information

Article

An Attention-Based Model Using Character Composition of Entities in Chinese Relation Extraction

Xiaoyu Han [1,2,3], Yue Zhang [1,2], Wenkai Zhang [1,2] and Tinglei Huang [4,*]

[1] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China; hxy8149989@163.com (X.H.); zhangyue@aircas.ac.cn (Y.Z.); zhang.wenkai@outlook.com (W.Z.)

[2] Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China

[3] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China

[4] Institute of Software, Chinese Academy of Sciences, Beijing 100190, China

* Correspondence: tlhuang@mail.ie.ac.cn

Received: 16 December 2019; Accepted: 30 January 2020; Published: 31 January 2020

Abstract: Relation extraction is a vital task in natural language processing. It aims to identify the relationship between two specified entities in a sentence. Besides information contained in the sentence, additional information about the entities is verified to be helpful in relation extraction. Additional information such as entity type getting by NER (Named Entity Recognition) and description provided by knowledge base both have their limitations. Nevertheless, there exists another way to provide additional information which can overcome these limitations in Chinese relation extraction. As Chinese characters usually have explicit meanings and can carry more information than English letters. We suggest that characters that constitute the entities can provide additional information which is helpful for the relation extraction task, especially in large scale datasets. This assumption has never been verified before. The main obstacle is the lack of large-scale Chinese relation datasets. In this paper, first, we generate a large scale Chinese relation extraction dataset based on a Chinese encyclopedia. Second, we propose an attention-based model using the characters that compose the entities. The result on the generated dataset shows that these characters can provide useful information for the Chinese relation extraction task. By using this information, the attention mechanism we used can recognize the crucial part of the sentence that can express the relation. The proposed model outperforms other baseline models on our Chinese relation extraction dataset.

Keywords: relation extraction; Chinese; character; attention; distant supervision

1. Introduction

Relation extraction aims to identify the relationship between two specified entities in a sentence. For example, from the sentence "LeBron James was born in Akron, Ohio.", we can get triple informaiton (LeBron James, Birthplace, Akron). Since it was put forward, relation extraction has been one of the most critical tasks in NLP (Nature Language Processing) and played a crucial role in QA (Question-Answer), Knowledge Graph construction, and many other applications.

There have been many studies in relation extraction, both in English and other languages. These methods show a trend from initial rule-based methods, traditional feature-based models, such as SVM (Support Vector Machine) [1] and probabilistic graphical models [2], to neural network-based approaches [3,4]. At the same time, the research focus also changes from supervised learning to distant supervised learning [5].

Besides finding different ways of modeling the sentences, researchers also try to use additional information such as entity information in the task. Some studies use entity type [4] and entity descriptions [6]. However, both of these methods have their limitations. The number of entity types obtained by the NER (Named Entity Recognition) system is not enough, especially in large scale relation extraction. Even though there is a large knowledge base, only a small part of the entities in the dataset can find the appropriate descriptions when using the entity descriptions. However, there exists another way that can overcome these limitations to provide information about the entities in Chinese relation extraction tasks. A notable difference between English and Chinese is the characters. In Chinese, there exists another way to provide information about the entities. A notable difference between English and Chinese is the characters. In English, there are only 26 letters. Most of them do not have specific meanings. In Chinese, there are thousands of frequently-used characters, and plenty of them have explicit meanings. Based on this difference, we suggest that we can get information about the entities, such as type, color and location. from characters that constitute the entities. For example, as shown in Figure 1. The word ’中国’ has two Chinese characters, ’中’ and ’国’. From the character ’国’, which can express a country, we can infer that the word means a country in high probability. Moreover, when given the word ’李小龙’, we may know that the word refers to a person as the first character ’李’ usually appears in the first name. So, by using character compositions of the entities, we can provide more information about the entities compared with the entity types provided by the NER system, and it can provide information of all the entities without extra resources.

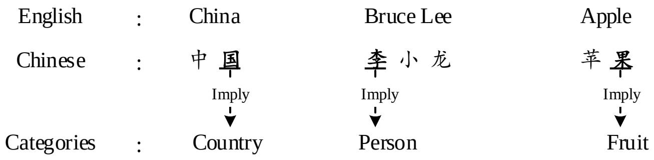


Figure 1. Information implied by Chinese characters.

The effect of this method is still not verified as far as we know. The main reason is lacking a large-scale open-domain dataset. To verify the hypothesis, this paper creates a large scale Chinese relation dataset based on a Chinese encyclopedia. Based on this dataset, we propose an attention-based model to verify the effectiveness of the character information provided by entity compositions in the Chinese relation extraction task. The experimental results show that by using this information, the attention mechanism can recognize the crucial part of the sentence through which we can infer the relationship between two entities. Furthermore, the proposed model also achieves better performance compared with other baseline models that are widely used in the relation extraction tasks. The main contribution of this work are as follows.

First, we build Baike dataset using distant supervision based on Baidubaike, a large scale online Chinese encyclopedia, to solve the problem of lacking large-scale open-domain datasets. We elaborate on the process of generating the dataset and analyze it from several aspects such as instance distribution and label accuracy of each relation during distant supervision. After comparing with other datasets, we believe our dataset is the most appropriate dataset for the large scale Chinese relation extraction task.

Second, we propose the BLSTM-CCAtt (bidirectional-LSTM model using Character Composition Attention) model, which is an attention-based neural network model using the information provided by Chinese character compositions of entities. Through this model, we illustrate how this information is useful to infer the relation between two entities in detail. Then we analyze how this information works in our model in detail. Moreover, the experiment results show that the proposed model gets the best F1 score among all the tested models on the Baike dataset.

2. Related Work

2.1. Neural Network in Relation Extraction

Neural Network has become the mainstream in NLP studies and it achieves the best performance in the relation extraction task. Socher et al. [3] propose the Recursive Matrix-Vector Model that uses Recursive Neural Network to model the shortest dependency path (SDP) between entities in the sentence. Zeng et al. [4] introduce Convolutional Neural Network (CNN) to the relation extraction task. These two studies are the earliest work using neural networks in relation classification. The result shows that these methods get better results than the traditional feature-based methods. Zeng et al. [7] propose Piecewise Convolutional Neural Network (PCNN). PCNN separates the sentence into three parts by the two given entities and uses max-pooling separately after the convolutional layer. Xu et al. [8] and Xu et al. [9] use CNN and Long Short Term Memory Network (LSTM) to model the SDP between the two given entities respectively. Liu et al. [10] consider the subtrees attached to the SDP. Before modeling SDP by CNN, the embedding of subtrees getting by recursive neural network is appended. Since the attention-based models improve the performance of many NLP tasks, attention is also used in relation classification. Zhou et al. [11] propose a LSTM model with attention. Wang et al. [12] choose multi-layer CNN with attention. Both work show better performance than the models without attention.

2.2. Distant Supervision

Lacking labeled data is a major problem in relation extraction. Especially in large scale knowledge graph construction that involves thousands of relations, the cost of labeling data manually is unacceptable. To solve this problem, Mintz et al. [5] propose distant supervision using triples from freebase to label unstructured text. Data generated by distant supervision is quite noisy. To alleviate the noise, Riedel et al. [13], Hoffmann et al. [14] and Surdeanu et al. [2] use graph model to find which instances are labeled incorrectly. In the area of neural relation extraction, Zeng et al. [7] use multi-instance learning at the first time. In their work, they use sentence bags as the input of their model instead of one simple sentence. When training the model, they select the sentence with the max calculated probability to update the parameters. Lin et al. [15] adopt attention to optimize instance selection. Qin et al. [16] use Generative Adversarial Network (GAN) to solve this problem of wrong labeled instances.

2.3. Chinese Relation Extraction

Studies in Chinese relation extraction are far less than English. One crucial reason is lacking large-scale datasets. Many previous work use the ACE 2005 Chinese corpus (LDC2006T0 6) dataset that is quite small for neural-net-based methods. So, some work choose to make their own dataset. For example, Chen et al. [17] make a dataset which contains three types of relations and test mult-instance learning on it. Wen et al. [18] use a dataset based on Chinese SanWen and propose a structure regularized neural network. Most of these previous work are based on word-level or character-level. So, some work decide to use multi-grained models to take advantage of both levels. The latest one of them is proposed by Li et al. [19] which uses a lattice-based structure to dynamically integrate word-level features into the character-based method.

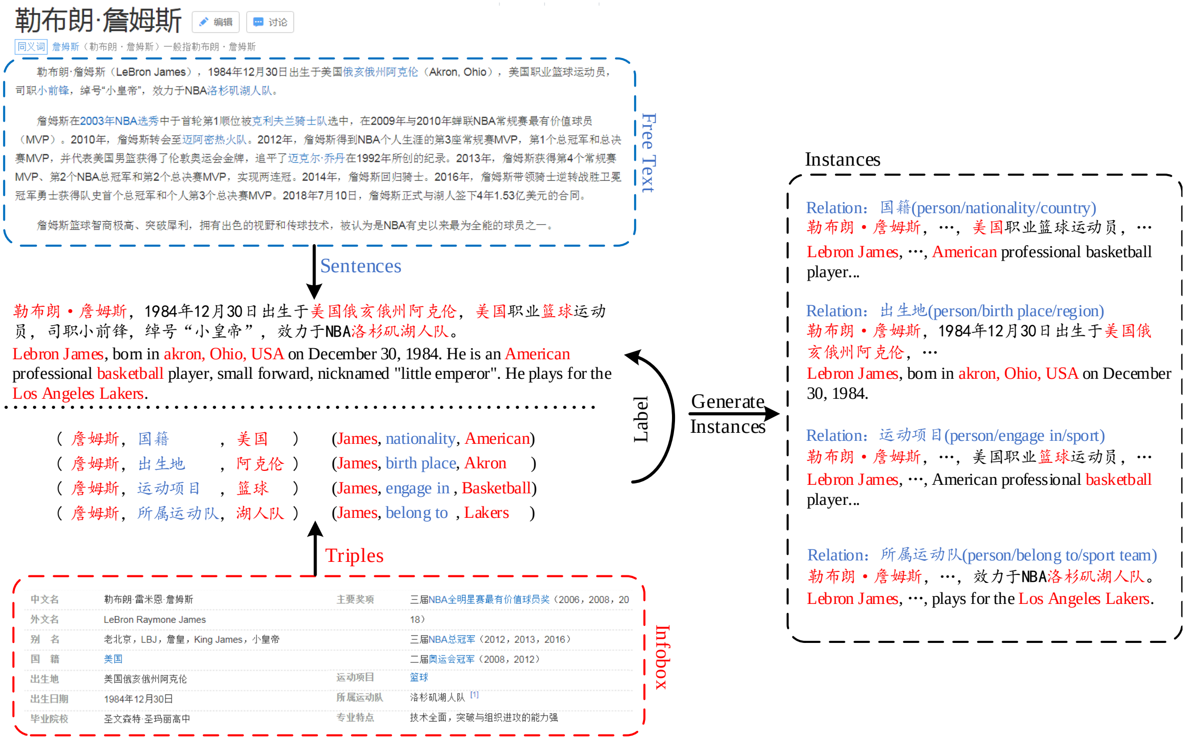
3. Dataset Construction

Dataset is a critical part of relation extraction. It determines whether the model trained by the dataset can apply to real-world problems. However, current Chinese relation extraction datasets are either too small or in a specific domain. So, our goal is to create a large scale open domain Chinese relation extraction dataset. As for now, there are two usual ways to create datasets. The first one is labeling all the data manually. In this way, we can get a high-quality dataset in which each instance is guaranteed to be right. However, this method is not appropriate to create large scale datasets for its cost. The second one is the distant supervision method proposed by Mintz [5] that uses known triples to label unstructured text. Although the quality of the dataset generated by distant supervision is not as good as that of the manually labeled dataset due to the introduction of wrong labeled instances in the process of auto-labelling, compared with the manually annotated data set, the labeling cost is negligible. Therefore, we choose distant supervision to generate our dataset. In this section, we illustrate the process of generating our Chinese relation extraction dataset, which is named the Baike dataset.

3.1. Dataset Collection

The most widely used dataset in English is NYT'10 dataset [13] that uses triples in Freebase to label raw text in NewYork Times. Here we use Baidubaike, an online Chinese encyclopedia, to generate the dataset. Compared with Freebase that provides triples to indicate relations between entities, Baidubaike is more like Wikipedia that contains text, tables, and pictures to describe a real-world thing that we treat as an entity. So unlike the NYT'10 dataset, we can both obtain triples and text.

The detailed process is shown as follows. First, we crawl one million pages from Baidubaike. These pages contain tables and text as Figure 2 shows. After filtration and disambiguation, each page can be treated as an introduction to an entity. In each page, the tables provide structured information about the entity such as birthplace or profession. After excluding items that describe attributes about the entity like height and weight, we can get relations about this entity that finally form triples. Then, we use these triples to label unstructured text in the page to get instances. All aliases are considered in the labeling process.



勒布朗·詹姆斯（LeBron James），1984年12月30日出生于美国俄亥俄州阿克伦（Akron, Ohio），美国职业篮球运动员，司职小前锋，绰号“小皇帝”，效力于NBA洛杉矶湖人队。

詹姆斯在2003年NBA选秀中于首轮第1顺位被克利夫兰骑士队选中，在2009年与2010年蝉联NBA常规赛最有价值球员（MVP）。2010年，詹姆斯转会至迈阿密热火队。2012年，詹姆斯得到NBA个人生涯的第3座常规赛MVP，第1个总冠军和总决赛MVP，并代表美国男篮获得了伦敦奥运会金牌，追平了迈克尔·乔丹在1992年所创的纪录。2013年，詹姆斯获得第4个常规赛MVP、第2个NBA总冠军和第2个总决赛MVP，实现两连冠。2014年，詹姆斯回归骑士。2016年，詹姆斯带领骑士逆转战胜卫冕冠军勇士获得队史首个总冠军和个人第3个总决赛MVP。2018年7月10日，詹姆斯正式与湖人签下4年1.53亿美元的合同。

詹姆斯篮球智商极高、突破犀利，拥有出色的视野和传球技术，被认为是NBA有史以来最为全能的球员之一。

勒布朗・詹姆斯，1984年12月30日出生于美国俄亥俄州阿克伦，美国职业篮球运动员，司职小前锋，绰号“小皇帝”，效力于NBA洛杉矶湖人队。

Lebron James, born in akron, Ohio, USA on December 30, 1984. He is an American professional basketball player, small forward, nicknamed "little emperor". He plays for the Los Angeles Lakers.

(詹姆斯, 国籍 , 美国) (James, nationality, American)
(詹姆斯, 出生地 , 阿克伦) (James, birth place, Akron)
(詹姆斯, 运动项目 , 篮球) (James, engage in , Basketball)
(詹姆斯, 所属运动队, 湖人队) (James, belong to , Lakers)

中文名	勒布朗·雷米恩·詹姆斯	主要奖项	三届NBA全明星赛最有价值球员奖（2006，2008，2018）
外文名	LeBron Raymone James		
别　名	老北京，LBJ，詹皇，King James，小皇帝		三届NBA总冠军（2012，2013，2016）
国　籍	美国		二届奥运会冠军（2008，2012）
出生地	美国俄亥俄州阿克伦	运动项目	篮球
出生日期	1984年12月30日	所属运动队	洛杉矶湖人队 [1]
毕业院校	圣文森特·圣玛丽高中	专业特点	技术全面，突破与组织进攻的能力强

Relation：国籍(person/nationality/country)
勒布朗・詹姆斯，…，美国职业篮球运动员，…
Lebron James, …, American professional basketball player...

Relation：出生地(person/birth place/region)
勒布朗・詹姆斯，1984年12月30日出生于美国俄亥俄州阿克伦，…
Lebron James, born in akron, Ohio, USA on December 30, 1984.

Relation：运动项目(person/engage in/sport)
勒布朗・詹姆斯，…，美国职业篮球运动员，…
Lebron James, …, American professional basketball player...

Relation：所属运动队(person/belong to/sport team)
勒布朗・詹姆斯，…，效力于NBA洛杉矶湖人队。
Lebron James, …, plays for the Los Angeles Lakers.

Figure 2. Extract labeled instances from Baidubaike.

After all the previous processes, we get 1,496,491 instances in 1444 relations. The number of instances of most relations is quite small that the model can hardly learn enough information to distinguish these relations. So, we select 53 of them that have more than 5000 instances as candidates. Unlike Freebase, relations described by tables in Baidubaike are irregular, and a large portion of them are ambiguous. We eventually select 30 relations that have relatively clear explanations among these candidates. The negative samples that mean the relation is not in these 30 relations are randomly chosen from the unselected relations. However, the data is quite imbalanced. The largest number of instances among these relations is more than 150,000, while the smallest one is only about 5000. To alleviate this problem, we randomly subsample the relations which contain too many instances.

After subsample, we divide the training and testing set by the proportion of 50:1. The minimum number of instances for each relation in the test set is limited to 200. In the dividing process, triples in the testing set are not allowed to appear in the training set so that the result can be less affected by the over-fitting of the trained model. All the instances in the testing set are labeled manually to eliminate the influence of mislabelling in the evaluating process.

3.2. Dataset Analysis

In this section, we analyze various aspects of the proposed dataset to provide a deeper understanding of the dataset and illustrate why it is more appropriate for the Chinese relation extraction task. The details of our dataset are shown in Table 1. The first column of the table is relation types in Chinese. The second column lists the interpretations of all the relation types. Each of them is described as 'head entity type/relation description/tail entity type' like Wikipedia. The head and tail entities correspond to the subject and object in the triple. The entity types confirm which kind of entity can appear in the relation. The relation description describes the relationship between these two entities. The third and fourth columns are numbers of triple and instance of each relation. The last column is the label accuracy of distant supervision estimated by the process of labeling the testing set. The accuracy of each relationship is calculated by dividing the number of instances correctly labeled by the total number of instances. The distribution of instances and triples is shown in Figure 3.

Table 1. Information of proposed dataset.

idx	Relation	Interpretation	#trip.	#inst.	Accuracy
0	NA	not in the relations above	42,394	48,930	NA
1	国籍	person/nationality/country	16,339	28,335	53.91%
2	职业	person/engage in/job	20,759	28,148	85.86%
3	出生地	person/place of birth/region	18,684	24,193	85.67%
4	主演	film work/actor or actress/person	39,104	49,544	93.49%
5	类型	film work(literary work)/type/film type(literary type)	17,465	27,536	56.76%
6	作者	literary work/writer/person	17,763	23,744	92.47%
7	所属地区	region(organization)/belong to/region	12,179	27,830	82.63%
8	代表作品	person/representative work/work	17,180	26,374	88.01%
9	经营范围	organization/scope of business/business	7738	19,774	68.49%
10	导演	film work/director/person	15,475	20,273	89.86%
11	毕业院校	person/graduate institution/organization	13,634	18,818	87.01%
12	运动项目	person/participate in/sport	7584	17,577	91.74%
13	总部地点	organization/location of headquarters/place	5483	13,970	75.58%
14	民族	person/race belongs to/race	9843	13,259	95.76%
15	出版社	literary work/publisher/publishing company	12,863	13,519	98.92%
16	下辖地区	region/contain/region	7262	12,258	33.33%
17	著名景点	region/contain/landscape	7435	12,452	61.56%
18	制片地区	film work/producer area/region	7920	12,241	53.52%
19	性别	person/belong to/sex	7637	8494	97.83%
20	编剧	film work/screenwriter/person	5532	8070	53.14%
21	科	animal and plant life/belong to/family	6801	8025	95.25%
22	歌曲原唱	song/singer/person	4368	7727	79.12%
23	所属国家	region(landscape)/belong to/region	2372	6284	85.02%
24	分布区域	animal and plant life/distribution/region	4328	6489	70.41%
25	主要食材	food/main ingredients/ingredient	3209	5144	83.25%
26	登场作品	character/come on stage/film work(literary work)	4037	5451	93.41%
27	常见症状	disease/common symptom/symptom	3068	4130	63.07%
28	所处时代	person/belong to/era	3467	4858	90.08%
29	所属运动队	person/belong to/sport team	3080	4814	86.46%
30	隶属	organization/belong to/organization	2818	4457	51.71%

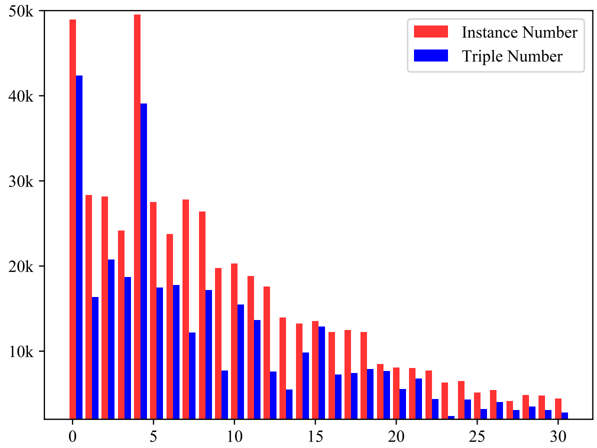


Figure 3. Instance/Triple Distribution in Baike dataset. The x-axis is the index of relations. The y-axis is the number of instances or triples.

First, we analyze the relation types in our dataset. Table 1 shows that the relation types in our dataset cover a broad scope. As shown in the last column of Table 1, theThe accuracy of distant supervision is quite different among relations. The average accuracy of all the labeled relations is 68.28%. The "出版社" relation has the highest accuracy, which is 94.83%. The accuracy of "下辖地区" relation is only 26.72%. This difference might be reflected in the relation extraction result. Methods of reducing noises in the dataset are helpful in the extraction process.

Then, we compared our dataset with two frequently used datasets. As shown in Table 2, our dataset contains more relation types and instances. The Chinese SanWen dataset [20] contains 9 types of relations among 726 Chinese literature articles, 29,096 sentences. The ACE 2005 dataset contains 8023 relation facts with 18 relation subtypes collected from newswires, broadcasts, and weblogs. Our dataset includes 463,788 instances in 30 relation types from different fields. Compared with other datasets, our dataset is much larger and covers wider fields. In the real world open-domain relation extraction task, there exist many kinds of sentences and thousands of relations. These small-scale or specific-domain datasets are incompetent to the task. So our dataset is more appropriate.In conclusion, our dataset is more suitable for large-scale open-domain relation extraction task.

Table 2. Comparison of datasets.

Dataset	**#cls.**	**#inst./cls**	**#inst.**	**Open Domain**
ACE2005	18	446	8023	True
SanWen	9	3233	29,096	False
Baike (Proposed)	30	13,393	401,787	True

4. Proposed Model

In this section we proposed a neural network model named as BLSTM-CCAtt that uses the character composition of the entities to provide additional information. The overall structure of our model is shown in Figure 4. The construction of this model is similar to most previous models that start with encoders and end up with a softmax classifier. However, unlike other work, we use the character composition to provide additional information about the entities. There are three encoders in our model, which is one sentence encoder and two entity encoders. We compare several frequently used encoders to select the most appropriate one. After comparison and analysis, bidirectional-LSTM (BLSTM) is chosen as all these three encoders. When encoding the sentence, attention mechanism uses the outputs of entity encoders as the query to give weight to the words or characters and get the vector expression of this sentence. After a full connection layer, a softmax classifier is used to classify the relationship between the entities.

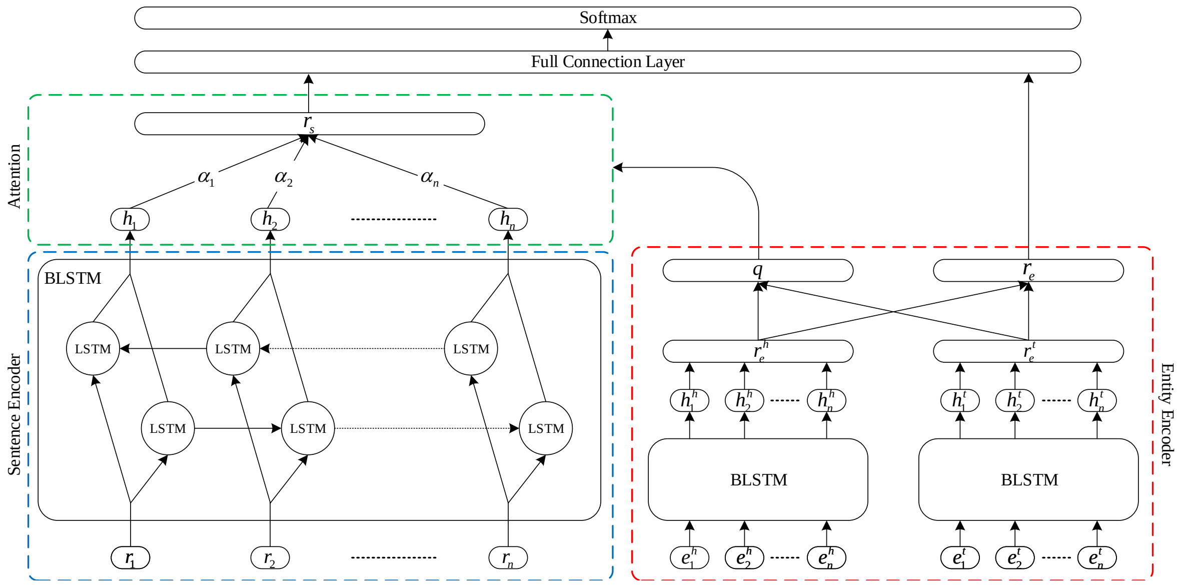


Figure 4. Proposed Attention-based model, $\boldsymbol{r_i}$ represents the i-th input of the sentence, $\boldsymbol{e_i^h}$ and $\boldsymbol{e_i^t}$ is the embedding of the i-th input of the head and tail entity. $\boldsymbol{q}$ and $\boldsymbol{r_e}$ are calculated by the head entity $\boldsymbol{r_e^h}$ and tail entity $\boldsymbol{r_e^t}$ in different ways. The weight α_i is calculated by $\boldsymbol{q}$ and $\boldsymbol{h}$ the hidden states using attention.

4.1. Embedding

Following most neural network models, the first step of our model is to transform the input tokens into low-dimensional vectors. When encoding sentences, the "input tokens" refers to words or characters according to whether the encoder is word-level or character-level. These input tokens are transformed into vectors by looking up the pre-trained embeddings. Position feature [4] is used to specify the given entity pair. It also needs to be transformed into vectors by looking up the position embeddings. When encoding entities, the "input tokens" refers to the Chinese characters that composed the two entities. These characters are also transformed into vectors.

Given a sentence with n input tokens $S = \{s_1, s_2, \ldots, s_n\}$, two marked entities e_h and e_t, and an embedding matrix E_s of dimension $d_c \times |V|$, where d_c is a hyper-parameter that indicates the dimension of the embedding vector, and V stands for the vocabulary, every input token s_i is represented as vector $\boldsymbol{v_i} \in \mathbb{R}^{d_c}$ after projected into the embedding space. Position feature is widely used in previous work, and the effect is verified. For each token s_i in the sentence, we can get two relative distances to the two entities. The distances are mapped to randomly initialized vectors $\boldsymbol{p_i^h}$ and $\boldsymbol{p_i^t}$, $\boldsymbol{p_i} \in \mathbb{R}^{d_p}$ where d_p is a hyper-parameter which indicates the dimension of position vector. The final representation of token s_i in the sentence is the connection of the token embedding and two position embeddings, which is $\boldsymbol{r_i} = [\boldsymbol{v_i} : \boldsymbol{p_i^h} : \boldsymbol{p_i^t}]$, $\boldsymbol{r_i} \in \mathbb{R}^{d_c + 2 \times d_p}$. The sentence is finally represented as $R = \{\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r_n}\}$.

The representation of entities is similar to the sentence. Each character c_i that constitutes an entity $E = \{c_1, c_2, \ldots, c_m\}$ is mapped to $\boldsymbol{e_i} \in \mathbb{R}^{d_c}$ using embedding matrix E_e. The entity is finally represented as $E = \{\boldsymbol{e_1}, \boldsymbol{e_2}, \ldots, \boldsymbol{e_m}\}$.

4.2. Encoders

As mentioned above, plenty of models have been used to encode a given sentence including CNN, RNN, and more complex neural networks. To select the appropriate encoder, we consider both word-level and character-level models. Le et al. [21] illustrate that shallow-and-wide networks have better performance than deep models with word inputs. On the other side, deep models indeed give better performances than shallow networks when the text input is represented as a sequence of characters. However, the property of Chinese decides that character-level models can be simpler than English. So, after comparing several models, we eventually select BLSTM as our sentence encoder. There are three reasons for using BLSTM in our model. First, BLSTM shows similar or even better performance when given character composition information of entities. Second, LSTM-based models have more explicit meanings in the attention mechanism that is used in the next step than CNN-based models. Last, BLSTM is quite simple compared with other complex models, which means it has fewer parameters and faster calculating speed. The detailed encoding process is shown in Figure 4. Given a sentence $R = \{\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r_n}\}$, the hidden states of forward LSTM $\boldsymbol{H_f}$ are

$$\boldsymbol{H_f} = \{\boldsymbol{h_1^f}, \boldsymbol{h_2^f}, \ldots, \boldsymbol{h_n^f}\} = LSTM\{\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r_n}\} \tag{1}$$

and backward LSTM $\boldsymbol{H_b}$ is

$$\boldsymbol{H_b} = \{\boldsymbol{h_1^b}, \boldsymbol{h_2^b}, \ldots, \boldsymbol{h_n^b}\} = LSTM\{\boldsymbol{r_n}, \boldsymbol{r_{n-1}}, \ldots, \boldsymbol{r_1}\} \tag{2}$$

The final hidden states of BLSTM sentence encoder are

$$\boldsymbol{H_s} = \{\boldsymbol{h_1}, \boldsymbol{h_2}, \ldots, \boldsymbol{h_n}\} \tag{3}$$

where $\boldsymbol{h_i} = [\boldsymbol{h_i^f} : \boldsymbol{h_i^b}]$.

Given an entity $E = \{\boldsymbol{e_1}, \boldsymbol{e_2}, \ldots, \boldsymbol{e_n}\}$, we use a BLSTM encoder to encode the entity just like encoding the sentences. The hidden states of entity encoder are

$$\boldsymbol{H_e} = BLSTM\{\boldsymbol{e_1}, \boldsymbol{e_2}, \ldots, \boldsymbol{e_n}\} \tag{4}$$

where $\boldsymbol{e_i}$ is the embedding of the i-th character of the entity and the calculation of BLSTM is the same with the sentence encoder. After the BLSTM encoder, the average pooling result of the hidden states $\boldsymbol{r_{e_i}} \in \mathbb{R}^{d_e}$, where d_e is the size of hidden states of the entity encoder, is used as the representation of the entity.

4.3. Attention

After encoding the sentence and two entities, we use the attention mechanism to take the best advantage of the information provided by the character composition of the entities.

The attention mechanism is widely used in NLP tasks such as QA and Machine Translate. It aims to select the most relevant part concerning the given query. The workflow of attention mechanism is as follows. Given a series of states $\boldsymbol{V} = \{\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_n}\}$ where $\boldsymbol{v_i} \in \mathbb{R}^{d_v}$, keys $\boldsymbol{K} = \{\boldsymbol{k_1}, \boldsymbol{k_2}, \ldots, \boldsymbol{k_n}\}$ where $\boldsymbol{k_i} \in \mathbb{R}^{d_k}$, and one query $\boldsymbol{q} \in \mathbb{R}^{d_q}$. The output $\boldsymbol{x}$ is calculated as attention vector $\boldsymbol{\alpha}$ multiply by the states $\boldsymbol{V}$.

$$\boldsymbol{x} = \boldsymbol{\alpha V} \tag{5}$$

$\boldsymbol{\alpha}$ is calculated by $\boldsymbol{q}$ and $\boldsymbol{K}$ using attention function f_{att}.

$$\boldsymbol{\alpha} = softmax\Big(f_{att}(\boldsymbol{q}, \boldsymbol{K})\Big) \tag{6}$$

In most NLP tasks, the states V are also used as the keys K. In our model, we use the hidden states of our sentence encoder $\boldsymbol{H_s}$ as $\boldsymbol{V}$. So the attention vector can be calculated as,

$$\boldsymbol{\alpha} = softmax\Big(f_{att}(\boldsymbol{q}, \boldsymbol{H_s})\Big) \tag{7}$$

There are several forms of attention function f_{att}, the multiply form is frequently used and selected in our model. The function is as follows:

$$f_{att} = \boldsymbol{qWH_s} \tag{8}$$

where $W \in \mathbb{R}^{d_q \times d_V}$ is a parameter matrix. There is no query $\boldsymbol{q}$ in the relation extraction task. To solve this problem, we use the representations of two entities $\boldsymbol{r_e^h}$ and $\boldsymbol{r_e^t}$, where h and t indicate whether the entity is the head or tail entity, to generate the query $\boldsymbol{q}$. Previous work [22] has demonstrated the property of word embeddings, for example $w(''China'') - w(''Beijing'') = w(''Japan'') - w(''Tokyo'')$. This means the difference between two word embeddings can indicate the relationship of these two words more or less. This is more clear in KG embedding. The basic assumption of many knowledge graph embedding work [23,24] is that given a triple (h, l, t), where h and t are two entities in relation l, the embedding should satisfy equation $\boldsymbol{h} + \boldsymbol{l} = \boldsymbol{t}$. Base on this assumption, $\boldsymbol{q}$ is calculated as follows.

$$\boldsymbol{q} = \boldsymbol{r_e^h} - \boldsymbol{r_e^t} \tag{9}$$

The final representation of the sentence $\boldsymbol{r_s}$ is calculated as follows.

$$\boldsymbol{r_s} = softmax(\boldsymbol{qWH_s})\boldsymbol{H_s} \tag{10}$$

In order to emphasize the entity information, we connect the sentence and entity representation as the instance representation $\boldsymbol{r}$.

$$\boldsymbol{r} = [\boldsymbol{r_s} : \boldsymbol{r_e}] \tag{11}$$

where $\boldsymbol{r_e}$ is the joint of the two entity representations.

$$\boldsymbol{r_e} = [\boldsymbol{r_e^h} : \boldsymbol{r_e^t}] \tag{12}$$

4.4. Multi-Instance Learning

Distant learning [5] has dramatically reduced the cost of getting labeled data and made it possible to generate large scale data sets. However, it is not perfect. The primary shortage is the wrong label problem. In order to solve this problem, multi-instance learning is introduced to the relation extraction task. Instead of one single sentence, the input of the network of multi-instance learning is a bag. Suppose there are m bags $\{B_1, B_2, \ldots, B_m\}$ and the k-th bag contains n instances $B_k = \{S_1, S_2, \ldots, S_n\}$ of the same entity pairs. Rather than labeling of each instance, multi-instance learning predicts the label of bags. So, the method of calculating the representation of bags is the key component of multi-instance learning. Several strategies such as selecting instance with the highest probability [7], attention-based method [6,15,25], adversarial training [26] and reinforcement learning [27,28] are used in previous work.

In this work, we use the sentence-level attention [15] that is simple and effective, as our multi-instance learning method. In this method, the representation of each bag is the weighted summation of the instances representations in the bag.

$$\boldsymbol{x} = \boldsymbol{\alpha_s R} \tag{13}$$

where $\boldsymbol{R} = \{\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r_n}\}$ is the matrix of the instance representations, and α_s is calculated as follows:

$$\alpha_s = softmax(\boldsymbol{RW_s l}) \tag{14}$$

where $\boldsymbol{W_s}$ is weighted diagonal matrix and $\boldsymbol{l}$ is the relation representation vector.

The prediction probability $\boldsymbol{p}$ of the bag is calculated as follows.

$$\boldsymbol{p} = softmax(\boldsymbol{Lx} + \boldsymbol{d}) \tag{15}$$

In this equation, $\boldsymbol{L}$ is the matrix of the relation representations, and $\boldsymbol{d}$ is the bias vector. Cross-entropy is used as the objective function. Adam algorithm [29] is adopted to minimize the objective function.

5. Experiments

In this section, we design a set of experiments to prove the advantage of our model and explain how our model works. First, we compare our model with several baseline models on our dataset. Second, we compare some popular encoders and try several ways to use the character composition information. After comparison, we find the method used in our model achieves the best performance. Then, we analyze how the attention mechanism used in our model works. Finally, we analyze the improvement of multi-instance learning on our dataset.

5.1. Experiment Result and Comparison

In this section, we compare several baseline models, which are widely used in relation extraction task, with the proposed model. These model are as follows:

CNN [4], the first CNN model used in relation classification. In this paper, we do not use the lexical features to avoid the influence of extra information getting by other tools.

PCNN [7], a piecewise CNN model that improves the CNN model by modifying the max-pooling method and use multi-instance learning.

BLSTM [30], a bidirectional RNN model for relation extraction. Herewe use LSTM instead of standard RNN cell.

Att-BLSTM [11], an attention-based bidirectional LSTM model.

BLSTM-SelfAtt [31], a self-attention based bidirectional LSTM model for sentence embedding. Here we add the position feature to figure out the two entities.

All the models are tested on the proposed Baike dataset. The multi-instance learning methods are removed from all the tested models to ignore the side effect. We conduct the experiments on both character-based and word-based versions of the models mentioned above. The AUC value and F1 score of these models are shown in Table 3. When calculating the F1 score, the negative samples are excluded. Each number in Table 3 is the average of 10 times experiments.

Table 3. AUC and F1-scores of different models.

Model	**Word Level**		**Character Level**	
	AUC	**F1**	**AUC**	**F1**
CNN	93.04	85.47	92.67	84.78
PCNN	93.72	85.88	92.82	84.79
BLSTM	93.82	86.43	92.86	85.12
Att-BLSTM	94.12	86.94	93.45	85.97
BLSTM-SelfAtt	94.11	86.99	93.64	86.05
BLSTM-CCAtt(Proposed)	94.76	87.30	94.26	86.13

The result shows that the proposed BLSTM-CCAtt model achieves the best performance among all the models in both in word-level and character-level. The performance of the LSTM-based models is better than the CNN-based ones. BLSTM-CCAtt (proposed), Att-BLSTM, and BLSTM-SelfAtt models all use attention methods. These attention-based models outperform the basic BLSTM model. The difference among these three models is the BLSTM-CCAtt model use character compositions of the two entities to generate the query $\boldsymbol{q}$ while the other two models use random initialized vectors as query $\boldsymbol{q}$. This difference demonstrates the advantage of using character compositions of entities. Compared with these baseline models, the F1 score of BLSTM-CCAtt is higher than the CNN-based models by about 1.5 and higher than other attention-based models by about 0.4. The improvement of the BLSTM-CCAtt model is significant on our Baike dataset.

5.2. Usage of the Character Composition Information

According to our hypothesis, character composition information can bring extra information, which can be beneficial for our relation extraction task. Many factors can affect the results of using this information, such as encoder selection and ways of using this information. Since many encoders have been used to encode the sentences, there exist several ways to use the character composition information. In this section, we test five popular encoders and three ways of using character composition information to find how we can take the most advantage of the character composition information. The result shows that the method we used in our model achieves the best result.

The tested five sentence encoders are CNN, PCNN, BLSTM, BLSTM-RES [32] and BLSTM-SelfAtt. Some of these encoders are used in previous work. In our method, these encoders are just part of our model. The results of these encoders are used together with the character composition information, which is obtained by the entity encoders, to obtain the final classification results.

There are three ways of using the character composition information. The first one abandons the attention mechanism and directly connects the sentence representation from sentence encoder with the entity representation from the entity encoders as the instance representation to predict the relationship between the entities. The second way uses the attention mechanism, which uses entity representation to calculate sentence representation. The calculated sentence representation is treated as the instance representation. The third way is the proposed method, which is called as Att&Con (Attention and Connection). In this method, the instance representation is the concatenation of the sentence representation calculated by the attention mechanism and the entity representation.

We try these methods on each encoder to find out in which situation we can take the most advantage of the character composition information. However, not all encoders can use these three methods. For example, self-attention-based models usually do not need external queries. So, the BLSTM-SelfAtt encoder only uses the first method. We do not use attention-based methods on CNN and PCNN because we believe that LSTM-based attention models are more interpretable in NLP tasks, although some work use CNN-based attention model [12]. All the tested combinations are list in Table 4. To emphasize the effect of character composition, we also try to use sentence representation with no character composition information. In this situation, the model is the same as previous work. All the methods are tested in both word-level and character-level. The result is shown in Table 4.

Table 4. Comparison of F1 score in different situation.

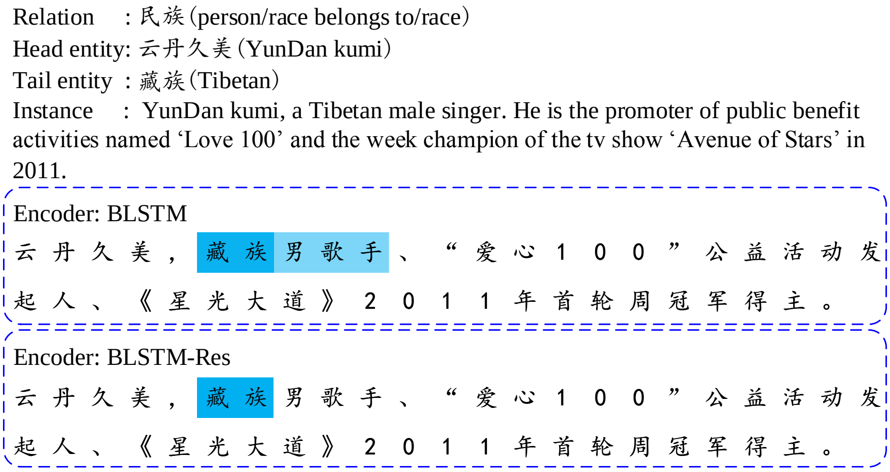
Encoder	Word Level				Character Level			
	None	Connect	Attention	Att&Con	None	Connect	Attention	Att&Con
CNN	85.47	86.10	NA	NA	84.78	85.11	NA	NA
PCNN	85.88	86.19	NA	NA	84.79	85.03	NA	NA
BLSTM	86.43	86.72	86.69	87.30	85.12	85.51	85.67	86.13
BLSTM-Res	86.84	86.96	86.01	86.12	85.61	85.83	84.39	84.90
BLSTM-SelfAtt	86.99	86.65	NA	NA	86.05	86.03	NA	NA

From Table 4, we can see that the performances of the RNN-based encoders are better than the CNN-based ones in all situations. When using the connection method, the F1 score of each model is improved except the BLSTM-SelfAtt model. The improvement in character-level is smaller than word-level. The reason is that in character-level, the information provided by character composition is included in the sentence. In word-level, this information is a useful supplement. In addition, there is no improvement in the BLSTM-SelfAtt model. The reason is that the self-attention mechanism gives higher weights to important elements so that it can capture enough information from the two entities. When using the attention method, BLSTM gets better results in both character-level and word-level, while the performance of BLSTM-RES using attention gets worse. We believe that compared with the connection method, the introduction of character composition information in the attention method is indirect. It tries to use the information to find the crucial part in the sentence that can decide the relation between the two entities. This mechanism fails in BLSTM-RES. Because, in BLSTM-RES, the attention mechanism tends to ignore most words in a sentence. The detailed analysis is listed in Section 5.3. When using the Att&Con method, the result shows that the proposed BLSTM-CCAtt model, which uses BLSTM encoder and Att&Con, can get the best result since it can take the most advantage of the character composition information.

5.3. Attention Analyze

The attention mechanism is a crucial part of the proposed BLSTM-CCAtt model. In this section, we illustrate how the attention mechanism works and demonstrate whether the attention mechanism can find the crucial parts of the sentence using character composition information. The crucial parts of the sentence are the words through which we can infer the relationship between the two entities. We explain the attention mechanism in three circumstances.

First, we focus on relations which we can deduce the relation through entities, such as '性别' and '民族'. In these relations, the set of entity in one side is quite small and hardly appear in other relations. For example, when given an entity '藏族', which means the Zang nationality, and the other one is a person name in a sentence, this sentence may belong to the '民族' relation in very high probability even consider the negative examples. As shown in Figure 5, the weight of the key entity calculated by entity representation is very high and other tokens in this sentence are nearly ignored, especially in BLSTM-Res model. This is the simplest situation, and both models make similar choices.



Relation : 民族 (person/race belongs to/race)
Head entity: 云丹久美 (YunDan kumi)
Tail entity : 藏族 (Tibetan)
Instance : YunDan kumi, a Tibetan male singer. He is the promoter of public benefit activities named 'Love 100' and the week champion of the tv show 'Avenue of Stars' in 2011.

Encoder: BLSTM
云 丹 久 美 ， 藏 族 男 歌 手 、 " 爱 心 1 0 0 " 公 益 活 动 发
起 人 、 《 星 光 大 道 》 2 0 1 1 年 首 轮 周 冠 军 得 主 。

Encoder: BLSTM-Res
云 丹 久 美 ， 藏 族 男 歌 手 、 " 爱 心 1 0 0 " 公 益 活 动 发
起 人 、 《 星 光 大 道 》 2 0 1 1 年 首 轮 周 冠 军 得 主 。

Figure 5. Attention analysis of '民族' relation, the darker the color is the higher the weight is.

Then, we consider the relations in which the set of its entity on one side is small but appear in more than one relation. The typical kind of these relations is which contain entities that represent countries such as '国籍' and '所属国家'. A typical example is given in Figure 6. In this example, the word '中国' which means China is the key clue to infer the relation. But only through this word, the relation can not be clearly judged because it can appear in both relations. The behaviors of these two models are different here. The BLSTM model gives higher weight to other words when emphasizing on the word '中国'. Through these words, the classifier can get information to make the right decision. On the other side, BLSTM-Res model only focuses on the key entity and ignore the other words. So, it can hardly give the right answer.

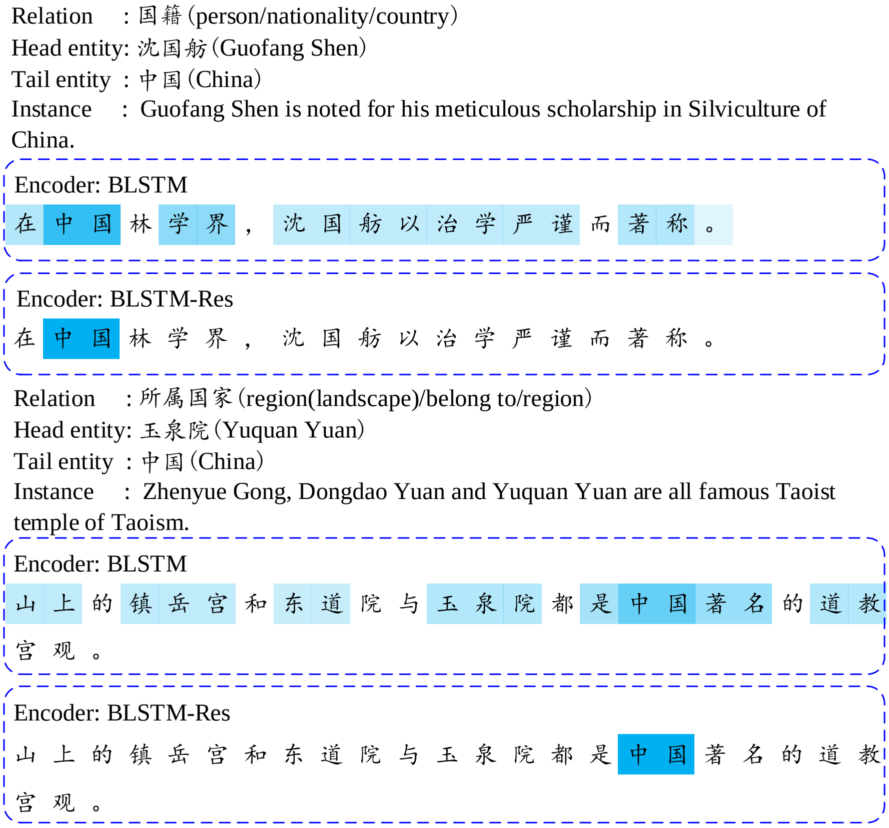

Relation : 国籍 (person/nationality/country)
Head entity: 沈国舫 (Guofang Shen)
Tail entity : 中国 (China)
Instance : Guofang Shen is noted for his meticulous scholarship in Silviculture of China.
Encoder: BLSTM
在 中 国 林 学 界 ， 沈 国 舫 以 治 学 严 谨 而 著 称 。
Encoder: BLSTM-Res
在 中 国 林 学 界 ， 沈 国 舫 以 治 学 严 谨 而 著 称 。
Relation : 所属国家 (region(landscape)/belong to/region)
Head entity: 玉泉院 (Yuquan Yuan)
Tail entity : 中国 (China)
Instance : Zhenyue Gong, Dongdao Yuan and Yuquan Yuan are all famous Taoist temple of Taoism.
Encoder: BLSTM
山 上 的 镇 岳 宫 和 东 道 院 与 玉 泉 院 都 是 中 国 著 名 的 道 教 宫 观 。
Encoder: BLSTM-Res
山 上 的 镇 岳 宫 和 东 道 院 与 玉 泉 院 都 是 中 国 著 名 的 道 教 宫 观 。

Figure 6. Attention analysis of '国籍' and '所属国家' relation.

In both two kinds of relations mentioned above, entities play a crucial role in determining the relation. In both situations, our model focuses on key entities that can decide the relationship. So, here comes the question of whether our model only focuses on the given entities. We analyze some other relations in which the entities are less important than some other keywords in the sentences and can provide few clue. It is uncertain whether the proposed attention mechanism can still find the critical part. We select '作者' and '歌曲原唱' relations which meet the requirements to analyze which part the attention focus on. The result is shown in Figure 7. In the '作者' relation, the word '作者', which can be interpreted as writer, has higher weight than other items. In the '歌曲原唱' relation, both models focus on the word '演唱', which means 'singing', and the word '一首', which is a quantifier usually used on songs. So, in these relations, the attention mechanism can still find out the critical part.

From all the situations mention above, we conclude that character information provided by entity composition can provide helpful clues to judge the relation. Besides that, we also find an interesting fact which may be closely related to the failure of attention mechanism in the BLSTM-Res model. When making the decision, the BLSTM-Res model tends to allocate high weights for a few words and ignore other words compared with the BLSTM model, leading to a loss of necessary information in the sentence. So, we also conclude that BLSTM is more suitable than BLSTM-Res to be the sentence encoder in the proposed model.

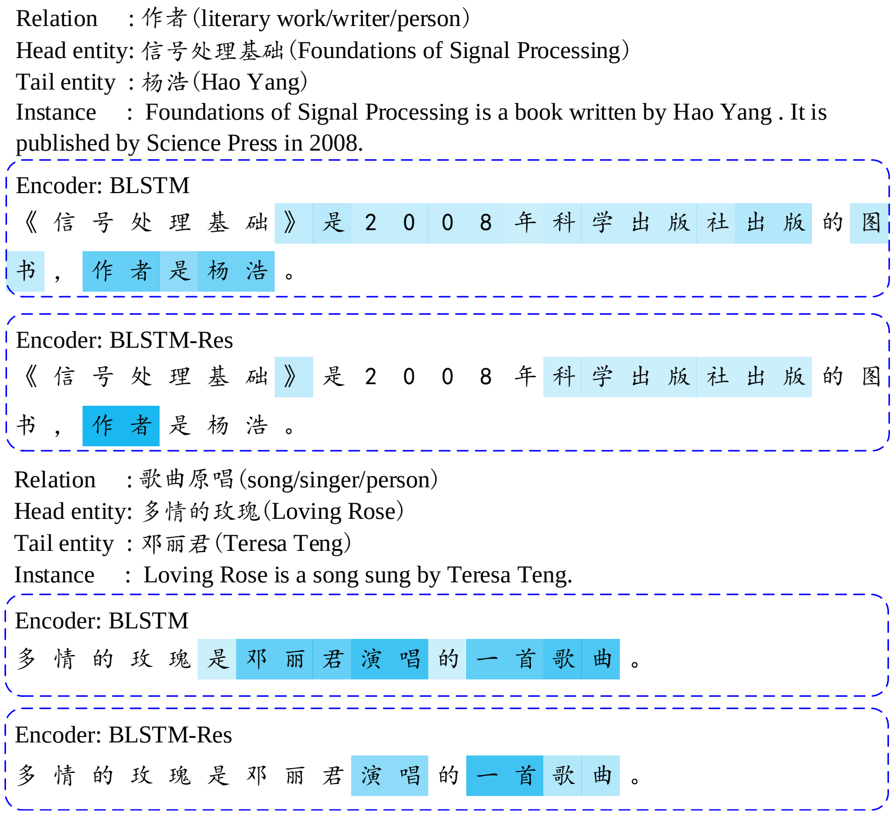

Relation : 作者 (literary work/writer/person)
Head entity: 信号处理基础 (Foundations of Signal Processing)
Tail entity : 杨浩 (Hao Yang)
Instance : Foundations of Signal Processing is a book written by Hao Yang . It is published by Science Press in 2008.
Encoder: BLSTM
《信号处理基础》是2008年科学出版社出版的图书，作者是杨浩。
Encoder: BLSTM-Res
《信号处理基础》是2008年科学出版社出版的图书，作者是杨浩。
Relation : 歌曲原唱 (song/singer/person)
Head entity: 多情的玫瑰 (Loving Rose)
Tail entity : 邓丽君 (Teresa Teng)
Instance : Loving Rose is a song sung by Teresa Teng.
Encoder: BLSTM
多情的玫瑰是邓丽君演唱的一首歌曲。
Encoder: BLSTM-Res
多情的玫瑰是邓丽君演唱的一首歌曲。

Figure 7. Attention analysis of '作者' and '歌曲原唱' relation.

5.4. Multi-instance Learning Analysis

In this section, we will analyze how distant supervision influences the classification result and how multi-instance learning can improve the performance of the models using our dataset. So, we analyze the classification result of each relation in the proposed model in word-level. The result is shown in Table 5.

From Table 5, we find that compared with label accuracy, the properties of relation itself are more important. For example, in '下辖地区' relation, the classification F1 score is 85.71 and 84.87 (Multi-instance Learning) even though the label accuracy is only 33.33%. By contrast, in '所属地区' the classification F1 score is 58.16 and 60.43 (Multi-instance Learning) although the label accuracy is 82.63%. It is in high probability caused by the uncertainty of the relations. In '所属地区' relation, the first entity may refer to a region, and it also can refer to an organization. Other relations, such as '类型' and '所属国家', have the same issue. When using the multi-instance learning method, the F1 score of the proposed model is improved from 87.30 to 87.89. After analyzing the improvement of each relation, we find that unlike our hypothesis before, which is the promotion of relations with low label accuracy is higher than that with higher label accuracy, the promotion is average, and it seems to be not related to the label accuracy.

Table 5. F1 score of each relation using proposed model in word level.

idx	Relation	Accuracy	F1	F1(MI)	idx	Relation	Accuracy	F1	F1(MI)
1	国籍	53.91%	81.06	82.87	16	下辖地区	33.33%	85.71	84.87
2	职业	85.86%	83.19	84.62	17	著名景点	61.56%	85.79	87.91
3	出生地	85.67%	94.01	94.12	18	制片地区	53.52%	68.37	70.68
4	主演	93.49%	98.73	98.96	19	性别	97.83%	99.06	99.35
5	类型	56.76%	42.45	42.38	20	编剧	53.14%	87.79	89.84
6	作者	92.47%	95.60	96.12	21	科	95.25%	98.78	98.92
7	所属地区	82.63%	58.16	60.43	22	歌曲原唱	79.12%	93.16	93.22
8	代表作品	88.01%	88.46	89.43	23	所属国家	85.02%	76.35	79.21
9	经营范围	68.49%	83.72	84.63	24	分布区域	70.41%	94.55	94.51
10	导演	89.86%	94.23	95.05	25	主要食材	83.25%	91.62	92.66
11	毕业院校	87.01%	94.40	93.96	26	登场作品	93.41%	92.31	93.39
12	运动项目	91.74%	98.22	98.94	27	常见症状	63.07%	95.75	94.05
13	总部地点	75.58%	83.29	82.98	28	所处时代	90.08%	95.41	95.92
14	民族	95.76%	99.07	98.86	29	所属运动队	86.46%	97.23	96.94
15	出版社	98.92%	99.31	99.30	30	隶属	51.71%	73.82	73.73

6. Conclusions and Future Work

Extra information that can not obtain directly from the sentence is verified to be helpful in relation extraction. The information used by previous work such as entity type obtained from NLP tools and knowledge bases all has their limitations. Many Chinese characters have unique meanings. Using the information provided by these characters can improve many tasks in Chinese language processing. In Chinese relation extraction, characters that constitute the entities can provide additional information. In this paper, we do several work to verify the effectiveness of this information.

First, to solve the problem of lacking dataset, we generate a dataset based on Baidubaike using distant supervision. Compared with previous datasets, our dataset is more appropriate for the large scale open domain Chinese relation extraction task. Second, we propose an attention-based model. By analyzing the attention mechanism, we find that using this information can effectively find out the vital part of the sentence. Furthermore, the model achieves the best performance among all tested models. Besides, we analyze the relationship between label accuracy and classification result. We find that the critical factor is the complexity of each relation instead of label accuracy.

When comparing with previous models and selecting the encoders, this paper mainly uses some representative model, rather than the latest state-of-the-art models. The reason is that by using some representative model, the effectiveness of introducing character information can be proved. Testing other models may be a supplement of our work and can be done in future work.

Author Contributions: Conceptualization, X.H.; Formal analysis, X.H.; Funding acquisition, T.H.; Methodology, X.H.; Project administration, T.H.; Software, X.H.; Supervision, T.H.; Writing–original draft, X.H.; Writing–review & editing, Y.Z., W.Z. and T.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, G.; Su, J.; Zhang, J.; Zhang, M. Exploring various knowledge in relation extraction. In Proceedings of the 43rd annual meeting on association for computational linguistics. Association for Computational Linguistics, Ann Arbor, MI, USA, 25–30 June 2005, pp. 427–434.
2. Surdeanu, M.; Tibshirani, J.; Nallapati, R.; Manning, C.D. Multi-instance Multi-label Learning for Relation Extraction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, 12–14 July 2012; pp. 455–465.

3. Socher, R.; Huval, B.; Manning, C.D.; Ng, A.Y. Semantic Compositionality through Recursive Matrix-Vector Spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, 12–14 July 2012; pp. 1201–1211.
4. Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation Classification via Convolutional Deep Neural Network. Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 11 August 2014; pp. 2335–2344.
5. Mintz, M.; Bills, S.; Snow, R.; Jurafsky, D. Distant Supervision for Relation Extraction without Labeled Data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2, Singapore, 2–7 August 2009; pp. 1003–1011.
6. Ji, G.; Liu, K.; He, S.; Zhao, J. Distant Supervision for Relation Extraction with Sentence-level Attention and Entity descriptions. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
7. Zeng, D.; Liu, K.; Chen, Y.; Zhao, J. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1753–1762.
8. Xu, K.; Feng, Y.; Huang, S.; Zhao, D. Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 536–540.
9. Xu, Y.; Mou, L.; Li, G.; Chen, Y.; Peng, H.; Jin, Z. Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1785–1794.
10. Liu, Y.; Wei, F.; Li, S.; Ji, H.; Zhou, M.; Houfeng, W. A Dependency-Based Neural Network for Relation Classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Beijing, China, 26–31 July 2015; pp. 285–290.
11. Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; Xu, B. Attention-based Bidirectional Long Short-term Memory Networks for Relation Classification. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Berlin, Germany, 7–12 August 2016; pp. 207–212.
12. Wang, L.; Cao, Z.; de Melo, G.; Liu, Z. Relation Classification via Multi-Level Attention CNNs. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 1298–1307.
13. Riedel, S.; Yao, L.; McCallum, A. Modeling Relations and Their Mentions without Labeled Text. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Dublin, Ireland, 10–14 September 2010; pp. 148–163.
14. Hoffmann, R.; Zhang, C.; Ling, X.; Zettlemoyer, L.; Weld, D.S. Knowledge-based Weak Supervision for Information Extraction of Overlapping Relations. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Portland, OR, USA, 19–24 June 2011; pp. 541–550.
15. Lin, Y.; Shen, S.; Liu, Z.; Luan, H.; Sun, M. Neural Relation Extraction with Selective Attention over Instances. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 2124–2133.
16. Qin, P.; Weiran, X.; Wang, W.Y. DSGAN: Generative Adversarial Training for Distant Supervision Relation Extraction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 496–505.
17. Chen, Y.J.; Hsu, J.Y.J. Chinese Relation Extraction by Multiple Instance Learning. In Proceedings of the PWorkshops at the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–13 February 2016.
18. Wen, J.; Sun, X.; Ren, X.; Su, Q. Structure Regularized Neural Network for Entity Relation Classification for Chinese Literature Text. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 365–370.

19. Li, Z.; Ding, N.; Liu, Z.; Zheng, H.; Shen, Y. Chinese Relation Extraction with Multi-Grained Information and External Linguistic Knowledge. In Proceedings of the 57th Conference of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 4377–4386.
20. Xu, J.; Wen, J.; Sun, X.; Su, Q. A discourse-level named entity recognition and relation extraction dataset for chinese literature text. *arXiv* **2017**, arXiv:1711.07010.
21. Le, H.T.; Cerisara, C.; Denis, A. Do Convolutional Networks need to be Deep for Text Classification? In Proceedings of the Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–3 Feburary 2018.
22. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
23. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2787–2795.
24. Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In Proceedings of the Twenty-ninth AAAI conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015.
25. Luo, B.; Feng, Y.; Wang, Z.; Zhu, Z.; Huang, S.; Yan, R.; Zhao, D. Learning with Noise: Enhance Distantly Supervised Relation Extraction with Dynamic Transition Matrix. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 430–439.
26. Wu, Y.; Bamman, D.; Russell, S. Adversarial Training for Relation Extraction. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; pp. 1778–1783.
27. Zhang, T.; Huang, M.; Zhao, L. Learning Structured Representation for Text Classification via Reinforcement Learning. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
28. Feng, J.; Huang, M.; Zhao, L.; Yang, Y.; Zhu, X. Reinforcement Learning for Relation Classification from Noisy Data. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
29. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
30. Zhang, D.; Wang, D. Relation classification via recurrent neural network. *arXiv* **2015**, arXiv:1508.01006.
31. Lin, Z.; Feng, M.; dos Santos, C.N.; Yu, M.; Xiang, B.; Zhou, B.; Bengio, Y. A Structured Self-attentive Sentence Embedding. *arXiv* **2017**, arXiv:1703.03130.
32. Yu, M.; Yin, W.; Hasan, K.S.; dos Santos, C.; Xiang, B.; Zhou, B. Improved Neural Relation Detection for Knowledge Base Question Answering. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 571–581.