Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 5, No. 2, 2023

Chinese E-commerce NER Using RoBERTa-wmm under the Machine Reading Comprehension Paradigm

Mengpei Li, Jun Pan *

School of Science, Zhejiang University of Science and Technology, Hangzhou 310023, China
* **Corresponding author**: Jun Pan (Email: panjun78@qq.com)

Abstract: In practical applications within the e-commerce domain, there is often a requirement to identify product entities and their corresponding brand entities, based on their descriptions. However, there has been a relatively limited focus on studies addressing the Named Entities Recognition in the e-commerce domain. we crawled data from e-commerce websites and transformed them into a Named Entity Recognition dataset, which is suitable for Machine Reading Comprehension. Since the questions Machine Reading Comprehension contain a priori semantic information about the types of the entities, we propose a model that uses the MRC modeling paradigm to solve the task of recognizing brand entities as well as commodity entities in the e-commerce domain. The model encodes the contexts and the corresponding questions using the RoBERTa-wwm model, and then further extracts the semantic information of the contexts using an attention network. We utilize SoftMax as the decoding layer to get the head index and tail index of the entity, and finally use the matching module to get the entity index. Through experiments on two e-commerce datasets, the results show that the new method can significantly improve the recognition effect of Chinese NER in e-commerce domain.

Keywords: Deep Learning; Machine Reading Comprehension; Named Entity Recognition.

1. Introduction

Named Entity Recognition aims to identify certain entities with specific meanings from the text and categorize the entities into different classes. Named Entity Recognition is useful for Question-Answer Systems [1], Machine Translation [2], Event Extraction [3] and Entity Linking [4] and other downstream tasks.

Named Entity Recognition is often regarded as a sequence labeling problem or a Machine Reading Comprehension (MRC) task. The process of MRC involves enabling the model to comprehend the context of a provided text and the questions, followed by extracting the relevant information from the text that can answer the given questions.

In the MRC modeling paradigm, we use the question to represent the type of entity, and the model understands the text and then gives the corresponding answer according to the question. For example, when we need to extract the product entities in the sentence, the task is transformed into answering the question "Find out which products are mentioned in the sentence". When we need to extract brands, the task translates into answering the question "Find out which brands are mentioned in the sentence".

Compared with the MRC-based approach, the sequence-labeling-based approach treats the entity categories as class indexes and loses the linguistic information of the categories. For example, the two entity categories of "product" and "brand" are only represented by "0" and "1" to represent their corresponding classes and do not have semantic information. The questions constructed in MRC contain a priori information about the entity categories, for example, the question "find the products in the text" will provide corresponding semantics of "product". The question " find the brands in the text " will provide the corresponding semantics of "brand". Therefore, the overall NER performance is improved. Since the constructed questions contain a priori information, the accuracy of the a priori information will have an impact on the answers. Therefore, we conducted experiments on the effect of several different question templates.

2. Related Work

The Reading Comprehension Modeling paradigm is widely used with answering systems in the field of NLP [5], Question Answering [6], Information Retrieval [7] and other domains. As an upstream task of NLP, NER tasks can also be modeled as machine reading comprehension tasks [8]. Li et al. [9] proposed a unified framework for NER based on the reading comprehension paradigm that can handle both flat and nested entities. Presently, NER models that employ the MRC paradigm are primarily utilized to address NER challenges in the fields of biology [10] and finance [11]. In contrast, there has been relatively less research focused on named entity recognition within the e-commerce domain using the MRC paradigm.

3. Model Architecture

In the MRC-based Chinese entity recognition task, when given a sequence of input segments $X = \{x_1, x_2, \cdots, x_n\}$, where x represents the words that constitute a sentence and n represents the length of the sentence. The purpose of the MRC-based entity recognition modeling paradigm is to first construct the corresponding problem $Q = \{q_1, q_2, \cdots, q_m\}$ according to the entity types, where each entity type corresponds to one problem. The model obtains the answer to the question from S based on Q as a text fragment $A = \{x^1_{head,tail}, x^2_{head,tail}, ..., x^j_{head,tail}\}$ in S, where j represents the number of entities obtained from the question Q, and head and tail represent the corresponding positions of the entities in the original input segment S.

The overall structure of our proposed model is shown in

Figure 1. First, the RoBERTa-wwm model is used to encode the context and the corresponding problem, and only the output U corresponding to input sequence X is retained, and then the U is inputted to the attention layer, and the corresponding context feature H will be obtained finally. Then the context feature H is used to predict the head index position and tail index position of the corresponding entity in X. The matching module is used to match the predicted head index position and tail index position to finally get the entity position corresponding to the question in X.

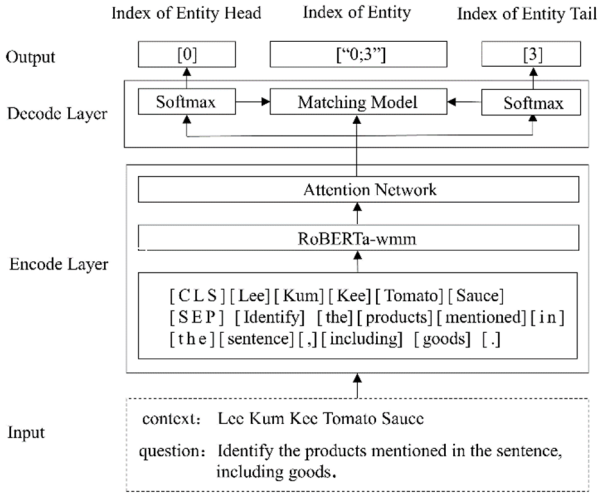

Figure 1. Model Structure

3.1. Input Representation Layer

For a given sequence of inputs $X=\{x_1,x_2,\cdots,x_n\}$ and a problem constructed based on the entity types $Q=\{q_1,q_2,\cdots,q_m\}$, we utilize both to construct the input representation $I=\{[CLS],x_1,x_2,\cdots,x_n,[SEP],q_1,q_2,...,q_m\}$ to the RoBERTa-wmm pre-trained model, where [CLS] and [SEP] denote special tokens.

3.2. Encoding Layer

After RoBERTa-wmm encoding, we get the corresponding output V that contains the understanding of the problem. we only keep the features $U\in\mathbb{R}^{n\times d}$ that are related to the context of the input sequence in V. To further utilize the contextual information, The U is sent to the attention network, where the formula for attention is shown below:

$$\text{Attention}(Q,K,V)=softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

Q, K, V Refer to the query matrix, key matrix, and value matrix obtained from U after transformation, respectively. d_k Represents the dimension of the input vector.

Eventually, after encoding the context features by using attention to them, we obtain the relevant encoded representation of the sentence $H=\{h_1,h_2,\cdots,h_n\}$, where $H\in\mathbb{R}^{n\times d}$.

3.3. Entity Head Prediction and Tail Prediction Layers

The model utilizes the contextual feature H to predict the probability of each token in the entire input sequence as the head or the tail of the entity, respectively.

The specific formula is shown below.

$$Z_1=HW_1 \tag{2}$$

$$Z_2=HW_2 \tag{3}$$

Where $W_1\in\mathbb{R}^{d\times 2}$ and $W_2\in\mathbb{R}^{d\times 2}$ are trainable parameters, $Z_1\in\mathbb{R}^{n\times 2}$ and $Z_2\in\mathbb{R}^{n\times 2}$ are entity head prediction features and entity tail prediction features, respectively. For each line of Z_1 and Z_2, Softmax function is used to get the probability of entity head prediction P_{head} and entity tail prediction P_{tail}.

$$P_{head}=softmax_{each\ \text{row}}(Z_1) \tag{4}$$

$$P_{tail}=softmax_{each\ \text{row}}(Z_2) \tag{5}$$

Eventually we get the probability that each token in the input sequence corresponds to the head and tail of the entity.

In order to obtain the head position prediction of the entity and the tail prediction result of the entity, we apply the following formulas on the corresponding prediction probabilities of each token separately:

$$\hat{K}_{head}=\{i\mid \arg max(p^i_{head}), i=1,\ldots,n\} \tag{6}$$

$$\hat{K}_{tail}=\{j\mid \arg max(p^j_{tail}), j=1,\ldots,n\} \tag{7}$$

Where p^i_{head} represents the i row of the P_{head} probability matrix and p^j_{tail} represents the j row of the P_{tail} probability matrix. $\hat{K}_{head}$ represents the index of the position in the sentence that is likely to be the head of the entity, and $\hat{K}_{head}$ represents the index of the position in the input sentence that is likely to be the tail of the entity.

3.4. Entity Head and Tail Matching Layer

In this stage, we combine and match all the head indexes i_{head} in the entity head collection $\hat{K}_{head}$ and all the tail indexes j_{tail} in the entity tail index collection $\hat{K}_{tail}$. To get the probability of predicted entity head and tail matching, we use binary classification as follows:

$$P_{i_{head},j_{tail}}=sigmoid\left(w\cdot concat\left(H_{i_{head}},H_{j_{tail}}\right)\right) \tag{8}$$

$w\in\mathbb{R}^{1\times 2d}$ are the trainable weights.

4. Training and Testing

4.1. Training

In the training phase, when given S training samples $[(x_{(i)};y'_{(i)}),\cdots,(x_{(S)};y'_{(s)})]$. We use the cross-entropy function as the loss function. The cross-entropy loss function is shown as follows.

$$L=\frac{1}{S}\sum_i-[y_i\cdot\log(p_i)+(1-y_i)\cdot\log(1-p_i)] \tag{9}$$

We utilize the cross-entropy loss function to calculate the difference between the predicted probability and the true label, the formula for this process is shown below:

$$L_{head}=L(P_{head},Y_{head}) \tag{10}$$

$$L_{tail} = L\left(P_{tail}, Y_{tail}\right) \tag{11}$$

$$L_{head:tail} = L\left(P_{head:tail}, Y_{head:tail}\right) \tag{12}$$

Where Y_{head} , Y_{tail} , $Y_{head:tail}$ represent the real labels about entity head, tail and entity matching span respectively. L_{head} , L_{tail} , $L_{head:tail}$ represent the loss function for entity head position prediction, the loss function for tail position prediction, and the loss function for entity span matching prediction, respectively.

The final objective loss function of the model is shown below:

$$L_{total} = \alpha L_{head:tail} + L_{head} + L_{tail} \tag{13}$$

where α is the hyperparameter.

4.2. Testing

In the testing phase, in order to get the final answer, we first get the head index and tail index of the entity in the entity head set $\hat{K}_{head}$ and the entity tail index set $\hat{K}_{tail}$. Subsequently, we use the entity head-tail pairing model to match them and get the final entity index.

5. Experiments

5.1. Data Sets

The two datasets used for the experiments are from the e-commerce domain and both contain entity types of product names, brand names.

JD-E-Commerce: Ding et al. [12] have published a dataset E-commerce in the field of e-commerce, which contains two types of entities: brands and products. However, considering that the E-commerce dataset is small, in order to be able to further demonstrate the validity of the model, we additionally collected relevant data from e-commerce websites to construct the JD-E-Commerce dataset.

We use web crawler to get the text related to the title content of the product from the shopping site. The crawling technique is based on the Python and utilizes the Selenium and Requests modules. Finally, we collected a total of 10,456 texts by crawling. We use the open-source Label Studio annotation tool to manually annotate entities, which supports fast matching of entity labels through a visual interface, and supports export in multiple formats. We adopt the BIOES annotation scheme to annotate entities for two types of entities.

We divide the JD-E-Commerce dataset into training set, validation set, and test set in the ratio of 6:2:2.

Table 1 shows the detailed statistical information of the JD-E-Commerce dataset.

Table 1. Statistical information of JD-E-commerce dataset

JD-E-commerce dataset	training set	validation set	test set
sentences	6275	2091	2090
characters	339.5k	113.3k	113.1k
Maximum number of characters	98	91	97
Minimum number of characters	8	13	15

E-commerce: for the E-commerce dataset, we used the same method as Ding et al [13] . Table 2 shows the statistical information of the E-commerce dataset.

Table 2. Statistical information on E-commerce dataset

E-commerce dataset	training set	validation set	test set
character	119.1k	14.9k	14.7k
sentences	3989	500	498

We need to transform the labeled dataset into a dataset suitable for MRC. Rajpurkar et al.[13] in 2016 proposed SQuAD, a Stanford question and answer dataset that can be used for MRC. The dataset in this paper was constructed in a form similar to the SQuAD dataset. We then wrote scripts to convert the format of the dataset from the BIESO labeled format to the MRC dataset format.

5.2. Evaluation Metrics

There are three main evaluation metrics currently used for Named Entity Recognition, which are Precision, Recall, and F1-score. Precision is the ratio between the correctly predicted positive examples and all the examples predicted as positive. Recall is the ratio between correctly predicted positive examples and all actual positive examples. $F1$ is calculation of the value metric utilizing both P and R , which are comprehensive evaluation metrics. $F1$ is widely used to evaluate the performance of the model. We use $F1$ value, P and R as the evaluation metrics. The calculation of the three metrics is shown below:

$$P = \frac{TP}{TP + FP} \times 100\% \tag{14}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{15}$$

$$F1 = \frac{2 \times P \times R}{P + R} \times 100\% \tag{16}$$

Where TP is the sample that was correctly predicted as positive. FP is the sample that was incorrectly predicted as positive. FN is the sample that was incorrectly predicted as negative.

5.3. Results and Analysis

We demonstrate the effectiveness of our proposed model by comparing the experimental results with existing models on both JD-E-Commerce dataset and E-commerce dataset.

For both datasets, we use the following model as a comparison model:

(1) LSTM+CRF: The classic sequence labeling based NER model.

(2) RoBERTa-wwm model based on sequence labeling: With the modeling paradigm of sequence labeling, the results are obtained by fine-tuning on the dataset using the RoBERTa-wwm model.

(3) RoBERTa-wwm model based on MRC: MRC is used as a modeling paradigm and RoBERTa-wwm model is utilized as an encoder.

JD-E-Commerce: Tables 3 show the performance of our model on JD-E-Commerce dataset. From the results, it can be seen that our proposed model gets the highest F1 value compared to other models. Comparing with the classical model LSTM+CRF, it improves 4.52% in F1 value metrics. Comparing with sequence labeling based RoBERTa-wwm model, it improves 1.44% in F1 value. The comparison

between sequence labeling based RoBERTa-wwm model and MRC based RoBERTa-wwm, we can prove that the modeling approach of reading comprehension can effectively improve the performance of the model.

Table 3. Experimental Comparison Results in the JD-E-Commerce

Model	P	R	F1
Bi-LSTM+CRF	82.91	81.40	82.14
RoBERTa-wwm based on sequence labeling	84.75	85.69	85.22
RoBERTa-wwm based on MRC	86.27	85.53	85.90
Our model	87.19	86.13	86.66

E-commerce: Table 4 shows the performance of our model on the E-commerce dataset. We use the model proposed by Ding et al [12], the producers of the E-commerce dataset, as a comparative model. Their model is based on a graph neural network that captures the information in the Gazetteer through a multi-graph structure. Comparing with the classical Bi-LSTM + CRF and the model proposed Ding et al, our model improves 5.80% and 4.13% in F1, respectively. Comparing with sequence labeling based RoBERTa-wwm model and MRC-based RoBERTa-wwm model, it improves 1.17% and 0.51% in F1 value, respectively.

Table 4. Experimental Results in E-Commerce

Model	P	R	F1
Bi-LSTM+CRF	76.14	71.17	73.57
Ding et al.[13]	76.21	74.31	75.24
RoBERTa-wwm based on sequence labeling	78.45	77.97	78.20
RoBERTa-wwm based on MRC	78.01	77.72	78.86
Our model	79.96	78.79	79.37

5.4. Different Question Templates Comparison Experiment

Since the question contains a priori information, the accuracy of the a priori information will have an impact on the answer. If the a priori information is wrong, it will make the whole model perform poorly. Meanwhile, the richness of the a priori information will also have an impact on the overall performance of the model. In order to explore the impact of different question templates on the model performance, we conducted corresponding experiments for several different question template about e-commerce related entities. We constructed corresponding Q&A datasets for each of the following three questioning templates.

(1) Template 1: Describe the entity category in as much detail as possible by referring to the annotation template used to annotate the entity. Adopt the annotation template that we use when manually annotating entities.

(2) Template 2: Construct questions directly with keywords for entity categories.

(3) Template 3: Constructing questions with explanations corresponding to entity categories. We refer to Wikipedia's explanation.

Specific details about the question template are shown in Table 5.

Table 5. Different Questioning Templates

Question template	Brand entity	Product entity
Template 1	Identify the brands in the sentence, including companies.	Identify the products mentioned in the sentence, including goods.
Template 2	Find the brands mentioned in the sentence	Find the product mentioned in the sentence
Template 3	Identify the symbolic signs in the sentence that indicate a product to the public	Find anything produced in the sentence

We conducted the corresponding experiments on the JD-E-Commerce as well as E-Commerce datasets by constructing the corresponding dataset forms with these three question construction templates, respectively. Table 6 and 7 shows the experimental results of the impact of different questioning templates on the model performance.

Table 6. Different Questioning Templates in JD-E-Commerce

Question templates	P	R	F1
Template 1	87.19	86.13	86.66
Template 2	86.64	86.21	86.42
Template 3	85.32	84.14	84.72

Table 7. Different Questioning Templates in E-Commerce

Question templates	P	R	F1
Template 1	79.96	78.79	79.37
Template 2	79.87	78.64	79.25
Template 3	78.49	76.88	77.67

From the results, the model can obtain better performance in the e-commerce domain when we construct the question templates with either manual annotations or keywords of entity categories. The manual annotation of the dataset carries more semantically similar words, which helps the model to accurately recognize the entity categories. When we construct the question as an interpretation of the entity category, it leads to a decrease in the performance of entity recognition, where the F1 value of the model decreases to the extent of 1.94% as well as 1.7% on the JD-E-Commerce dataset as well as on the E-Commerce dataset. This excessive interpretation prevents the model from using it as explicit a priori information. Therefore, when there is a bias in the questions, it can lead to an overall performance degradation. The best way is to construct the question based on the annotation template during manual annotation, giving some of the proximate synonyms of the entity types to facilitate semantic matching by the model.

5.5. Effect of Different Loss Function Weights on the Model

The target loss function consists of loss functions used to control the entity span matching, entity head position and tail position prediction tasks. The hyperparameters α determine the contribution of the loss function in the entity matching task to the overall training task. We conducted several corresponding experiments on the size of the hyperparameters with an interval of 0.1 units between (0,1]. The effect of the loss weight α on the final experimental results is shown in Figures 2.

According to the results shown in the figure, for the JD-E-Commerce dataset, the recognition rate of entities is the highest when the weight of the loss function is set to 0.8. For the E-Commerce dataset, the recognition rate of entities is highest when the loss function weights are set to 0.7. Improper configuration of the hyperparameter often results in the degradation of entity recognition performance. Therefore, when performing Chinese E-commerce NER, we need to choose the corresponding loss weights.

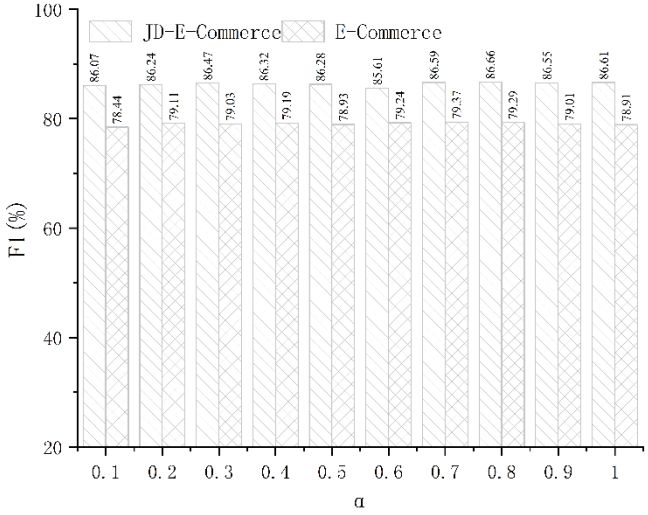

Figure 2. Effect of Loss Weights α on Experimental Results

6. Conclusion

In this paper, we propose a model using an MRC modeling paradigm to solve the problem of Named Entity Recognition in the E-commerce domain. Since the questions in MRC contain a priori semantic information about the entity type, the model performs better compared to models based on sequence labeling. In order to prove the effectiveness of the model, we collected product description data from shopping websites through crawlers and transformed it into a dataset suitable for MRC using scripts. We conducted relevant experiments on two e-commerce entity recognition datasets to demonstrate the effectiveness of the method.

Acknowledgments

This work is supported in part Zhejiang Public Welfare Technology Application Research Project of China (Grant: LGN21F020003).

References

[1] Diefenbach D, Lopez V, Singh K, et al. Core techniques of question answering systems over knowledge bases: a survey [J]. Knowledge and Information systems, 2018, 55(3): 529-569.

[2] Dandapat S, Way A. Improved named entity recognition using machine translation-based cross-lingual information[J]. Computaciónу Sistemas, 2016, 20(3): 495-504.

[3] Cheng Q, Liu J, Qu X, et al. HacRED: A large-scale relation extraction dataset toward hard cases in practical applications [C]. Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021: 2819-2831.

[4] Martins P H, Marinho Z, Martins A F T. Joint learning of named entity recognition and entity linking[C]. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. student Research Workshop, 2019: 190-196.

[5] Levy O, Seo M, Choi E, et al. Zero-Shot Relation Extraction via Reading Comprehension[C]. Proceedings of the 21st Conference on Computational Natural Language Learning. 2017: 333-342.

[6] McCann B, Keskar N S, Xiong C, et al. The natural language decathlon: multitask learning as question answering[J]. arXiv preprint arXiv:1806.08730, 2018. [Online]. Available: https:// arxiv. org/pdf/1806.08730.pdf

[7] Li X, Yin F, Sun Z, et al. Entity-Relation Extraction as Multi-Turn Question Answering[C]. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 1340-1350.

[8] Fei Y, Xu X. GFMRC: A Machine Reading Comprehension Model for Named Entity Recognition[J]. Pattern Recognition Letters, 2023.

[9] Li X, Feng J, Meng Y, et al. A Unified MRC Framework for Named Entity Recognition[C]. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 5849-5859.

[10] Chen P, Wang J, Lin H, et al. Knowledge Adaptive Multi-way Matching Network for Biomedical Named Entity Recognition via Machine Reading Comprehension[J]. IEEE Transactions on Computational Biology and Bioinformatics, 2023.

[11] Zhang Y, Zhang H. FinBERT-MRC: Financial Named Entity Recognition Using BERT Under the Machine Reading Comprehension Paradigm[J]. Neural Processing Letters, 2023: 1-21.

[12] Ding R, Xie P, Zhang X, et al. A neural multi-digraph model for Chinese NER with gazetteers[C]. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019:1462-1467.

[13] Rajpurkar P, Zhang J, Lopyrev K, et al. SQuAD: 100,000+ Questions for Machine Comprehension of Text[C]. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016: 2383-2392.