



Enhanced Dynamic Label Allocation for Mathematical Formula Named Entity Recognition in Learning Path Recommendations

Hongchen Liu¹ and Qingchuan Zhang^{1,*}

¹National Engineering Research Centre for Agri-Product Quality Traceability, Beijing Technology and Business University, Beijing 100048, China

Abstract

In the field of natural language processing, Named entity recognition (NER) is an essential task. Mathematical formulas usually contain a large number of terminologies, units of measure and other proprietary knowledge, and the integration of this information into the knowledge graph can significantly enhance the semantic expression ability of the graph. By identifying the named entities in data formulas, the key concepts, entities and relationships between them in the knowledge graph can be extracted, establishing basis for the construction of the knowledge graph and making it easier to interpret and analyse in practical applications. Furthermore, the structured knowledge derived from this process can facilitate personalized learning path recommendations by mapping identified entities to educational resources and prerequisite relationships. Aiming at the problem of insufficient recognition ability of existing models for mathematical formula entities, a mathematical formula named entity

recognition method combining enhanced dynamic allocation of labels is proposed. A mathematical formula entity recognition model consisted of BERT (Bidirectional Encoder Representation from Transformer), BiLSTM (Bidirectional Long Short-term Memory) and Transformer was constructed, namely BERT-formula. The feature representation of deep semantic information is enhanced by adding extra sequences to the original vector representation for splicing at the model input; and the entity label prediction problem is regarded as a one-to-many linear allocation problem, and an auction algorithm is introduced to acquire the optimal allocation result with the smallest cost. Experiments demonstrate that the accuracy of the model prediction on the mathematical formula set is 98.8%, and the F1 value is 98.8%, which is improved by 1.51 and 1.05 percentage points compared with BERT-BiLSTM-CRF. It is evident that the approach performs well on the objective of identifying mathematical formula entities.

Keywords: named entity recognition (NER), mathematics, bidirectional encoder representations from transformer (BERT), deep learning, auction algorithm.



Submitted: 13 December 2024

Accepted: 14 April 2025

Published: 20 May 2025

Vol. 1, No. 1, 2025.

10.62762/FEIR.2024.416675

*Corresponding author:

✉ Qingchuan Zhang

zqc1982@126.com

Citation

Liu, H., & Zhang, Q. (2025). Enhanced Dynamic Label Allocation for Mathematical Formula Named Entity Recognition in Learning Path Recommendations. *Frontiers in Educational Innovation and Research*, 1(1), 10–21.



© 2025 by the Authors. Published by Institute of Central Computation and Knowledge. This is an open access article under the CC BY license (<https://creativecommons.org/licenses/by/4.0/>).

1 Introduction

With the rapid development of information technology, technological breakthroughs in the fields of big data, cloud computing, artificial intelligence and other fields continue to emerge, which has brought about profound changes in all walks of life. In the field of operations research, mathematical modelling and optimization analysis methods play a crucial role in solving complex problems. Therefore, for mathematical formulas in operations research, we need to find it and distinguish it, and identify the named entities in the formulas accurately, which helps to improve the efficiency and accuracy of operations research.

Latex is a coding language for the representation of formulas in operations research, and it is also involved in other industries or professions related to mathematical formulas. Nonetheless, the topic of named entity recognition in operations research has seen comparatively little investigation and effort, and no sizeable and accessible datasets have been compiled. As a result, dealing with current problems in the field of formulas in operations research with existing techniques achieves mediocre results due to the lack of sufficient reference and experimental data.

Named Entity Recognition is a fundamental task for several natural language processing applications such as knowledge base question and answer systems, machine translation, information retrieval, sentiment analysis, and knowledge mapping. Finding specific entities, such as names of individuals, locations, organizations, etc. from a piece of text, and distinguish them from non-entities is the main objective of NER. Named entity recognition of latex texts makes further research possible. Accordingly, this paper constructs a dataset from the text of operations research courses in schools and related digital resources, and conducts research on entity recognition in operations research on the basis of the dataset.

This paper proposes a mathematical formula named entity recognition model consisting of BERT, BiLSTM and Transformer, for addressing the above issues. The BERT model is used to obtain the vector representation of each character, which is spliced with the vector representation of a randomly initialized instance query, and then jointly fed into BERT for encoding, and then the query semantics are enhanced by using One-Way Self-Attention so that the query can be modelled in terms of its connections with each other. This is followed by feature extraction through the BiLSTM

layer and transformer layer, and finally the final predicted labelling results are obtained by finding the optimal allocation by finding the minimum cost matrix for the allocation problem.

2 Related Work

The research progress in named entity recognition techniques can be divided into the following stages:

The first stage is the use of dictionary and rule based approach. Firstly, a candidate dictionary is obtained based on statistical analysis, and then manual screening is used while extracting the important terms in the domain. Using the a priori knowledge of the lexicon, potential entities in the sentence are matched, after which they are filtered according to rules. Rule-based approaches tend to rely too much on manually defined rules and templates, and thus may have limitations in their coverage for complex linguistic expressions and diverse inputs. Moreover, manually constructed rules are subjective and prone to bias and errors.

The second stage is the statistical machine learning based approach, in which researchers begin to try to use statistical models for named entity recognition. For instance, in a research finished by Yu et al. [1], they utilized Hidden Markov Model (HMM) for the recognition of Chinese named entities; Huang et al. [2] suggested an approach combining Support Vector Machine (SVM) with transformation-based error-driven learning for biological entity recognition; A named entity recognition technique based on conditional random fields was proposed by Feng et al. [3]. Statistical machine learning based methods no longer rely on hand-constructed tedious rules, however, they need a large number of training sets with clear labels, which still takes a lot of effort and resources.

The third stage is deep learning based named entity recognition. With the theory and application of deep learning gradually coming into people's attention, in addition to the improvement of algorithms and computer performance, the depth and width of neural networks [4] are also increasing. As a result, many neural network structures have emerged that are particularly well known today, the Convolutional Neural Network (CNN) [5], the Recurrent Neural Network (RNN) [6], the Long Short Term Memory Network (LSTM), and even more intricate deep learning models like BERT (Bidirectional Encoder Representation from Transformer) [7]. Deep learning

models can learn and extract features from data on their own, in contrast to standard machine learning techniques that require feature extraction by hand. Deep learning greatly saves the human resources required for feature fusion, and deep learning models can automatically learn complex language patterns with strong generalization ability, which enables them to be better applied to practical tasks. For example, unidirectional Long Short Term Memory (LSTM) networks [8] are widely used in NER tasks due to their strong sequential feature extraction ability and are often combined with CRF (LSTM-CRF [9]) to achieve better recognition results. However, since unidirectional LSTM networks is limited to extracting unidirectional text features, Lample et al. [10] then proposed a Bidirectional LSTM (BiLSTM) network on this basis to obtain global contextual deep features, and then combined with CRF to form a BiLSTM-CRF neural network, it enhances the model's effect even further, and since then the model has gradually become the mainstream model for deep learning to solve NER problems in various fields. For example, Zhou et al. [11] promoted knowledge mining of ancient Chinese medicine books by extract text features using the BiLSTM-CRF method; Cheng et al. [12] applied the BiLSTM-CRF model to the field of ancient Chinese literature, and realized the processing tasks of automatic sentence breakage, automatic word division, lexical annotation and other processing tasks of the ancient Chinese information, and achieved very good results.

Meanwhile there are many scholars who have improved and innovated the BiLSTM-CRF model. For example, Huang et al. [13] introduced an external cybersecurity lexicon to enhance the features of cybersecurity texts based on the BiLSTM-CRF model, and obtained a favorable outcome on the cybersecurity dataset. On the field of agriculture, Zhou et al. [14] firstly processed the long text of the dataset into short text, and then input it into the ERNIE model for encoding to get this representation that preserves semantic associations, and subsequently enter it into BiLSTM-CRF to address the issues of low efficiency of the soil fertility named entity recognition method and poor text processing effect. Regarding the study of Chinese, Li et al. [15] introduced Hybrid Attention mechanism (Hybrid Attention) into BiLSTM-CRF model to achieve good semantic analysis ability and accurately represent the negation semantics in a sentence.

The Iterative Expanded Convolutional Neural Network

(IDCNN), which can effectively extract local features across a wide acceptance domain, was also applied for Named Entity Recognition for the first time by Strubell et al. [16]. In order to solve the named entity identification issue in electronic medical records, Chen et al. [17] presented an attention mechanism based on the IDCNN-CRF model, and the special step-size of the inflated convolution can extract the text features more accurately with excellent recognition results.

To improve the word vectors' semantic representation, scholars have proposed pre-trained language models. Peters et al. [18] proposed ELMo model based on BiLSTM structure, which is able to extract bi-directional textual features. Radford et al. [19] proposed GPT (Generative Pre-trained Transformer) model, which is able to capture more distant semantic features. is able to capture more distant semantic information, but because it is a unidirectional model, it is unable to obtain global contextual information. Therefore, Devlin et al. [8] from Google team proposed the BERT model with bidirectional Transformer encoder structure. This boosts its performance on named entity recognition tests and further refines the semantic representation of word vectors. For example, Li et al. [20] utilized the BERT-CRF model to the joint extraction of maize breeding entity relationships, which provides an effective data base for the construction of maize breeding knowledge graph and other downstream tasks; Zheng et al. [21] utilized the BiLSTM-CRF model based on BERT for web content monitoring to identify sensitive words and variants, and the recognition effect is improved compared with other models; Yu et al. [22] proposed an ancient poetic place name recognition model, referred to as DABERT-CRF, using a data enhancement method while integrating BERT-CRF, as a way to promote further research on Chinese classical literature; Li et al. [23] used BERT and BiLSTM, combined with a bilinear attention mechanism, to successfully improve the semantic information and attain favorable outcomes in the Chinese recognition of medically named entities. Good results have been achieved on it.

The NER task was redefined as a machine-reading task in recent years by Luo et al. [24], Mengge et al. [25] and Zheng et al. [26], who demonstrated strong performance on both flat and nested datasets. With the goal to extract entities, they create type-specific queries based on external knowledge and treat phrases as contexts. They create PER-specific queries in natural language form, for instance, for the statement "U.S.

President Donald Trump is enjoying his vacation in Miami" in order to extract PER entities like "U.S. President" and "Donald Trump". However, only one type of entity can be extracted for each inference because searches are type-specific. This approach overlooks the inherent relationships between different entity types in addition to producing ineffective predictions. Furthermore, Type-specific searches are manually constructed using external knowledge, which makes realistic scenarios using hundreds of entity types difficult.

3 Methodology: A mathematical formula named entity recognition method combining enhanced dynamic allocation of labels

Based on the similarities between sequences of mathematical formulas and textual information, in everyday use mathematical formulas are often in the form of latex encoding, thus allowing useful features to be learned from the data in a textual form. BERT-formula is a hybrid model architecture based on the BERT model encoding. A vector representation of each character is first computed, spliced with a randomly initialized vector representation of the instance query, and then jointly fed into BERT for encoding, followed by the use of unidirectional Self-Attention, which allows the queries to model connections with each other, enhancing the query semantics [27]. This is followed by feature extraction through BiLSTM layer and transformer layer to improve the model's generalization capacity even more. Finally, a dynamic label assignment mechanism is designed to ascertain the best possible result for the assignment, and label assignment is regarded as a one-to-many Linear Assignment Problem (LAP) (Burkard and Cela, 1999) [28]. The final predicted labelling outcome is obtained by finding the minimum cost matrix of the allocation problem and finding the optimal allocation.

Figure 1 displays the model's framework as suggested in this paper. There are three components to it: encoder, entity prediction and dynamic label assignment, where entity classification and entity localization are the two subtasks of entity prediction. The input of the encoder part comes from textual information as well as instance queries that can learn global semantic information, and the vector representation is obtained by using embedding to extract rich syntactic and semantic elements; the entity prediction part mainly accomplishes the boundary prediction of the entity as well as the category

prediction of the entity, and if more than one prediction for the same entity occurs, The one chosen to be kept is the one with the highest probability value; the dynamic label entity assignment mainly According to the allocation cost generated in the previous part to form the Cost matrix, and then further use the algorithm of the linear allocation problem to calculate the label allocation matrix with the minimum cost, so as to achieve the instance query and the allocation of entity labels, and get the prediction results of each entity to complete the recognition task.

To indicate an example of training, we use (X, Y) , where X is a sentence with N words, $Y = \{ \langle X_k^l, X_k^r, X_k^t \rangle \}_{k=0}^{G-1}$ and the three represent the index of the left boundary of the k -th entity, the index of the right boundary, and the index of the entity type, respectively. In our study, $M(> G)$ globally learnable instance queries are set up, it extracts an entity from the phrase in each case. During training, they can independently learn query semantics and are initialized at random. Therefore, the challenge involves using the learnable instance inquiry I to extract entity Y from an input sentence X .

3.1 Encoder

Two components make up the model's input: an instance query of length M and the text of a latex mathematical formula of length N . The Encoder is responsible for stitching them together into a sequence and encoding them at the same time. The instance query of length M is a randomly generated sequence of a fixed-length segment, which is composed without the aid of external knowledge and learns deep semantic information between sentences.

Firstly, we start with the computation of embedding, with the help of Bert embedding we compute the Token embedding, Position embedding and Type embedding of the sequence, after that we stitch the two embedding information to get the E_{token} , $E_{position}$, E_{type} .

$$\begin{aligned} E_{token} &= \text{Concat}(V, I) \\ E_{position} &= \text{Concat}(P^w, P^q) \\ E_{type} &= \text{Concat}([U^w]^N, [U^q]^M) \end{aligned} \quad (1)$$

where V denotes the Token embedding of the word sequence, I denotes the vector representation of the instance query, P^w and P^q denote the learnable positional embedding of the text sequence and the instance query sequence, U^w and U^q denote the type embedding of the text and the type embedding of the

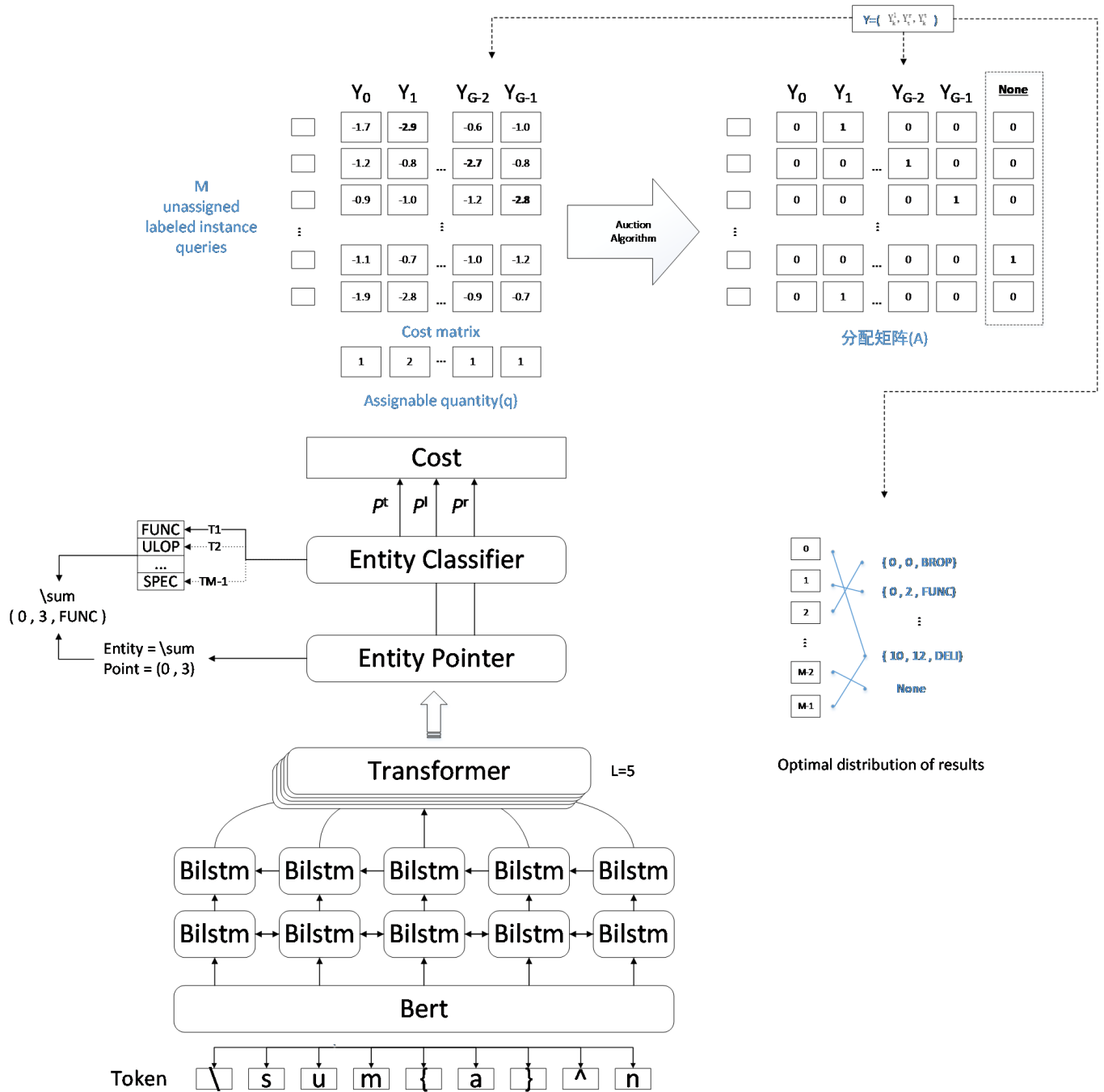


Figure 1. Structure of the model.

instance query respectively, $[.]^N$ and $[.]^M$ denotes the repetition of the N -th and M -th times.

The further input to the encoder can thus be represented as:

$$H_0 = E_{\text{token}} + E_{\text{position}} + E_{\text{type}} \in \mathbb{R}^{(N+M) \times h}$$

3.1.1 One-Way Self-Attention

Sentences can communicate with all instance queries thanks to common self-attention. Randomly produced instance queries might so alter the sentence's encoding

and ruin its semantics. In order to keep the semantics of the sentence relatively independent from the instance queries, we replace the unidirectional form of self-attention in BERT with a version that maintains the sentence semantics largely independent from the instance queries.

$$\begin{aligned} OW-SA(H) &= \alpha H W_v \\ \alpha &= \text{softmax} \left(\frac{H W_q (H W_k)^T}{\sqrt{h}} + M \right) \end{aligned} \quad (2)$$

where W_q, W_k, W_v are the parameters of the weight matrix, $M \in \{0, -\infty\}^{(N+M) \times (N+M)}$ is the mask matrix representing the attention scores. The components that are set to 0 are reserved units, whereas the elements that are set to $-\infty$ are removed units. The top-right sub-matrix in our approach is a full matrix of size $N \times M$ with all other members set to 0. This prevents instance queries from participating in sentence encoding. Furthermore, the self-attention between instance queries can improve their query semantics and model the connections between them.

Following BERT encoding, we use two bidirectional LSTM layers and L (experimentally set to 5) extra transformer layers to further encode the phrases at the character level. Finally, we will split $H \in \mathbb{R}^{(N+M) \times h}$ into two parts: sentence encoding $H^w \in \mathbb{R}^{N \times h}$ and instance query encoding $H^q \in \mathbb{R}^{M \times h}$.

3.2 Entity Prediction

One entity from a phrase can be predicted by each instance query, and a maximum of M entities can be predicted concurrently with M instance inquiries. One way to think of entity prediction is as a combination of boundary and category prediction. We have designed entity pointers and entity classifiers from different perspectives.

3.2.1 Entity Pointer

Firstly, we use two linear layers to interact with each character in the phrase for the i -th instance query H_i^q . The fused representation of the i -th instance query and the j -th character can be computed as:

$$S_{ij}^\delta = \text{ReLU}(H_i^q W_\delta^q + H_j^w W_\delta^w) \quad (3)$$

where $\delta \in \{l, r\}$ represents the left and right boundaries, $W_\delta^q, W_\delta^w \in \mathbb{R}^{h \times h}$ are trainable projection parameters. Next, we determine the possibility that the sentence's j -th word represents a left or right boundary.

$$P_{ij}^\delta = \text{sigmoid}(S_{ij}^\delta W_\delta + b_\delta) \quad (4)$$

where $W_\delta \in \mathbb{R}^h$ and $b_\delta \in \mathbb{R}^h$ are learnable parameters.

3.2.2 Entity Classifier

Information about entity boundaries is helpful for classifying entities. We use $P_i^\delta = [P_i^{\delta_0}, P_i^{\delta_1}, \dots, P_i^{\delta_{N-1}}]$, $\delta \in \{l, r\}$ to evaluate each word and relate it to the instance query. It is possible to calculate the

boundary-aware representation of the i -th instance query as:

$$S_i^\delta = \text{ReLU}([H_i^q W_t^q; P_i^l H^w; P_i^r H^w]) \quad (5)$$

where $W_t^q \in \mathbb{R}^{h \times k}$ is a learnable parameter. The probability that the entity that the i -th instance query is attempting to query falls into category c can then be obtained:

$$P_{ic}^t = \frac{\exp(S_i^t W_t^c + b_t^c)}{\sum_{c' \in \mathcal{E}} \exp(S_i^t W_t^{c'} + b_t^{c'})} \quad (6)$$

where $W_t^{c'} \in \mathbb{R}^h$ and $b_t^{c'}$ are learnable parameters.

Finally, the entity predicted by the i -th instance query is $T_i = (T_i^l, T_i^r, T_i^t)$. $T_i^l = \arg \max_j (P_{ij}^l)$, $T_i^r = \arg \max_j (P_{ij}^r)$ is the left and right boundaries, and $T_i^t = \arg \max_c (P_{ic}^t)$ is the entity type.

We extract entities in parallel by performing entity classification and entity localization for every instance query. If multiple instance queries predict different types when locating the same entity, then we keep the one with the highest classification probability as the prediction.

3.3 Dynamic Label Assignment

Since instance queries are implicit (not in natural language form), we are unable to assign optimal entities to them in advance. In order to address this issue, we are going to assign labels to instance queries during training dynamically. Label assignment is specifically thought of as a linear assignment problem. Any entity can be assigned to any instance query, and the cost incurred may vary due to instance query assignment. We define the cost of assigning the k th entity ($Y_k = \langle Y_k^l, Y_k^r, Y_k^t \rangle$) to the i th instance query as:

$$\text{Cost}_{ik} = -(P_{iY_k^t}^t + P_{iY_k^l}^l + P_{iY_k^r}^r) \quad (7)$$

where Y_k^l, Y_k^r, Y_k^t indicates the entity type of the k -th entity, the index of the left boundary and the right boundary. In order to allocate as many entities as possible, A maximum of one entity per query and a maximum of one query per entity must be allocated, which ensures that the total cost of allocation is minimized. Nevertheless, Numerous instance queries are not allocated to the best possible entities, and the one-to-one rule fully utilizes instance queries. In order to enable one entity to be assigned to more than one instance query, we therefore extend the traditional LAP (Linear Assignment Problem) to one-to-many.

The optimization objective of this one-to-many LAP is explained as:

$$\begin{aligned} \min \sum_{i=0}^{M-1} \sum_{k=0}^{G-1} A_{ik} \text{Cost}_{ik} \quad \text{s.t.} \\ \sum_k A_{ik} \leq 1, \sum_i A_{ik} = q_k, \forall i, k, A_{ik} \in \{0, 1\} \end{aligned} \quad (8)$$

where $A \in \{0, 1\}^{M \times G}$ is the matrix of allocation, G indicates the number of entities, $A_{ik} = 1$ denotes the k -th entity allocated to the i -th instance query. q_k indicates the assignable quantity of the k -th entity, $Q = \sum_k q_k$ represents all entities' total assignable quantity. The assignable numbers of each entity type are balanced in our experiments.

Then we used the Auction Algorithm to settle the entity allocation problem by generating the label allocation matrix with the lowest possible overall cost. However, there are more instance inquiries than there are entity labels available for allocation ($M > Q$). Consequently, certain instance queries won't be assigned to any entity label. We expand the allocation matrix by one column and assign them "None" labels. The vector a of the new column is set up as follows:

$$a = \begin{cases} 0, & \sum_k A_{ik} = 1 \\ 1, & \sum_k A_{ik} = 0 \end{cases} \quad (9)$$

On account of new allocation matrix $\hat{A} \in \{0, 1\}^{M \times (G+1)}$, we are able to obtain the labels of M instance queries $\hat{Y} = Y.\text{indexby}(\pi^*)$, where $\pi^* = \arg \max_{\text{dim}=1}(\hat{A})$ is the index vector of labels of instance queries in the optimal allocation case.

3.4 Training objectives

After the above steps, we as well compute the entity prediction results for M instance queries and get their labels when the total allocation cost is minimum. We specify the classification loss and boundary loss for training model. We utilize the binary cross entropy function to get the loss values for left and right border prediction:

$$L_b = - \sum_{\delta \in \{l, r\}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [\hat{Y}_i^c = c] \log P_{ic}^t + [\hat{Y}_i^\delta \neq c] \log(1 - P_{ic}^t) \quad (10)$$

We employ the cross entropy function for entity categorization to calculate the loss value:

$$L_b = - \sum_{i=0}^{M-1} \sum_{c \in \epsilon} [\hat{Y}_i^t = c] \log P_{ic}^t \quad (11)$$

where $1[\omega]$ represents an indicator function that ω takes 1 when true and 0 otherwise.

Following Al-Rfou et al. [29] and Carion et al. [30], after every character level transformer layer, we add an entity pointer and an entity classifier, allowing us to obtain two loss values at each layer. As a result, the overall loss on training set D can be described as:

$$L = \sum_D \sum_{\tau=1}^L L_t^\tau + L_b^\tau \quad (12)$$

where L_t^τ, L_b^τ are the classification loss and boundary loss of the τ layer, respectively. In order to predict, we solely use entity prediction at the last layer for prediction.

4 Research on Learning Path Recommendation Based on Knowledge Graph

Knowledge graphs (KGs) serve as a pedagogical and computational scaffold for structuring fragmented knowledge domains, enabling the generation of logically coherent learning paths that respect intrinsic educational dependencies such as prerequisite sequences, hierarchical inclusions, and conceptual analogies. Unlike traditional methods, which often prioritize isolated content delivery, KGs formalize knowledge topology through nodes (e.g., "Gradient Descent") and edges (e.g., "is_prerequisite_for"), allowing systematic traversal from a target node (e.g., "Convolutional Neural Networks") to prerequisite foundations (e.g., "Linear Algebra" or "Image Processing"). This structural rigor ensures learning continuity by mandating that no critical intermediate concepts are omitted, thereby preserving the integrity of pedagogical logic. For instance, omitting "Backpropagation" in a path toward "Deep Reinforcement Learning" would violate prerequisite dependencies, undermining learning efficacy. To operationalize this, a hybrid recommendation strategy is proposed, integrating inclusion-relation prioritization and multi-criteria node ranking. First, inclusion relations (e.g., "Machine Learning Fundamentals" \supset "Supervised Learning") are extracted using subgraph isolation:

$$G_{\text{include}} = \{v \in V \mid \exists e(v, v_{\text{target}}) r_{\text{include}} \in E\} \quad (13)$$

where G_{include} denotes the subgraph of nodes directly containing or contained by the target node v_{target} . Subsequently, candidate nodes are ranked by a composite metric balancing association

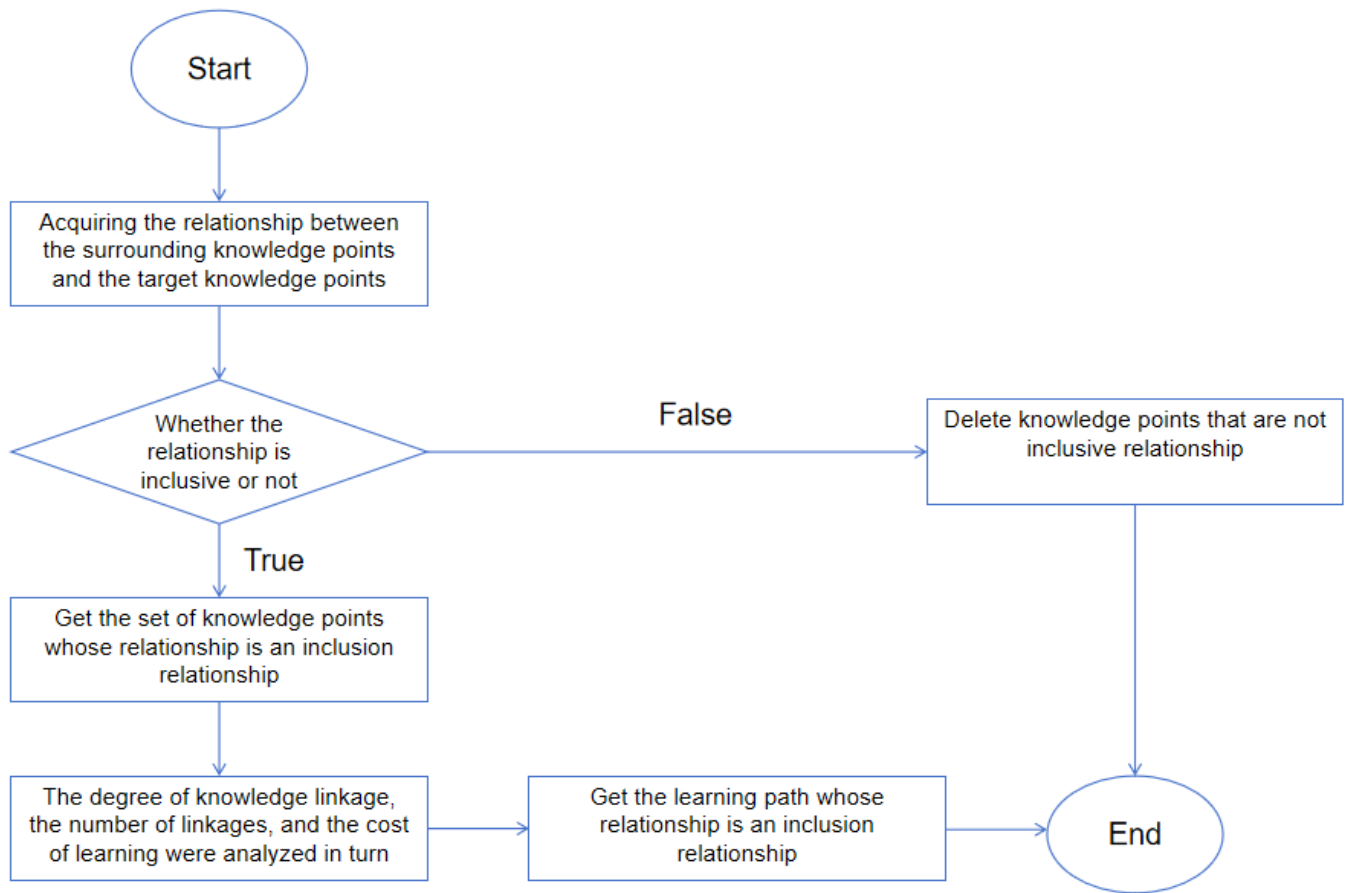


Figure 2. Knowledge graph based learning path recommendation scheme map.

strength and learning cost. Association strength is quantified via a weighted combination of inverse shortest-path distance $\text{Distance}(v, v_{\text{target}})$ and node centrality $\text{Degree}(v)$:

$$\text{Score}_{\text{association}}(v) = \alpha \cdot \frac{1}{\text{Distance}(v, v_{\text{target}})} + \beta \cdot \text{Degree}(v) \quad (14)$$

where α and β calibrated through learner feedback or A/B testing. Nodes with higher scores are prioritized, while ties are resolved using learning cost $C(v)$, modeled as a function of historical learner engagement (t_{avg}) and resource complexity $RC(v)$:

$$C(v) = \gamma \cdot t_{\text{avg}} + (1 - \gamma) \cdot RC(v) \quad (15)$$

where γ adjusts the weight of temporal versus cognitive load. This framework aligns with graph-based recommendation paradigms such as Zhang et al. [31] graph convolutional network (GCN) approach, which leverages node embeddings to capture latent educational dependencies, and Zheng et al. [32] ant colony optimization variant, dynamically balancing exploration of novel concepts and exploitation of known pathways. Furthermore, Shi

et al. [33] validate the necessity of multidimensional KGs, demonstrating that integrating node centrality (e.g., PageRank scores) and learner-specific factors (e.g., proficiency levels) enhances path personalization for learners, as measured by retention metrics. The methodology's robustness is further reinforced by its compatibility with ontology-enriched KGs, where named entity recognition (NER) refines node categorization—for example, distinguishing "Bayesian Inference" as a theoretical versus applied concept—ensuring granular alignment with curricular taxonomies [34]. Figure 2 shows the general process of learning path recommendation based on knowledge graph.

5 Experiments

In this paper, we use manual annotation to extract more than four thousand latex mathematical formulas from the text of about 100,000 words of operations research textbook, which contains more than 10,000 entities and annotate the latex formulas, and there are five types of entity categories defined, namely: Function operator, Unary logical operator, Binary relation operator, Delimiter and Special operator.

Table 1 shows certain entities' pertinent descriptions and examples.

5.1 Labeling standards

In the named entity recognition task, there are two commonly used annotation approaches, which are BIO annotation method and BIOES annotation method. In the BIO annotation approach, B-begin denotes the entity's first character, I-inside its middle character, and O-outside its character that is unrelated to any of the other entities. (which can be interpreted as the character that doesn't exist in any entity). When using the BIOES annotation method, the B stands for the entity's beginning, the I for its middle, the O for the character unrelated to the entity, the E-end for its end, and the S-single for the single character, which is an entity in and of itself. This article uses the BIO annotation method.

5.2 Assessment indicators

Three indicators—precision (P), recall (R), and F1 value (F1)—are used in this research as the criteria for assessing the model's efficacy. P is how many true positive samples are there in the samples predicted to be positive; and R is how many positive samples are predicted correctly with respect to the original positive samples. When we use these two metrics to judge the merits of a model, one would often hope that the values of both precision and recall are at a high level, but in practice this is often not the case because the two metrics are contradictory in some cases. In order to give a more comprehensive judgment of the model, the reconciled average of precision and recall, F1, is introduced. P, R and F1 are calculated as follows:

$$P = \frac{T_p}{T_p + F_p} \times 100\% \quad (16)$$

$$R = \frac{T_p}{T_p + F_N} \times 100\% \quad (17)$$

$$F1 = 2 \times \frac{P \times R}{P + R} \times 100\% \quad (18)$$

where T_p is the number of entities that the model properly identified, F_p is the number of unrelated entities that the model identified, and F_N indicates the quantity of related entities that the model failed to recognize.

5.3 Experimental environment and model parameter settings

The experiments are based on the torch framework to build neural network models, and the detailed

configuration of the experimental environment is shown in Table 2.

The main training parameters involved in the experiment are shown in Table 3. The quantity of Transformer layers used for the BERT model is 12, hidden layer dimension 768, number of head 12, optimizer AdamW, learning rate 0.00002, Lstm_dim 384, Batch_size 4, dropout 0.4, gradient_clip 1, weight_decay 0.1, max_len 512, epochs 50.

5.4 Experimental results and analysis

To confirm the capability of the proposed model (BERT-formula) for entity recognition of mathematical formulas in operations research, it was compared with four models, BiLSTM, BiLSTM-CRF and Bert-BiLSTM-CRF, in the same experimental environment in terms of three indexes: accuracy, recall, and F1 value. The results of the experiments are shown in Table 4.

From Table 4, it is evident for us to test the effect of each model for the entity recognition of operational research formulas. It can be seen that the capability of BiLSTM alone is very limited, while after combining with CRF, The model's recognition impact has significantly improved. The BERT-BiLSTM-CRF model has a large improvement in the recognition effect due to the fact that BERT has the capability of extracting local and contextual features, which produces feature vectors with a more accurate representation.

6 Conclusion

This research investigates named entity recognition of mathematical formulas based on text in the field of operations research, and proposes a model for named entity recognition of mathematical formulas, BERT-formula. an efficient feature representation of mathematical formulas using embedding is spliced with a vector representation of randomly initialized instance queries, which are then jointly input to BERT for encoding, and then encoded using one-way Self-Attention, which allows the queries to model connections with each other and enhances the query semantics. After that, feature extraction is performed through BiLSTM layer and transformer layer, and finally the final predicted labeling results are obtained by finding the optimal allocation by finding the minimum cost matrix for the allocation problem. The experimental findings demonstrate that, in comparison to the conventional NER technique, the inference speed of the approach presented in this study is significantly enhanced, and it is also superior in terms of recognition

Table 1. Representation and examples of entities.

Type of entity	Notation	Description	Example
Function operator	FUNC	Minimum, maximum	\min, \max, \dots
Unary logical operator	ULOP	Include, less than, more than...	\in, \leq, \geq, \dots
Binary relation operator	BROP	Intersection, union ...	\land, \vee, \dots
Delimiter	DELI	Array...	$\begin{pmatrix} \dots \end{pmatrix}, \dots$
Special operator	SPEC	Root sign, fraction	$\sqrt{}, \frac{}{}, \dots$

Table 2. Experiment environment.

Environment	Configuration
OS	Windows 11
Hardware	CPU
	Intel(R) Core(TM) i7-12700H 2.30 GHz
	GPU
	GeForce RTX 3060
	Memory
	32 GB
Software	Python
	Python 3.6
	Pytorch
	torch 1.9.1

Table 3. Model parameters.

Environment	Configuration
Transformer	12
Hidden_size	768
Number of head	12
Optimizer	AdamW
Learning_rate	0.00002
Lstm_size	384
Drop_out	0.4
Gradient_clip	1
Weight_decay	0.1
Batch_size	4

Table 4. Comparison of named entity recognition results of different models.

Methods	Accuracy	Recall	F1
BiLSTM	36.93	37.60	37.26
BiLSTM-CRF	70.41	74.86	72.57
BERT-BiLSTM-CRF	97.37	98.28	97.82
Ours	98.88	98.88	98.87

results. The identified formula entities can further support adaptive learning path recommendations by mapping domain-specific knowledge components to targeted educational resources. The named entity recognition of operations research formulas achieved in this research establishes a strong basis for downstream NLP tasks in the field of operations research, and the example queries in this paper do not rely on external knowledge, but throughout the training phase, acquire the query semantics associated with the entity type and location, which saves a lot

of manual consumption; the feature of not relying on the knowledge related to a specific domain also makes the method can be easily applied to other domains. At the same time, for the problem of fewer datasets of text labeling of mathematical formulas in the field of operations research, a dataset of operations research latex formulas is constructed, which contains nearly five thousand data, and it is anticipated that it will contribute in some way to the advancement of operations research. Subsequently, the size of the dataset will be further expanded to further verify the transferability of the model and apply the model to other fields.

Data Availability Statement

The source code is available at <https://github.com/Ctrius/formula>.

Funding

This work was supported in part by the Project of Construction and Support for high-level Innovative Teams of Beijing Municipal Institutions under Grant BPHR20220104; in part by the Beijing Scholars Program under Grant 099; in part by the IFLYTEK University Intelligent Teaching Innovation Research Special Project under Grant 2022XF055.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

- [1] Yu, H. K., Zhang, H. P., Liu, Q., Lu, X. Q., & Shi, S. C. (2006). Chinese named entity identification using cascaded hidden Markov model. *Tongxin Xuebao/Journal on Communications*, 27(2), 87-94.
- [2] Huang, H. W. (2009). *SVM combined with error-driven learning for biological entity recognition*. National University of Defense Technology.

- [3] Feng, Y., Yu, H., Sun, G., & Zhao, Y. (2016). Domain-specific term recognition method based on word embedding and conditional random field. *Journal of Computer Applications*, 36(11), 3146-3151.
- [4] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504-507. [CrossRef]
- [5] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. [CrossRef]
- [6] Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In *Interspeech* (Vol. 2, No. 3, pp. 1045-1048).
- [7] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780. [CrossRef]
- [8] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186). [CrossRef]
- [9] Hammerton, J. (2003). Named entity recognition with long short-term memory. *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL*, 172-175.
- [10] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural Architectures for Named Entity Recognition. In *Proceedings of NAACL-HLT* (pp. 260-270). [CrossRef]
- [11] Zhou, J. W., Wang, K., Wu, Y. L., et al.. (2024). Research on Named Entity Recognition of Shen Nong's Materia Medica Based on BiLSTM-CRF. *Journal of Chengdu University of Traditional Chinese Medicine*, 47(03), 54-59. [CrossRef]
- [12] Cheng, N., Li, B., Ge, S., Hao, X., & Feng, M. (2020). A joint model of automatic sentence segmentation and lexical analysis for ancient Chinese based on BiLSTM-CRF model. *Journal of Chinese Information Processing*, 34(4), 1-9.
- [13] Huang, Z. Y., Yu, Y. N., Lin, R. M., et al.. (2024). Knowledge graph construction for network security base on modified BiLSTM-CRF. *Modern Electronics Technique*, 47(06), 15-21. [CrossRef]
- [14] Zhou, L. L., Chen, L., Ji, F., et al.. (2023). ERNIE-BiLSTM-CRF Model-based entity recognition study in soil fertility. *Horticulture and Seed*, 43(09), 97-101. [CrossRef]
- [15] Li, J., Lyu, G., Li, R., et al.. (2023). Chinese Negative Semantic Representation and Annotation Combined with Hybrid Attention Mechanism and BiLSTM-CRF. *Computer Engineering and Applications*, 59(09), 167-175.
- [16] Strubell, E., Verga, P., Belanger, D., et al.. (2017). Fast and accurate entity recognition with iterated dilated convolutions. *Proceedings of EMNLP*, 2670-2680.
- [17] Chen, T. Y., & Feng, S. (2022). Research on named entity recognition method and model stability of electronic medical record based on IDCNN+CRF and attention mechanism. *China Digital Medicine*, 17(11), 1-5.
- [18] Peters, M. E., Neumann, M., Iyyer, M., et al.. (2018). Deep contextualized word representations. *Proceedings of NAACL-HLT*, 2227-2237.
- [19] Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. (2018). Improving language understanding by generative pre-training. *OpenAI preprint*.
- [20] Li, S., & Pang, W. (2023). Joint Extraction Method of Entity and Relation in Maize Breeding Based on BERT-CRF and Word Embedding. *Transactions of the Chinese Society of Agricultural Machinery*, 1-16. [CrossRef]
- [21] Zheng, X., Li, B., Feng, Z., et al.. (2023). Entity Recognition of Network Sensitive Words and Variants Based on BERT-BiLSTM-CRF. *Computer and Digital Engineering*, 51(07), 1585-1589.
- [22] Yu, X., & Chang, E. (2023). Automatic Recognition of Place Names in Ancient PoetryBased on DA-BERT-CRF Models: Taking the AncientPoetries of Nanjing as an Example. *Library Journal*, 42(10), 87-94+73. [CrossRef]
- [23] Li, X., Feng, J., Meng, Y., Han, Q., Wu, F., & Li, J. (2020). A unified MRC framework for named entity recognition. *Proceedings of ACL*, 5849-5859.
- [24] Luo, X., Li, T., & Jia, Z. (2024). Chinese medical named entity recognition based on self-attention mechanism and lexicon enhancement. *Journal of Computer Applications*, 44(2), 385-392.
- [25] Xue, M., Yu, B., Zhang, Z., Liu, T., Zhang, Y., & Wang, B. (2020). Coarse-toFine Pre-training for Named Entity Recognition. *Proceedings of EMNLP*, 6345-6354.
- [26] Zheng, H., Qin, B., & Xu, M. (2021, January). Chinese medical named entity recognition using crf-mt-adapt and ner-mrc. In *2021 2nd International Conference on Computing and Data Science (CDS)* (pp. 362-365). IEEE. [CrossRef]
- [27] Shen, Y., Wang, X., Tan, Z., Xu, G., Xie, P., Huang, F., ... & Zhuang, Y. (2022). Parallel instance query network for named entity recognition. *arXiv preprint arXiv:2203.10545*. [CrossRef]
- [28] Burkard, R. E., & Cela, E. (1999). Linear assignment problems and extensions. In *Handbook of combinatorial optimization: Supplement volume A* (pp. 75-149). Boston, MA: Springer US. [CrossRef]
- [29] Al-Rfou, R., Choe, D., Constant, N., Guo, M., & Jones, L. (2019, July). Character-level language modeling with deeper self-attention. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp.

- 3159-3166). [CrossRef]
- [30] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, August). End-to-end object detection with transformers. In *European conference on computer vision* (pp. 213-229). Cham: Springer International Publishing. [CrossRef]
- [31] Zhang, X., Liu, S., & Wang, H. (2023). Personalized learning path recommendation for e-learning based on knowledge graph and graph convolutional network. *International journal of software engineering and knowledge engineering*, 33(01), 109-131. [CrossRef]
- [32] Zheng, Y., Wang, D., Zhang, J., Li, Y., Xu, Y., Zhao, Y., & Zheng, Y. (2024). A unified framework for personalized learning pathway recommendation in e-learning contexts. *Education and Information Technologies*, 1-38. [CrossRef]
- [33] Shi, D., Wang, T., Xing, H., & Xu, H. (2020). A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning. *Knowledge-Based Systems*, 195, 105618. [CrossRef]
- [34] Duan, S., Chen, K., Yang, Y., & Shi, S. (2023, August). Research on personalized learning recommendation based on subject knowledge graphs and learner portraits. In *International Conference on Computer Science and Educational Informatization* (pp. 367-374). Singapore: Springer Nature Singapore. [CrossRef]



Hongchen Liu received his B.E. degree in Information Management and Information System from Beijing Technology and Business University in 2022. In the same year, he was admitted by Management Science and Engineering program. Nowadays, He is a student in the School of Computer and Artificial Intelligence and at the National Engineering Research Center for Agricultural Product Quality Traceability, Beijing Technology and Business University. His research interests include big data mining and natural language processing. (Email: Lhongchen2000@163.com)



Qingchuan Zhang received a B.E. degree and a M.E. degree in Computer Science and Computer Application Technology from Hebei University of Science and Technology in 2004 and 2008, respectively. He received a Ph.D. degree in Computer Application Technology from University of Science and Technology Beijing in 2013. Nowadays, he is an assistant professor at the National Engineering Research Center for Agricultural Product Quality Traceability, Beijing Technology and Business University. His research interests include big data mining, natural language processing, and semantic computing. (Email: zhangqingchuan@bttbu.edu.cn)