. Then, we extract all the target key-value pairs from the JSON files using those handcrafted rules and convert them into triplets.

Second, we map the original triplets into formal triplets with respect to the predefined schema. To do this, we calculate the similarity between the predicates (properties or relations) of original triplets and the predefined schema. We filter out all the predicates that are not in the schema and map the remaining predicates to the schema that has the highest matching scores.

To this end, we have built an E&P knowledge graph from the structured and semi-structured data from the upstream of PetroChina. The statistics of structured knowledge extraction is shown in Table 1.

Table 1. Statistics of Structured Knowledge

Source	Category	Entity	Triplet
Production data	3	113 167	448 586
Encyclopedia data	81	15 787	135 205
Total	84	128 954	583 791

3.2 Unstructured Knowledge Extraction

To further enrich the knowledge graph, we extract more triplets from online petroleum documents. Since these documents are unstructured, we utilize multiple NLP methods to extract knowledge from them. The overall framework is illustrated in Fig.3, which includes three parts, namely 1) data preprocessing, 2) extraction processing, and 3) postprocessing. Technical details will be introduced in the following subsections.

3.2.1 Data Preprocess

The unstructured documents come from China Petroleum Exploration③. In total, 200 papers in HTML format and 452 papers in PDF format are crawled. We use PDFMiner④ to transform the PDF

②http://baike.yooso.com.cn, Dec. 2019.
③http://www.xml-data.org/ZGSYKT/html/2019/3/20190301.htm, Dec. 2019.
④https://pypi.org/project/pdfminer/, Dec. 2019.

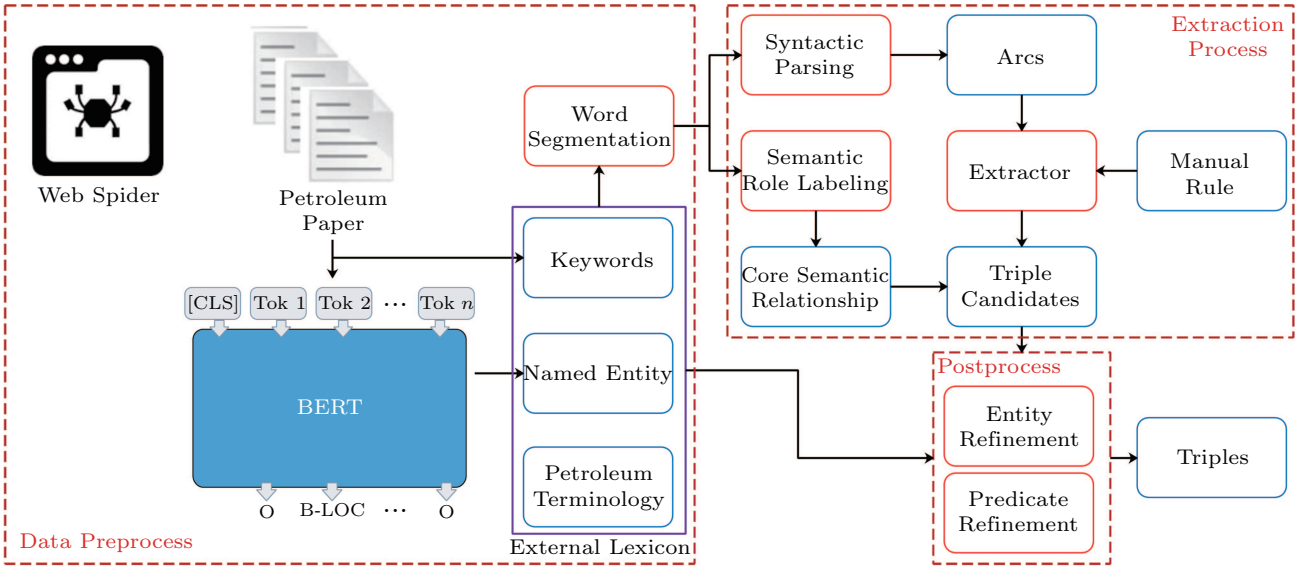


Fig.3. Framework of unstructured knowledge extraction. "Tok" means token.

files into text format. After data cleaning and deduplication, we extract 593 papers in plain text format. Then, we use the Language Technology Platform (LTP) system [34] for word segmentation and part-of-speech (POS) tagging. Note that since LTP is trained on the specific domain, it cannot obtain ideal performance in our case. Consequently, we construct an external lexicon that consists of existing petroleum terminology, keywords in crawled papers, and named entities recognized by a pre-trained BERT model.

3.2.2 *Extraction Process*

Due to the lack of labeling data, we extract triplets through dependency parsing and semantic role labeling.

Dependency parsing is defined as the task to analyze the grammatical structure of a sentence and parse relationships as a tree structure. Fig.4 shows an example of such a tree structure. Note that the red words are the tags of parse relationships⑤. To be specific, we choose some special verbs as predicates, such as "位于" (located in) and "具有" (have). Then, we extract entities which are centered on these predicates and connected by specific dependency tags:

1) subject of verb (SBV) and object of verb (VOB), from which we could obtain triplets like (潜山带/buried hill, 位于/locate in, 凹陷东部/eastern depression);

2) attribute relation (ATT) and object of verb (VOB), from which we could obtain triplets like (潜山带/buried hill, 具有/have, 背景/background).

Semantic role labeling is defined as the task to recognize arguments for a given predicate and assign semantic role labels to them. As shown in the same sentence, the predicate (REL) like "位于" (located in) is the keyword in the sentence and expresses some actions. It usually is a verb or an adjective word. The core semantic role (ARG0), such as "南马庄" (Nanmazhuang) and "潜山带" (buried hill), usually

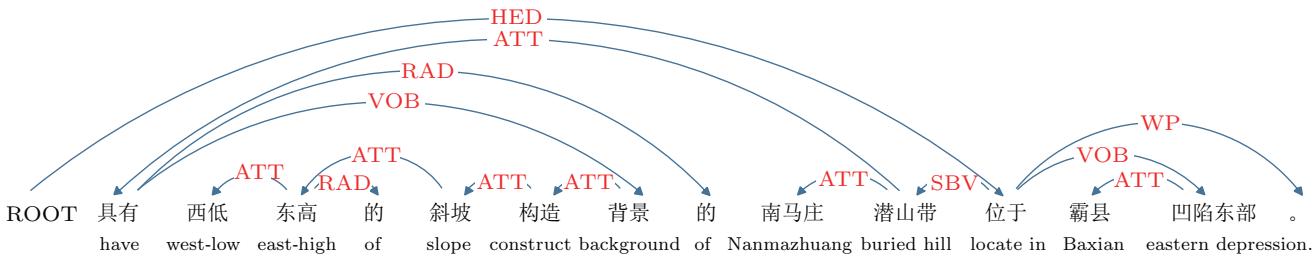


Fig.4. Example of dependency parsing.

⑤http://www.ltp-cloud.com/intro#dp_how, Dec. 2019.

indicates the performer of the action, while role (ARG1), such as "霸县" (Baxian) and "凹陷东部" (eastern depression), usually indicates the influence or recipient of the action. After semantic role labeling, we extract the semantic role of the performer and the recipient of an action with the predicate to form a triplet like (南马庄潜山带/Nanmazhuang buried hill, 位于/located in, 霸县凹陷东部/Baxian eastern depression). To obtain better results, we add some manual rules to filter out some wrong samples.

3.2.3 *Postprocess*

After obtaining triplet candidates, it is necessary to filter the inappropriate ones.

Entity Refinement. First, we filter entities that do not appear in the external dictionary and merge adjacent entities. Then, to obtain more meaningful entities, we merge the entities with the words that have attribute (ATT) relations in the dependency parsing tree or share the same role labels in the result of semantic role labeling. As a result, we obtain more realistic entities such as "南马庄潜山带" (Nanmazhuang buried hill) and "西低东高的斜坡构造背景" (west-low east-high slope structure background) instead of meaningless words like "潜山带" (buried hill) and "背景" (background).

Predicate Refinement. Only the top 10% of frequent predicates are kept since the rest appear very few times. The relation frequency is shown in Fig.5. After that, we merge the predicates whose meanings are similar.

Evaluation. In order to evaluate the postprocessing, a human study is carried out, similar to [35]. Three experts with several years of experience in the petroleum domain perform the evaluation. Specifically, we randomly select 200 triples and test the precision. Some instances of extracted triplets are shown in Table 2 and the experimental results are summarized in Table 3.

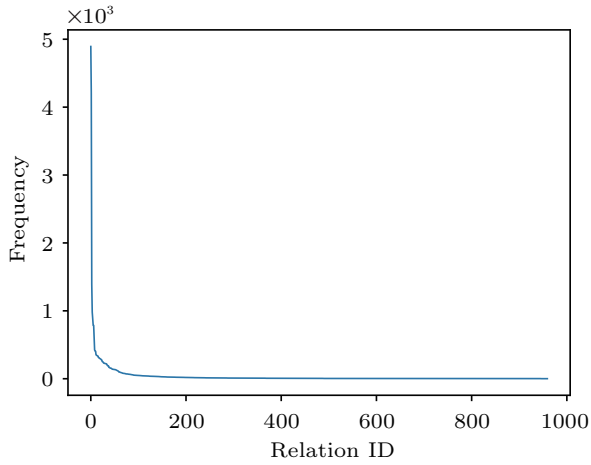


Fig.5. Relation frequency.

4 Applications of Knowledge Graph

Equipped with the knowledge graph, we can apply it in many scenarios of E&P. In this section, we will show two typical applications that benefit from the knowledge graph.

4.1 Semantic Search

In PetroChina, the researchers and engineers often need to search a large amount of documents and industrial data in their daily work. Therefore, an efficient knowledge retrieval and semantic search service is essential for researchers and engineers. In this subsection, we will introduce a semantic search service based on the knowledge graph.

In the process of constructing the E&P knowledge graph, we have integrated different kinds of data from multiple data sources in a single knowledge graph. In this way, a unified data access interface can be provided for up-level applications. This is the basis for a semantic search service.

Table 2. Extracted Relation Triples

Subject	Relation	Object
冀中坳陷 (Jizhong depression)	位于(locate in)	华北 (North China)
沁水盆地 (Qinshui basin)	位于 (locate in)	山西省 (Shanxi province)
二连盆地 (Erlian basin)	位于 (locate in)	内蒙古自治区 (Inner Mongolia Autonomous Region)
烃源岩 (source rock)	是 (is a)	生油岩 (source rock of petroleum)
泊松比 (Poisson's ratio)	是 (is a)	地层岩性参数 (Stratigraphic lithology parameters)
三水盆地 (Sanshui basin)	是 (is a)	盆地 (basin)
凹陷周缘 (periphery of depression)	形成 (form)	沉积中心(depocenter)
构造层 (structural layer)	形成 (form)	隔挡式褶皱 (partition style fold)
油气充注 (hydrocarbon charging)	形成 (form)	烃包裹体 (hydrocarbon inclusion)

Table 3. Results of Unstructured Knowledge Extraction

	#Entities	#Triplets	Precision
Without postprocess	27 053	36 260	0.50
With postprocess	22 727	12 828	0.58

Note: # means number of.

The semantic search system consists of two parts: the online and the offline part. The online part handles user input and returns the answer. The offline part mines query templates used in the online part. The whole system is summarized in Fig.6.

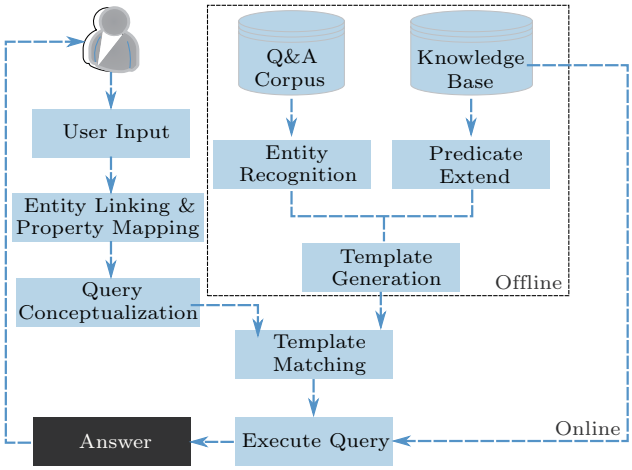


Fig.6. Overview of semantic search service.

In the offline part, we generate query templates from a question and answering (QA) corpus (user query log in our case). To further improve the performance, we manually write a few templates as well.

In the online part, whenever we get a user input, we first recognize the entities and properties with entity linking tools. In our system, we develop a dictionary based entity linking tool. Specifically, we construct a fine state automaton which encodes all known entities and their aliases. In the running time, the Aho-Corasick algorithm is used to match all token sequences in the dictionary with the user input. With all the recognized mentions, we link them to the entities in the knowledge graph. The method of property recognition and mapping is similar to the aforementioned process.

After we get the entities and properties in the user input, we will try to match the user input to one of the query templates we mined in the offline part. Specifically, we first conceptualize the user input by replacing the recognized entities and properties with special symbols and then calculate the semantic similarity between the conceptualized user input and the query templates.

Finally, we fill the matched template with the proper values and perform the query in the knowledge graph. In our system, we use the Graph Engine Service[6] as our graph database to host the knowledge graph. Thus, we convert the template into a Gremlin query statement and execute the query.

To better illustrate the process of semantic search, we provide a real user input and show the output of each step. The example is shown in Fig.7.

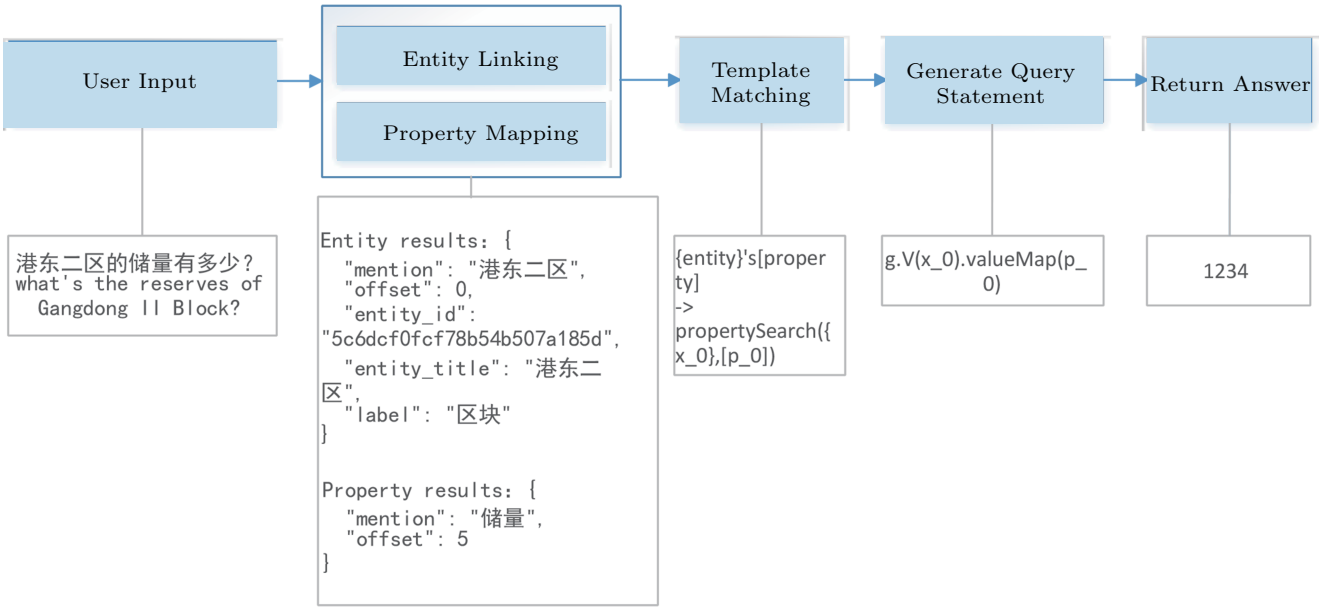


Fig.7. Example of semantic search.

[6]https://www.huaweicloud.com/en-us/product/ges.html, Nov. 2019.

4.2 Well Log Interpretation

In the petroleum industry, the most essential operation is oil and gas reservoir exploration. In the early stage, the exploration mainly relies on the seismic facies analysis from the drilling data, such as well log recording. The experts would spend much time to read these massive recording data and compute the porosity and permeability. Since the geological conditions may be utterly disparate in two different fields, these traditional methods highly rely on the experts' experience. Thus, automated algorithms to determine reservoirs are urgent requirements in the petroleum industry. Along this line, in this subsection, we will introduce an automated well log interpretation module based on our E&P knowledge graph.

In the real industrial scenario, reservoir exploration mainly relies on experts' interpretations on various well logs. Manual interpretations are inefficient because of the massive well log data. Moreover, the geological conditions of reservoirs are complex and miscellaneous, which makes traditional manual interpretations highly relied on the experts' experiences.

Indeed, after well logging, massive data of digital measurements for the geologic facies have been recorded, such as gamma radiation (GR), resistivity, spontaneous potential, which is difficult for both experts and AI systems to process all the data simultaneously. To overcome this challenge, we develop a unified well log interpretation service to automatically detect and classify reservoirs. As shown in Fig.8, the model consists of two modules, i.e., the potential reservoirs detection (PRD) and the reservoir classification (RC).

In the PRD module, we use the existing experts' knowledge to detect potential reservoirs. Specifically, we first apply some necessary preprocessing procedures to the well log data. Then, for the parameter choosing, we can build a geologic model to compute the critical geologic properties, such as porosity, permeability and fluid saturation. From the expert rule interface, we can obtain the basic geologic knowledge of the area and expert rule on the geologic properties. Next, based on the expert rule, we can easily analyze the relationship between the geologic properties and probability of reservoirs. To this end, we can filter the depth interval with low potential.

In the RC module, we use machine learning methods to build a knowledge-based reservoir classification model. Along this line, we first reload the well log data of potential reservoirs filtered out by PRD. Then, from the model loading, we can build a new model or choose the checkpoint of our pre-trained model. Next, from PetroKG, we can load the feature choice of experts, and retrain or fine-tune the loading model. Finally, we can generate the reservoir classification results by the pre-trained machine learning model.

We conduct extensive experiments on the real industrial data of PetroChina. Indeed, from the RC module, we can easily choose the base classification model, such as k-nearest neighbors (KNN) [36], Random Forest (RF) [37], Multilayer Perceptron (MLP) [38], and Gradient Boosting Machine (GBM) [39]. More specifically, we set $k = 5$ with euclidean distance evaluation for KNN classifier and for RF model, and we set the number of

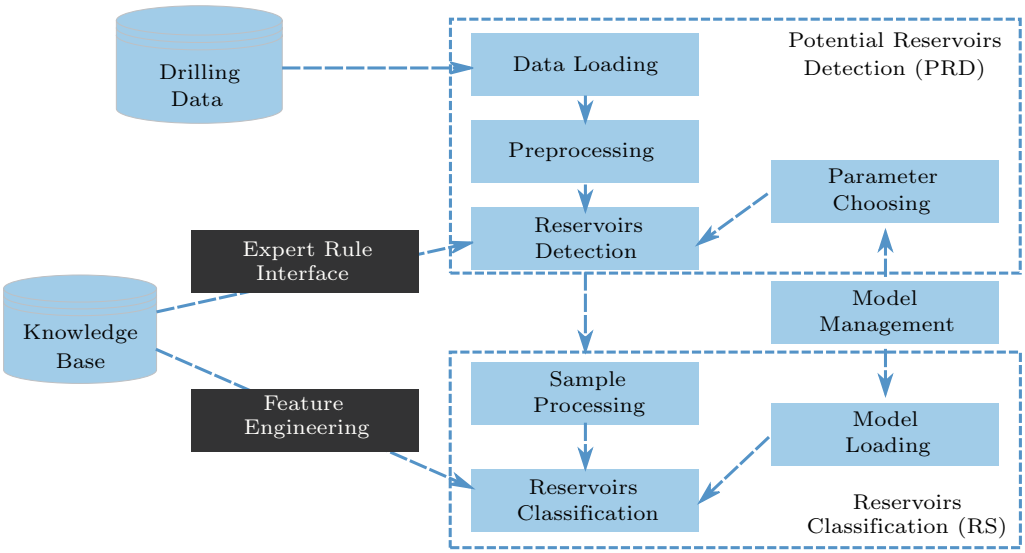


Fig.8. Overview of well log interpretation service.

estimators to 100. For the MLP model, we develop three layers, 128-dimensional perceptrons with the rectified linear unit and a softmax function for the reservoir classification task. For the GBM model, we implement two specific variants GBM-Naive and GBM-KG. For GBM-Naive, we develop a naive GBM model with the leaf-wise algorithm. We set the leaf number to 150 and learning rate to 0.01. Moreover, we implement a variant GBM model (i.e., GBM-KG) which incorporates the feature engineering based on the experts' knowledge in PetroKG. Table 4 demonstrates the compared results of different base models. We can see that with the application of PetroKG, our proposed well log interpretation model has achieved 86% average accuracy on the oil and gas exploring task, with more than 7.69% improvement over the traditional machine learning approaches.

Table 4. Comparison Results of Different Base Models

Model	Precision	Recall	$F1$-Score
KNN	0.67	0.68	0.67
RF	0.75	0.79	0.77
MLP	0.54	0.74	0.63
GBM-Naive	0.76	0.80	0.78
GBM-KG	0.82	0.86	0.84

Table 5 demonstrates the results of well log interpretation in various reservoir layers. It is worth to note that the results of the "Poor" layer and the "Oily Water" layer are closed to zero. That is because the sample of the "Poor" layer is very small. Moreover, the "Oily Water" layer and the "Water" layer have very similar physicochemical properties in the well logs, and the classification of these two layers is highly subjective. Fig.9 demonstrates a sample result of well log interpretation service. Visually, the interpretation of our model is very similar to the experts' interpretation results.

Table 5. Results of Well Log Interpretation

Layer	Number of Samples	Precision	Recall	$F1$-Score
Water	196 392	0.89	0.97	0.93
Oil	14 273	0.67	0.37	0.47
Dry	18 358	0.72	0.67	0.69
Water-bearing oil	9 656	0.43	0.18	0.26
Gas	1 263	0.37	0.17	0.23
Poor	522	0.00	0.00	0.00
Oily water	4 720	0.02	0.00	0.00
Total	245 184	0.82	0.86	0.84

5 Conclusions

In this paper, we explored how to integrate multi-sources heterogeneous data in the E&P area of PetroChina. We reported the practical methods and results in the process of constructing the knowledge graph. Also, we introduced two typical applications

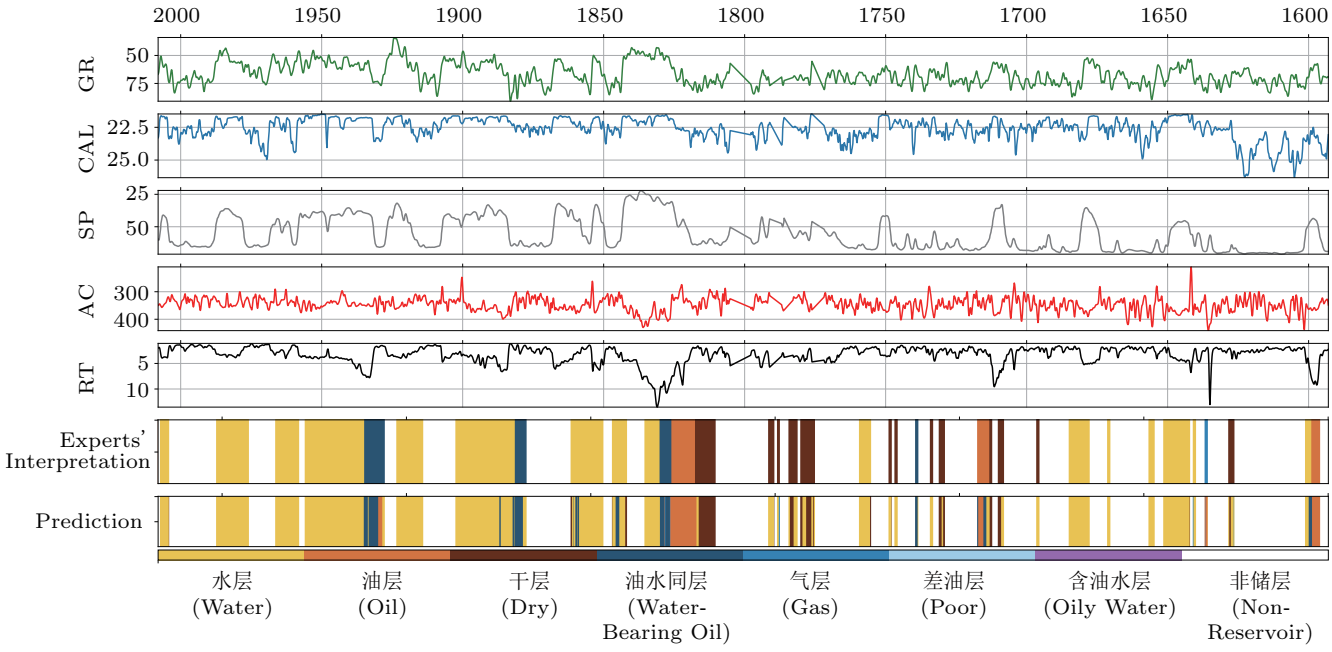


Fig.9. Sample result of the well log interpretation on the well uNs=#3-60-2. The five first rows show the digital measurements as a function of depth (in meters). Two facies rows illustrate the interpretation by experts and prediction by our model. The colorbar on the bottom gives the correspondences to the reservoir class.

of the E&P knowledge graph. We showed how to utilize the knowledge graph in the applications. The semantic search service directly returns the most match answer to users from the knowledge graph instead of an unstructured document and the well log interpretation method improves the accuracy by more than 7.69% with the knowledge graph as an external source of knowledge. With the benefit of knowledge graph, the applications offer better services to users in the oil and gas industry.

References

[1] Kazama J, Torisawa K. Exploiting Wikipedia as external knowledge for named entity recognition. In *Proc. the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, June 2007, pp.698-707.

[2] Ratinov L, Roth D. Design challenges and misconceptions in named entity recognition. In *Proc. the 13th Conference on Computational Natural Language Learning, Association for Computational Linguistics*, June 2009, pp.147-155.

[3] Luo G, Huang X, Lin C Y, Nie Z. Joint entity recognition and disambiguation. In *Proc. the 2015 Conference on Empirical Methods in Natural Language Processing*, September 2015, pp.879-888.

[4] Limsopatham N, Collier N. Bidirectional LSTM for named entity recognition in Twitter messages. In *Proc. the 2nd Workshop on Noisy User-Generated Text*, December 2016, pp.145-152.

[5] Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C. Neural architectures for named entity recognition. In *Proc. the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2016, pp.260-270.

[6] Ma X, Hovy E. End-to-end sequence labeling via bidirectional LSTM-CNNs-CRF. In *Proc. the 54th Annual Meeting of the Association for Computational Linguistics*, August 2016, pp.1064-1074.

[7] Devlin J, Chang M W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2019, pp.4171-4186.

[8] Angeli G, Premkumar M J J, Manning C D. Leveraging linguistic structure for open domain information extraction. In *Proc. the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, July 2015, pp.344-354.

[9] Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka E R, Mitchell T M. Toward an architecture for never-ending language learning. In *Proc. the 24th AAAI Conference on Artificial Intelligence*, July 2010, pp.1306-1313.

[10] Mitchell T, Fredkin E. Never-ending language learning. In *Proc. the 2014 IEEE International Conference on Big Data*, October 2014.

[11] Christensen J, Soderland S, Etzioni O. Semantic role labeling for open information extraction. In *Proc. the 1st NAACL HLT International Workshop on Formalisms and Methodology for Learning by Reading*, June 2010, pp.52-60.

[12] Christensen J, Mausam, Soderland S, Etzioni O. An analysis of open information extraction based on semantic role labeling. In *Proc. the 6th International Conference on Knowledge Capture*, June 2011, pp.113-120.

[13] Santos C N D, Xiang B, Zhou B. Classifying relations by ranking with convolutional neural networks. arXiv:1504.06580, 2015. https://arxiv.org/pdf/1504.06580.pdf, Nov. 2019.

[14] Wang L, Cao Z, de Melo G, Liu Z. Relation classification via multi-level attention CNNs. In *Proc. the 54th Annual Meeting of the Association for Computational Linguistics*, August 2016, pp.1298-1307.

[15] Zeng D, Liu K, Chen Y, Zhao J. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proc. the 2015 Conference on Empirical Methods in Natural Language Processing*, September 2015, pp.1753-1762.

[16] Miwa M, Bansal M. End-to-end relation extraction using LSTMs on sequences and tree structures. arXiv:1601.00770, 2016. https://arxiv.org/abs/1601.00770, Nov. 2019.

[17] Zheng S, Hao Y, Lu D, Bao H, Xu J, Hao H, Xu B. Joint entity and relation extraction based on a hybrid neural network. *Neurocomputing*, 2017, 257: 59-66.

[18] Zheng S, Wang F, Bao H, Hao Y, Zhou P, Xu B. Joint extraction of entities and relations based on a novel tagging scheme. arXiv:1706.05075 June 2017. https://arxiv.org/abs/1706.05075, Nov. 2019.

[19] Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In *Proc. the 27th Annual Conference on Neural Information Processing Systems*, December 2013, pp.2787-2795.

[20] Wang Z, Zhang J, Feng J, Chen Z. Knowledge graph embedding by translating on hyperplanes. In *Proc. the 28th AAAI Conference on Artificial Intelligence*, June 2014, pp.1112-1119.

[21] Lin Y, Liu Z, Sun M, Liu Y, Zhu X. Learning entity and relation embeddings for knowledge graph completion. In *Proc. the 29th AAAI Conference on Artificial Intelligence*, February 2015, pp.2181-2187.

[22] Ji G, He S, Xu L, Liu K, Zhao J. Knowledge graph embedding via dynamic mapping matrix. In *Proc. the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, July 2015, pp.687-696.

[23] Liu Q, Huang Z, Yin Y, Chen E, Xiong H, Su Y, Hu G. EKT: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*. doi:10.1109/TKDE.2019.2924374.

[24] Huang Z, Liu Q, Chen E, Zhao H, Gao M, Wei S, Su Y, Hu G. Question difficulty prediction for READING problems in standard tests. In *Proc. the 31st AAAI Conference on Artificial Intelligence*, February 2017, pp.1352-1359.

[25] Yuan J, Zheng Y, Xie X. Discovering regions of different functions in a city using human mobility and POIs. In *Proc. the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 2012, pp.186-194.

[26] Yuan J, Zheng Y, Zhang C, Xie W, Xie X, Sun G, Huang Y. T-drive: Driving directions based on taxi trajectories. In *Proc. the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, November 2010, pp.99-108.

[27] He Y, Mendis G J, Wei J. Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Transactions on Smart Grid*, 2017, 8(5): 2505-2516.

[28] Layouni M, Tahar S, Hamdi M S. A survey on the application of neural networks in the safety assessment of oil and gas pipelines. In *Proc. the 2014 IEEE Symposium on Computational Intelligence for Engineering Solutions*, December 2014, pp.95-102.

[29] Mohamed A, Hamdi M S, Tahar S. A machine learning approach for big data in oil and gas pipelines. In *Proc. the 3rd International Conference on Future Internet of Things and Cloud*, August 2015, pp.585-590.

[30] Priyadarshy S, Taylor A, Dev A, Venugopal S, Nair G G. Framework for prediction of NPT causes using unstructured reports. In *Proc. the 2017 Offshore Technology Conference*, May 2017.

[31] Hoffimann J, Mao Y, Wesley A, Taylor A. Sequence mining and pattern analysis in drilling reports with deep natural language processing. arXiv:1712.01476, 2017. https://arxiv.org/pdf/1712.01476.pdf, Nov. 2019.

[32] Ma Z, Vajargah A K, Lee H, Darabi H, Castineira D. Applications of machine learning and data mining in SpeedWise® drilling analytics: A case study. In *Proc. the 2018 Abu Dhabi International Petroleum Exhibition & Conference*, November 2018.

[33] Ge J, Li Z, Li T, Qiang B. Petroleum exploration domain ontology-based knowledge integration and sharing system construction. In *Proc. the 2011 International Conference on Network Computing and Information Security*, May 2011, pp.84-88.

[34] Che W, Li Z, Liu T. LTP: A Chinese language technology platform. In *Proc. the 23rd International Conference on Computational Linguistics: Demonstrations*, August 2010, pp.13-16.

[35] Bing Q, Anan L, Ting L. Unsupervised Chinese open entity relation extraction. *Jurnal of Computer Research and Development*, 2015, 52(5): 1029-1035. (in Chinese)

[36] Keller J M, Gray M R, Givens J A. A fuzzy K-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, 1985, 15(4): 580-585.

[37] Pal M. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 2005, 26 (1): 217-222.

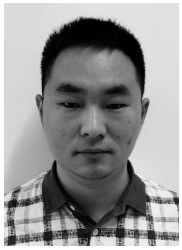
[38] Pal S K, Mitra S. Multilayer perceptron, fuzzy sets, and classification. *IEEE Transactions on Neural Networks*, 1992, 3(5): 683-697.

[39] Friedman J H. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 2001, 29(5): 1189-1232.

Xiang-Guang Zhou received his M.E. degree in oil and gas development from China University of Petroleum, Beijing, in 2002. He is currently a senior engineer in the Department of Computer Application Technology, PetroChina Research Institute of Petroleum Exploration & Development, Beijing. He has published 21 research papers in refereed journals.

Ren-Bin Gong received his Ph.D. degree in geologic engineering from China University of Petroleum (East China), Dongying, in 2007. He is currently a professor in the Department of Computer Application Technology, PetroChina Research Institute of Petroleum Exploration & Development, Beijing. His research interests include both oil and gas industry and computer applied technologies. He has published more than 20 research papers in refereed journals.

Fu-Geng Shi received his Ph.D. degree in oil and gas field development from PetroChina Research Institute of Petroleum Exploration & Development, Beijing, in 1994. He is currently a senior engineer in the Department of Computer Application Technology, PetroChina Research Institute of Petroleum Exploration & Development, Beijing. He has published 15 research papers in refereed journals.



Zhe-Feng Wang received his B.E. and Ph.D. degrees in computer science from the University of Science and Technology of China, Hefei, in 2012 and 2017 respectively. He is currently working as an engineer in the Department of Cloud & AI of Huawei Technologies, Hangzhou. His research interests include natural language processing, machine learning, and social network analysis. He has published more than 10 papers in refereed conference proceedings and journals such as VLDB, KDD, IJCAI, and IEEE Transactions on Knowledge and Data Engineering.