IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 22, 2019, accepted September 18, 2019, date of publication September 24, 2019, date of current version October 10, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2943545

Multi-Channel CNN Based Inner-Attention for Compound Sentence Relation Classification

KAILI SUN[1], YUAN LI[1], DUNHUA DENG[2], AND YANG LI[1]
[1]School of Computer Science, Central China Normal University, Wuhan 430079, China
[2]Research Center for Language and Language Education, Central China Normal University, Wuhan 430079, China

Corresponding author: Yuan Li (yuanli@mail.ccnu.edu.cn)

This work was supported in part by the National Social Science Fund of China under Grant 18BYY174, and in part by the Ministry of Education Humanities and Social Sciences Research Planning Fund of China under Grant 14YJA740020.

ABSTRACT Relation classification is a vital task in natural language processing, and it is screening for semantic relation between clauses in texts. This paper describes a study of relation classification on Chinese compound sentences without connectives. There exists an implicit relation in a compound sentence without connectives, which makes it difficult to realize the recognition of relation. The major challenges that relation classification modeling faces are how to obtain the contextual representation of sentence and relation dependence features between clauses. To solve this problem, we propose a novel Inatt-MCNN model to extract sentence features and classify relations by combining multi-channel CNN and Inner-attention mechanism. This network structure utilizes CNN to extract local features of sentences and Inner-attention to capture sentence-level feature representations for this relation classification task. Besides, since the Inner-attention is based on Bi-LSTM, the global and long-term dependence semantic information can be well obtained in Inatt-MCNN to promote the model performance. We conduct experiments on two public Chinese discourse datasets: the Chinese compound sentence corpus (CCCS) dataset and the Tsinghua Chinese Treebank(TCT) dataset. Compared with the previous public methods, Inatt-MCNN model has superior performance and achieves the highest accuracy, especially on the CCCS dataset.

INDEX TERMS Relation classification, multi-channel CNN, inner-attention mechanism, Chinese compound sentence without connectives.

I. INTRODUCTION

Besides character, word, and phrase, sentence is also an important level of research in natural language processing(NLP) applications. Compound sentence has a complex sentence structure, which contains two or more clauses in it. Nearly two-thirds of Chinese sentences are compound sentences, so the study of sentence levels is essentially the study of compound sentences. Compound sentence relation classification is an indispensable part of compound sentence research, and it is also a basic research problem in the understanding of natural language. The main task of relation classification is to study logical semantic relationship between clauses. To correctly judge logical relation, semantic relation between clauses within sentence should be first understood effectively. Therefore, the research on compound sentence relation is beneficial to the development of other NLP related fields such as discourse analysis [1], information extraction [2], automatic question-answering [3] and machine translation [4].

According to whether there are connectives in sentences, the relation classification of compound sentences can be divided into two categories: explicit relation sentence with connectives and implicit relation sentence without connectives. As is shown in example 1, the compound sentence contains connective 因为(because), which denotes a causality relation in the sentence. A sentence like example 1 is referred to as an explicit relation sentence with connectives. However, in example 2, the sentence does not contain connectives but it can be inferred that the former clause is the reason of the latter clause according to semantic information of the two clauses. Therefore, there also exists a causality relation in this sentence. A sentence like example 2 is referred to as an implicit relation sentence without connectives.

The associate editor coordinating the review of this manuscript and approving it for publication was Sotirios Goudos.

Example 1: 因为今天天气十分晴朗, 所以全家决定去 郊外游玩。(“Because it is a fine day today, the whole family decided to go to the suburbs”.)

Example 2: 今天天气十分晴朗，全家决定去郊外游玩。(“It is a fine day today, so the whole family decided to go to the suburbs”.)

This paper mainly studies the relation classification of the implicit relation sentence without connectives. Since there exists implicit relation in such kind of compound sentences, it should be given primary consideration to obtain the overall semantic information representation and semantic associated feature between clauses. In traditional methods, the bag-of-words model [5] is used to represent the sentence, which is the most common and popular method, because it has high efficiency, simplicity, and surprising accuracy. However, the bag-of-words model maps sentences or texts to an unordered collection of words. Without considering the word order, different sentences may have exactly the same representation, because the same words are used.

Until now, some machine learning methods have achieved good results in sentence classification [6], [7]. But in recent years, deep learning has been widely used in speech recognition and computer vision. The sequence models based on neural networks such as recursive neural network(RNN), long short-term memory(LSTM) and attention mechanism are becoming mainly popular methods. The reason why scholars use them is that they can capture the order information of words and learn the semantic information at a deep level.

In this article, we propose a joint model Inatt-MCNN, which is a combination of the Inner-attention mechanism and multi-channel Convolutional neural network(CNN), to solve the sentence representation and relation classification task. The Inatt-MCNN as a novel model can fully perform the strengths of both architectures.

The main contributions of this paper are as follows:

1) The idea of multi-channel sentence representation is adopted, two kinds of word vector training methods, namely Word2Vec [8] and Glove [9], to encode sentences. This makes it more abundantly to express sentences.
2) We use the Inner-attention mechanism to capture the key part information of sentence, and the associated feature between clauses will be generated by combining together the extracted features of the former clause and the latter clause, which enables CNN to well capture feature information describing the degree of relevance between clauses.
3) CNN is used to capture local semantic information of n-grams in various grains through multiple convolution filters with different sizes, which is proved powerful for relation classification and does not rely on external parse trees.

The rest of the paper is organized as follows. Section II presents related works, including some classically traditional and recent popular methods of generic sentence classification. Section III introduces the background. Section IV describes our model architecture in detail. Section V outlines the experimental setup, and Section VI discusses the empirical results and analysis. Finally, Section VII presents the conclusion and future work.

II. RELATED WORK

A. TRADITIONAL METHODS

In Chinese information processing, experts and scholars mainly focus on relation classification of sentences, which is a basic research problem. Before deep learning is widely applied to NLP, researchers mostly use rules-based and statistical learning methods, combining with linguistic ideas. And a variety of text-based linguistic features classification methods have been proposed. In 2003, Yu and Hatzivassiloglou [10] adopted the method of adjective annotation to classify opinion sentences and achieved good results. In 2008, from the perspective of cognitive linguistics, Zhou and Yuan [11] chose the subject and predicate of the sentence and made time adverbs, orientation adverbs, and positional adverbs as linguistic features. And they proposed a discriminant method based on support vector machine(SVM), to recognize coordination and non-coordination in sentences. In 2010, Nakagawa *et al.* [12] proposed a sentence classification method based on the dependency syntax tree and CRF model with hidden variables. In 2010, Shu and Yang [13] summarized semantic features between clauses from syntactic and semantic perspectives and used semantic relevance theory to confirm the hierarchy ascription of clauses in Chinese compound sentences. In 2012, based on Naive Bayes, Wang *et al.* [14] proposed two classification models, MNB and NB-SVM, for emotional and topic sentence classification tasks. In 2017, Yang *et al.* [15] proposed an automatic identification method for relation categories of Chinese compound sentences based on semantic relevance calculation according to the connectives and collocation theory. All of these lay the foundation for the research on relation classification of Chinese compound sentences and promote further development in Chinese information processing.

B. DEEP LEARNING METHODS

The application of deep learning in NLP has promoted the study of sentence classification problems. In 2014, Kim [16] applied CNN to the sentence classification task for the first time, which made full use of the advantages of CNN feature self-extraction. And he gave several variants of models. In 2017, Zhou *et al.* [17] combined CNN with LSTM to propose the C-LSTM model, which firstly used CNN to extract local features of phrases in sentences and then used LSTM to obtain the features with inter-phrase dependency information. This model had produced good results in the sentence classification task. In 2018, Hassan and Mahmood [18] proposed a CNN and RNN joint model, which used local features extracted by CNN as the input of RNN for emotional sentence classification.

In general, sentence classification model based on word vector and deep learning can produce better results in most classification tasks than traditional methods based on the statistical and linguistic model. However, no previous study of relation classification in Chinese compound sentences has been conducted. Therefore, this paper adopts the implicit relation sentence without connectives as the research object. In the meantime, Fuyi [19] proposed that compound sentences can be divided into three relation categories: causality, coordination, and transition. This paper is going to conduct an experiment based on it. Experimental results demonstrate the feasibility of the proposed model.

III. BACKGROUND

A. CONVOLUTIONAL NEURAL NETWORKS

The CNN is widely used in image processing [20], target detection [21], and even medical discovery [22]. In recent years, it has been applied to NLP systems and accomplished quite brilliant results [16], [23], [24]. The difference between CNN and traditional neural networks is that a convolutional layer and a pooling layer are added between the input layer and the fully connected layer.

In NLP, convolutional layers act on the vector matrix through a sliding window. CNN can have numerous convolutional layers, which contains nonlinear activation functions such as tanh or ReLu. And different from classical feed-forward neural network, CNN in each layer can use different size kernels, which have hundreds or thousands of filters, over the input layer to compute the output. Then the output is composed to generate convolutional feature map. The pooling operation [25] in the pooling layer, which is applied to the feature map of a convolutional layer. Its idea is to get the most useful value for each feature map. Currently, the commonly used pooling operations include average pooling, max Pooling, and stochastic pooling.

In most NLP tasks, the input of CNN mainly contains sentences and documents, which are represented as a vector matrix. And each row of the matrix is usually a vector that represents a word or character. And the number of rows is usually a length of sentence or document. A filter slides overall words of the matrix. Therefore, the width of the filters is equal to the width of an input matrix. Weights of each convolution window are shared to greatly reduce the number of parameters. Another advantage of CNN is that CNN can simplify the process of text pre-processing. The text feature with high recognition is extracted in the convolutional layer, and the workload of feature engineering is reduced.

B. BI-DIRECTIONAL LSTM NETWORKS

In traditional neural network models, all inputs are independent of each other, which makes the model unable to learn the sequence information of texts. RNN [26] can propagate historical information through chain neural network architecture. However, RNN has the problem of exploding and vanishing gradient [27], [28], when the gap between two-time steps becomes large. LSTM [29] is a special type of RNN, which can well solve the problem of exploding and vanishing gradient of RNN. Bi-LSTM is an expanded model of LSTM. Recently, Bi-LSTM units have become a popular architecture in language models [30] and speech recognition [31]. Most of the sequence data is dependent on overall information. Different from LSTM, Bi-LSTM is a bidirectional network, which can process sequence in two directions, forward and backward. It can help get overall context information of the sequential data. Additionally, Bi-LSTM successfully realized the learning of long-term dependencies by introducing a memory cell that can preserve state over long periods of time. The LSTM transition functions are defined as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (2)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (3)$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$
$$h_t = o_t \otimes \tanh(c_t) \quad (5)$$

where x_t is input at the current time step, σ is a sigmoid function that has an output in [0,1]. $\otimes$ denotes element-wise multiplication. Tanh is a hyperbolic tangent function that has an output in$[-1,1]$. i_t is the input gate, which controls how much new information is stored in the current memory cell. f_t is the forget gate, which controls what extent the information from the previous memory cell is going to be forgotten. o_t is the output gate, which controls what to output based on the memory cell. c_t is cell state vector.

Based on LSTM, Bi-LSTM adds reverse information flow as shown in figure 1:

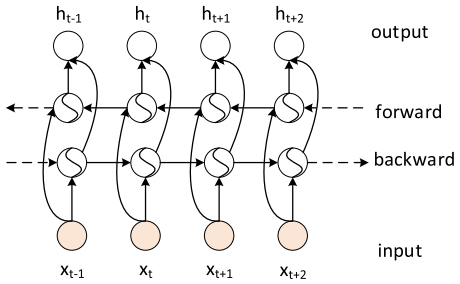


FIGURE 1. The Bi-LSTM framework.

The hidden states of forward and backward are obtained at each moment. Then connect them and get the finally bidirectional expression h_t :

$$h_t = \begin{bmatrix} \vec{h}_t \\ \overleftarrow{h}_t \end{bmatrix} \quad (6)$$

The vector h_t contains overall context information. Bi-LSTM takes the correlation between the previous and the latter moment into account and exhibits superior performance in the task of time series.

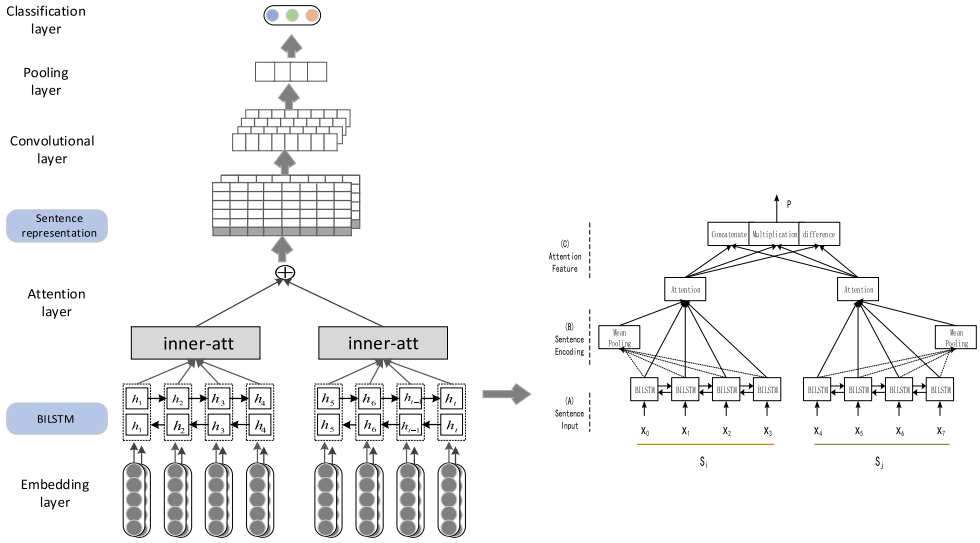


FIGURE 2. The proposed model framework.

IV. MODEL ARCHITECTURE

In this section, we show the details of the framework model, which consists of three main parts: sentence encoding stage, feature representation stage as well as the target classification stage.

A. THE EMBEDDED LAYER

In the absence of a large scale supervised training set, initializing word vectors with word vectors obtained from unsupervised neural language models is a popular method to improve performance [32], [33]. Recently, many studies have shown that the accuracy of the model can be improved by performing unsupervised, pre-trained word embedding.

The first layer transforms words into real-valued vectors that network can recognize and capture semantic information. We adopt two kinds of word embedding methods, Word2vec, and Glove, which are pre-trained on one million sentences from Chinese Wikipedia news. Firstly, language technology platform cloud LTP[①] is used to segment words for a compound sentence to get a segmented word sequence $[x_1, x_2, \ldots, x_n]$, with each word being derived from vocabulary V. Secondly, words are represented by distributed vectors $w \in R^{1\times d}$ and $g \in R^{1\times d}$, which are the d-dimensional vectors produced from Word2Vec and Glove respectively. Finally, words of every sentence are looked up in word embedding matrices $W \in R^{1\times|v|}$ and $G \in R^{1\times|v|}$.

Compound sentence representation is shown as follows :

$$w_{1:n} = w_1 \oplus w_2 \oplus w_3 \ldots \oplus w_n \tag{7}$$

$$g_{1:n} = g_1 \oplus g_2 \oplus g_3 \ldots \oplus g_n \tag{8}$$

Here, $\oplus$ is the cascade operator. In general, $w_{1:n}$ refer to the cascading of words $w_i, w_{i+1}, \ldots w_{i+j}$ and $g_{1:n}$ refer to the cascading of words $g_i, g_{i+1}, \ldots g_{i+j}$. n is the length of the sentence.

B. THE ATTENTION LAYER

In this layer, we use inner-attention mechanism [34], which is an improved model of attention mechanism. Bi-LSTM is added to this model. Therefore it will have both advantages, which can help to better capture bidirectional semantic dependence information and distribute different weight to words according to their importance in sentences.

A compound sentence is divided into the former clause and the latter clause as input in this layer. Bi-LSTM is adopted to encode sentence. And then average pooling layer is used on top of word-level Bi-LSTM to generate sentence feature representation. At last, attention weights for words are distributed. The inner-attention mechanism is formalized as follows :

$$M = \tanh(w^y Y + w^h R_{ave} \times e_L) \tag{9}$$

$$\alpha = soft\max(w^T M) \tag{10}$$

$$R_{att} = Y\alpha^T \tag{11}$$

where Y is a vector matrix obtained from Bi-LSTM, R_{ave} is an output of the average pooling layer. α is an attention weight matrix and R_{att} is an attention-weighted sentence representation.

Then an "associated vector" feature p will be generated by combining together R^{c1}_{att} and R^{c2}_{att}. Feature p denotes the associated information between clauses as follows:

$$p = (R^{c1}_{att} \oplus R^{c2}_{att}) \oplus (R^{c1}_{att} \cdot R^{c2}_{att}) \oplus (R^{c1}_{att} \Theta R^{c2}_{att}) \tag{12}$$

where R^{c1}_{att} and R^{c2}_{att} refer to attention-weighted representations of former clause and latter clause,$\oplus$ is a concatenation operator, $\cdot$ is element-wise multiplication operator, and Θ is element-wise difference operator.

C. THE CONVOLUTIONAL LAYER

In the convolutional layer, we use the output vector p of attention layer to produce different sentence representations

w' and g' as follows :

$$w' = w \oplus p \tag{13}$$

$$g' = g \oplus p \tag{14}$$

Here $\oplus$ is a concatenation operator, with $w' \in R^{n\times 2d}$, $g' \in R^{n\times 2d}$.

Then w' and g' are fed into the convolutional layer. And convolutional filters are $W_i^w \in R^{h_i\times 2d}$ and $W_i^g \in R^{h_i\times 2d}$ corresponding to sentence representations w' and g', where h_i is the size of each filter with $i \in [0, 3]$. Feature F_k^w and F_k^g are generated from a window of words $w'_{k:k+h-1}$ and $g'_{k:k+h-1}$ by :

$$F_k^w = f(W_i^w \cdot w'_{k:k+h_i-1} + b_i) \tag{15}$$

$$F_k^g = f(W_i^g \cdot g'_{k:k+h_i-1} + b_i) \tag{16}$$

where $b_i \in R$ is a bias term, f is an activation function such as rectified linear unit(ReLu). These filters are applied to each possible window of words in the compound sentence $\{w'_{1:h_i}, w'_{2:h_i+1}, \ldots w'_{n-h_i+1:n}\}$ and $\{g'_{1:h_i}, g'_{2:h_i+1}, \ldots g'_{n-h_i+1:n}\}$ to produce a feature map:

$$F^w = \{F_1^w, F_2^w, \ldots F_{n-h_i+1}^w\} \tag{17}$$

$$F^g = \{F_1^g, F_2^g, \ldots F_{n-h_i+1}^g\} \tag{18}$$

With $F^w \in R^{n-h_i+1}$, $F^g \in R^{n-h_i+1}$, in the experiment, more feature maps can be extracted by multiple filters of different sizes, which makes the feature information more abundant.

D. THE POOLING LAYER

In this layer, we apply a max-over-time pooling operation [35] to feature map. The maximum value is taken in the corresponding feature map of each filter, as shown in $\hat{F}^w = \max(F^w)$ and $\hat{F}^g = \max(F^g)$. Then the $\hat{F}^w$ and $\hat{F}^g$ are represented as :

$$\hat{F}^w = \{\hat{F}_1^w, \hat{F}_2^w, \ldots \hat{F}_{num\_w}^w\} \tag{19}$$

$$\hat{F}^g = \{\hat{F}_1^g, \hat{F}_2^g, \ldots \hat{F}_{num\_g}^g\} \tag{20}$$

Here *num_w* and *num_g* are the number of each type filters of w' and g'. Then $\hat{F}^w$ and $\hat{F}^g$ are connected to generate the feature representation Q as follows :

$$Q = \hat{F}^w \oplus \hat{F}^g \tag{21}$$

With $\hat{F}^w \in R^{num\_w}$, $\hat{F}^g \in R^{num\_g}$, $Q \in R^{num\_w+num\_g}$.

E. THE CLASSIFICATION LAYER

The classification layer is, in principle, a logistic regression classifier. The feature representation vector Q is input into this layer, followed by a softmax function to calculate the predictive probabilities distribution for all categories [32]. The classification result is p_i as follows :

$$z = w_s Q + b_s \tag{22}$$

$$p_i = \frac{\exp(z_i)}{\sum_{j=1}^{c} \exp(z_j)} \tag{23}$$

TABLE 1. Statistical description of datasets.

Dataset	Set	Sentence	Maximum length
CCCS	Train	19629	63
	Validation	6543	
	Test	6543	
TCT	Train	6946	54
	Validation	2315	
	Test	2315	

With $w_s \in R^{c\times(num\_w+num\_g)}$, $b_s \in R^c$. w_s and b_s are the weight and the bias term. We assume that there are c categories in our experiment.

V. EXPERIMENTAL SETUP

A. RELATIONSHIP CLASSIFICATION DATASETS

The performance of the proposed model is evaluated on two public Chinese discourse datasets: the Chinese compound sentence corpus(CCCS) [36] and the Tsinghua Chinese Treebank(TCT) [37], which are presently the datasets that mainly contain Chinese compound sentences.

B. THE DATASET I ––CCCS

The CCCS is a public dataset, which is a benchmark for Chinese compound sentence research, which contains more than 658,447 sentences mainly from publications of *Changjiang Daily* and *People's Daily*. We select 32,715 Chinese compound sentences without connectives from the corpus as the experimental data in this paper. And the dataset is randomly divided into training, verification and test sets by a ratio of 6: 2: 2.

C. THE DATASET II––TCT

The TCT is a corpus with one million words mainly from the authentic text of modern Chinese in the 1990s. It is divided into four categories: the literature, the academic, the news and the application. The proportions of literature, academic, news and application are 7.73%, 26.3%, 20.0% and 6.4%, respectively. In our experiment, the causality category, coordination category, and transition category are selected from the corpus, and a total of 11,577 sentences are used as the final experimental dataset. In Table 1, we present the details of the two datasets.

D. HYPERPARAMETERS AND TRAINING

This paper uses the pre-trained word vector from Chinese Wikipedia news. Its dimension is set to 300, and for words that do not exist in vocabulary, we initialize them from a uniform distribution [-0.25, 0.25]. As the convolutional layer in our approach requires fixed-length input, we use n to denote the maximum length of sentence in the dataset, and we pad sentences, which have a length less than n with zero vector at the end that indicate the unknown words. However,

TABLE 2. Parameter settings for model.

Parameter	CCCS	TCT
embedding_size	300	-
max_length	63	54
hidden_size	256	-
learning rate	0.0001	0.0005
dropout	0.5	-
l2_constrain	1.0	-
batch_size	50	20
filter_num	200	-
filter_size	(3,4,5,6)	-

the sentence that has a length longer than n, we simply cut extra words in the end to reach n.

The dimension of the hidden layer in Bi-LSTM is set to 256. In the convolution layer, multiple convolution filters with width are set to 3, 4, 5 and 6. The number of each type filters is 200, which is conducive to extracting more detailed feature information.

To avoid co-adaption in the experiment, the dropout strategy is used, and its value is set to 0.5. And the regularization with L2-norm [38] is added to the objective function of the experiment to improve the performance. In table 2, we show the set of primary hyperparameters for the proposed architecture.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

A. OPTIMIZATION

In this paper, we adopt the Adadelta updating criterion [39] and use the Stochastic Gradient Decline [40] to train the network over shuffled mini-batches. In this experiment, the back Propagation [41] algorithm is used to compute the gradient. Samples are split randomly in training, validation and test subset. We set the same size of validation as the corresponding test size. And the model is optimized by minimizing the cost function, which contains cross-entropy [42] and a constraint on l2-norms of the weight vectors. A loss function *Loss* is the cross entropy between the ground truths and predictions, which is defined as:

$$Loss = -\sum_{i}\sum_{j}\hat{y}_i^j \log y_i^j + \sum_{\theta\in\Theta}\lambda_\theta ||\theta||_2^2 \qquad (24)$$

where y is the predicted value for a given target, $\hat{y}$ is the actual value, i is the index of the sentence, j is the category index. Θ represents the set of parameters, λ_θ are the hyperparameters, the first term is the cross-entropy error, and the second term is a L2-regularization penalty.

B. BASELINE METHODS

To fully estimate the effectiveness of our model, we compare it with several strong baseline methods for relation classification task of the compound sentence :

- SVM [11]: It uses SVM and chooses main syntactic features such as subject, predicate, time adverb and orientation adverb of the sentence. Then the features are quantified according to CNKI. We adopt the SMO optimization algorithm in the experiment.
- Semantic-Relevancy [15]: It uses the syntactic theory of and the collocation theory of connectives. We calculate the value of semantic relevance for different collocation of connectives. The relation of a sentence is demonstrated by connectives corresponding to maximum value semantic relevancy.
- CNN [16]: This method uses a simple CNN to classify sentences and use word vectors as input. We choose the pre-trained word vector as input. Filters are set to [3]–[5], the number of each kind filter is set to 200 and dropout is set to 0.5.
- CNN-SVM [42]: It combines deep learning model CNN with traditional algorithm model SVM, and uses transfer learning as the idea to input extracted text features by CNN into SVM to classify compound sentences. The parameters contain that filters are set to [4]–[6], the number of each window size is set to 100 in CNN. And C is selected as 0.1 in SVM.
- ATT-CNN [43]: It uses attention mechanism to capture long-term dependence information and correlation between nonconsecutive words automatically and then sents them to CNN. The parameters are the same with [10] in CNN.
- C-LSTM [17]: It utilizes CNN to extract a sequence of higher-level phrase representations, then which are fed into an LSTM to get sentence representation. This model combines obtained local features and semantic information features to realize sentence classification. We set main parameters that the number of filters is set to 300, dropout is set to 0.5 and L2 regularization with a factor is 0.001.
- CCLA [44]: It uses CNN to capture local feature information. And the long-distance dependence information is obtained by using Bi-LSTM. CNN and Bi-LSTM are on the same layer. Then input the output of Bi-LSTM into attention mechanism to capture the critical words of sentence. Final sentence representation is sent to a fully connected layer, which is followed by a softmax function to classify sentence. Filters are set to [3]–[5], and dropout is 0.5.
- Inatt-CNN: It is a variant of our model. We use an inner-attention mechanism to capture the important words of the sentence and learn the overall semantic representation of sentence. And finally, the output of the previous procedures is fed to CNN for relation classification.

C. EXPERIMENTAL RESULTS AND ANALYSIS

In order to fully compare the differences of models and analyze the impact of each component on the experimental results, we show in Table 3 the specific evaluation indicators for each model.

TABLE 3. The performance of our approach compared to other approaches on the two dataset.

Model	ACC	
	CCCS	TCT
SVM[11]	79.31%	75.56%
Semantic-Relevance[15]	81.58%	77.02%
CNN-rand[16]	85.01%	84.12%
CNN-static[16]	85.06%	84.35%
CNN-non-static[16]	86.12%	85.71%
CNN-SVM[42]	85.82%	84.92%
ATT-CNN[43]	83.94%	81.64%
C-LSTM[17]	88.24%	86.43%
CCLA[44]	90.03%	88.74%
Inatt-CNN	92.64%	89.27%
Our approach	94.21%	90.10%

As can be seen from Table 3, the proposed model in this paper has superior performances over several other baseline methods. In terms of accuracy, deep learning methods increase by 14.9% and 12.54% compared with traditional methods on CCCS and TCT dataset respectively. The experimental results show that the multi-channel CNN network in our approach Inatt-MCNN is superior, which is 1.57% higher than Inatt-CNN on CCCS dataset. In our approach, we utilize the idea of the multi-channel, which can help to realize the multi-angle and multi-directional expression of word information. Therefore, more abundant semantic feature information will be captured and the effect of complementary information between words will be achieved.

Furthermore, there is a characteristic of time series in the study of natural language. Words are isolated in the sentence, and their order is not considered in a CNN network structure. However, LSTM has a better performance in solving the problem of time series. LSTM network has the ability to capture word order information and further learn semantic features from sentences. Therefore, the LSTM is added to CNN. Experimental results in TABLE 3 demonstrate that the accuracy has increased by 2.85% and 1.70% compared with the average accuracy of three type CNN models, on CCCS and TCT dataset. The accuracy of CCLA model is 1.79% and 2.31% higher than C-LSTM model on CCCS and TCT. Bi-LSTM and attention mechanism are added in CCLA. Bi-LSTM can capture the dependence information between words and attention mechanism can find the key part of sentence. Therefore, it can seen that they are beneficial to improve the performance of model.

TABLE 4. The performance of our approach compared to other approaches on the two dataset.

Model	F1-macro	
	CCCS	TCT
SVM[11]	74.84%	72.32%
Semantic-Relevance [15]	79.81%	85.08%
CNN-rand[16]	81.57%	80.79%
CNN-static[16]	82.28%	82.36%
CNN-non-static[16]	84.63%	84.05%
CNN-SVM[42]	83.73%	83.41%
ATT-CNN[43]	80.06%	79.35%
C-LSTM[17]	86.01%	83.74%
CCLA[44]	88.52%	86.02%
Inatt-CNN	89.09%	90.43%
Our approach	91.74%	88.36%

Lastly, we try to introduce the inner-attention mechanism since it not only distributes different attention weight to words to find the key words information but also captures semantic associated vector features between clauses of sentences. Our approach use inner-attention mechanism instead of attention mechanism, and experimental results show the effectiveness of our model, which provides a 4.18% improvement in accuracy over the CCLA method on CCCS.

Additionally, it can be seen from experimental results that adding the traditional model SVM based on CNN has a similar effect with using CNN alone, with a difference of 0.3%-0.81% in accuracy on CCCS and TCT dataset but the model complexity increases. Therefore, we need to adjust the structure of model according to the performance in different dataset and research objective.

From experimental results of ATT-CNN model, the attention mechanism is added to CNN, but the accuracy decrease a little about 1.45% compared with the average accuracy of three type CNN models on CCCS. But in the CCLA model, the accuracy is improved by 4.64% after adding Bi-LSTM network. It means that strong semantic learning of sentence must be ensured when adding attention mechanism. Therefore, blindly adding the attention mechanism may cause the loss of semantic information of original sentence representation. Similarly, in Table 4, it can be seen that the F1-macro score of our model is higher than that of other baseline methods. In figure 3 below, we can see the results comparison of all models on datasets more clearly.

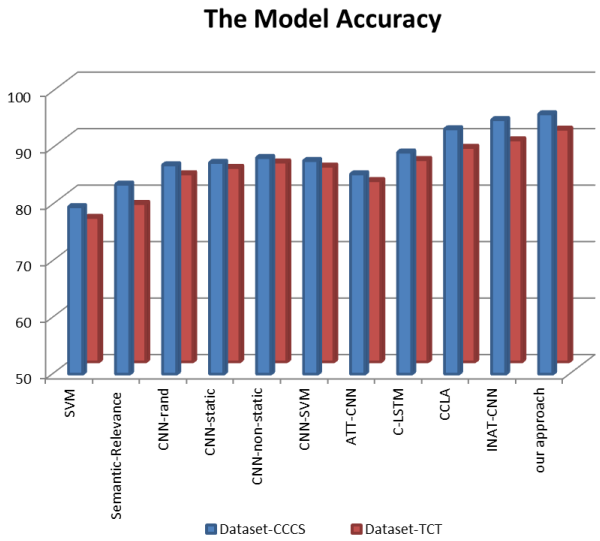


FIGURE 3. The results comparison of datasets.

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose a joint model Inatt-MCNN for relation classification modeling. This model can capture local and bidirectional long-term semantic dependencies information within sentences. More than that, this model can also obtain an associated feature, which contains associated information between clauses. Experiments show that the Inatt-MCNN model outperforms several superior baseline methods and realize expected classification results. In future work, we would modify the structure of the neural network model according to semantic rules of Chinese sentences, and study the deep-level semantic information and associated information within sentences. At the same time, we would use superior performance computing methods to get a faster training speed.

REFERENCES

[1] Y. Liu, S. Li, X. Zhang, and Z. Sui, "Implicit discourse relation classification via multi-task neural networks," *13th AAAI Conf. Artif. Intell.*, Mar. 2016, pp. 1507–1514.

[2] Y. Meng, and A. Rumshisky, "Context-aware neural model for temporal information extraction," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 527–536.

[3] S. Minaee and Z. Liu, "Automatic question-answering using a deep similarity neural network," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2017, pp. 923–927.

[4] J. Gehring, M. Auli, D. Grangier, and Y. N. Dauphin, "A convolutional encoder model for neural machine translation," 2016, *arXiv:1611.02344*. [Online]. Available: https://arxiv.org/abs/1611.02344

[5] T. K. Landauer and S. T. Dutnais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge," *Psychol. Rev.*, vol. 104, no. 2, pp. 211–240, 1997.

[6] J. Zhao, K. Liu, and G. Wang, "Adding redundant features for CRFs-based sentence sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2008, pp. 117–126.

[7] B. Rink, and S. Harabagiu, "UTD: Classifying semantic relations by combining lexical and semantic resources," in *Proc. 5th Int. Workshop Semantic Eval.*, 2010, pp. 256–259.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[9] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[10] H. Yu and V. Hatzivassiloglou, "Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2003, pp. 129–136.

[11] W. Zhou and C. Yuan, "Automatic recognizing of complex sentences with coordinating relation," *Appl. Res. Comput.*, vol. 25, no. 3, pp. 764–766, Mar. 2008.

[12] T. Nakagawa, K. Inui, and S. Kurohashi, "Dependency tree-based sentiment classification using CRFs with hidden variables," in *Proc. Hum. Lang. Technol. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2010, pp. 786–794.

[13] J. Shu and J. Yang, "The computing method of semantic relevancy of clauses in Chinese compound sentence," in *Proc. Int. Symp. Intell. Inf. Process. Trusted Comput.*, Oct. 2010, pp. 282–285.

[14] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguistics*, 2012, pp. 90–94.

[15] J. Yang, Z. Chen, X. Shen, and J. Hu, "Automatic recognition of relation category of non-saturated compound sentences with two clauses," *Appl. Res. Comput.*, vol. 34, no. 10, pp. 2950–2953, Oct. 2017.

[16] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: https://arxiv.org/abs/1408.5882

[17] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, "C-LSTM neural network for text classification," 2015, *arXiv:1511.08630*. [Online]. Available: https://arxiv.org/abs/1511.08630

[18] A. Hassan and A. Mahmood, "Convolutional recurrent deep learning model for sentence classification," *IEEE Access*, vol. 6, pp. 13949–13957, 2018.

[19] X. Fuyi, *Chinese Compound Sentence Research [M]*. Beijing, China: The Commercial Press, 2001.

[20] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "Robust 3D hand pose estimation in single depth images: From single-view CNN to multi-view CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3593–3601.

[21] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster R-CNN for object detection in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3339–3348.

[22] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke, "The rise of deep learning in drug discovery," *Drug Discovery Today*, vol. 23, no. 6, pp. 1241–1250, 2018.

[23] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 649–657.

[24] Y. Xiao and K. Cho, "Efficient character-level document classification by combining convolution and recurrent layers," 2016, *arXiv:1602.00367*. [Online]. Available: https://arxiv.org/abs/1602.00367

[25] B. Fernando, E. Gavves, J. Oramas M. A. Ghodrati, and T. Tuytelaars, "Rank pooling for action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 773–787, Apr. 2017.

[26] K. Kawakami, "Supervised sequence labelling with recurrent neural networks," Ph.D. dissertation, Stud. Comput. Intell., Berlin, Germany, 2008.

[27] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

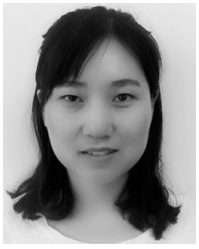
[28] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, 1998.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] T. Mikolov, A. Deoras, D. Povey, L. Burget, and J. H. Černocký, "Strategies for training large scale neural network language models," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2011, pp. 196–201.

[31] Y. Zhao, S. Takaki, H.-T. Luong, J. Yamagishi, D. Saito, and N. Minematsu, "Wasserstein GAN and waveform loss-based acoustic model training for multi-speaker text-to-speech synthesis systems using a WaveNet vocoder," *IEEE Access*, vol. 6, pp. 60478–60488, 2018.

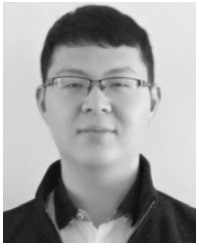
[32] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 151–161.
[33] R. Collobert, "Deep learning for efficient discriminative parsing," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, vol. 15, 2011, pp. 224–232.
[34] Y. Liu, C. Sun, L. Lin, and X. Wang, "Learning natural language inference using bidirectional LSTM model and inner-attention," 2016, *arXiv:1605.09090*. [Online]. Available: https://arxiv.org/abs/1605.09090
[35] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," 2011, *arXiv:1103.0398*. [Online]. Available: https://arxiv.org/abs/1103.0398
[36] F. Xing and S. Yao, "Construction and Utilization of Chinese Compound Sentence Corpus," in *Proc. Hnc Linguistics Res. Symp.*, 2005, pp. 432–439.
[37] Q. Zhou, "Annotation scheme for Chinese treebank," *J. Chin. Inf. Process.*, vol. 18, no. 4, pp. 2–9, Feb. 2004.
[38] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, *arXiv:1207.0580*. [Online]. Available: https://arxiv.org/abs/1207.0580
[39] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701
[40] N. Ketkar's, *Stochastic Gradient Descent*. London, U.K.: Optimization, 2014, pp. 113–132.
[41] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, 2005.
[42] Y. Cao, R. Xu, and T. Chen, "Combining convolutional neural network and support vector machine for sentiment classification," in *Social Media Processing*. Singapore: Springer, 2015, pp. 144–155.
[43] Z. Zhao and Y. Wu, "Attention-based convolutional neural networks for sentence classification," in *Proc. INTERSPEECH*, 2016, pp. 705–709.
[44] Y. Zhang, J. Zheng, Y. Jiang, G. Huang, and R. Chen, "A text sentiment classification modeling method based on coordinated CNN-LSTM-attention model" *Chin. J. Electron.*, vol. 28, no. 1, pp. 120–126, Jan. 2019.



KAILI SUN received the degree from the School of Computer Science, Central China Normal University, where she is currently pursuing the master's degree. Her current research interests include machine learning, deep learning, and natural language processing.

YUAN LI received the Ph.D. degree from the Huazhong University of Science and Technology, China. He is currently an Associate Professor with the Department of Computer Science and Technology, Central China Normal University. His research fields are natural language processing and Chinese information processing.

DUNHUA DENG received the master's degree from Central China Normal University, China, in 2004, where she is currently pursuing the Ph.D. degree. She is also an Associate Professor with the Department of Computer Science and Technology, Hubei University of Economics. Her main research fields include natural language processing and Chinese information processing.



YANG LI received the degree from Central China Normal University, where he is currently pursuing the master's degree. His current research interests include machine learning, deep learning, and natural language processing.