ROBOT INTENT RECOGNITION METHOD BASED ON STATE GRID BUSINESS OFFICE CONFERENCE SUBMISSIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Artificial intelligence is currently in an era of change, not only changing the artificial intelligence technology itself, but also changing human society. It has become more and more common to use artificial intelligence as the core human-computer interaction technology to replace manpower. Intention recognition is an important part of the human-machine dialogue system, and deep learning technology is gradually being applied to the task of intent recognition. However, intent recognition based on deep learning often has problems such as low recognition accuracy and slow recognition speed. In response to these problems, this paper designs a BERT fine-tuning to improve the network structure based on the pre-training model and proposes new continuous pre-training goals. To improve the accuracy of intent recognition, a method based on multi-teacher model compression is proposed to compress the pre-training model, which reduces the time consumption of model inference.

1 INTRODUCTION

With the development and progress of society, the service industry, which belongs to the tertiary industry, has developed rapidly. Data from 2018 shows that the tertiary industry accounts for approximately 59.7% of my country's GDP, which exceeds the sum of the primary and secondary industriesof Statistics. This shows that modern cities are increasingly inseparable from the tertiary industry, and people's demand for service industries is increasing. However, due to the rapid development of the service industry, the level of professional accomplishment of the personnel involved in the industry is not uniform, resulting in poor experience of people and a huge gap in the service industry. Therefore, people in modern society have a huge demand for a human-machine dialogue system that can respond in a timely manner and have excellent experience. Human beings have been pursuing to build a practical and reliable intelligent human-computer oral dialogue system. Jason D. Williams regards the speech-based dialogue system as a kind of computer interaction intermediary, which is mainly realized through human-computer interaction and speech understandingWilliams (2009).

Modern human-machine dialogue systems mainly use task-oriented dialogue systems Chen et al. (2017), in which oral comprehension is the core component of the dialogue system. Its goal is to convert natural language text output by speech recognition into computer-understandable text. The structured semantic representation is the entrance of the dialogue system interaction, which determines the effectiveness of the subsequent human-computer interaction of the dialogue system, and determines the user's experience of the human-machine dialogue system from the very beginningTur & De Mori (2011). Intention recognition is a key part of human-computer interaction, and it plays an important role in understanding the expression of user needs and giving feedback to the user.

Intention recognition is the entry point in the human-computer interaction dialogue. Intention recognition is to understand what the user wants to do. The form of intent is generally action executor + verb + noun, such as residents applying for electricity meters, business transfers and renaming, but of course it is not necessarily in this form, and changes according to specific circumstances. Intent recognition is often done in intent classification, that is, according to what the user said, it is classified into a pre-defined intent categoryCelikyilmaz et al. (2011). Intentions are sometimes called

conversational behaviorsAustin et al. (1975), indicating that users share certain information during a conversation and are constantly updated.

Intent recognition technology uses traditional machine learning or deep learning algorithms, and the accuracy is often not high enough, so in the actual landing process, the dialogue system misunderstands the user's intent, resulting in a bad user experience. Therefore, how to improve the accuracy of intention recognition is the focus of this article. In recent years, the pre-training model technology in natural language processing has shined. Through pre-training and fine-tuning the two-stage training method, the accuracy of text classification and intent recognition has been effectively improved.Devlin et al. (2019). But at the same time, due to the large size and high computational load of the pre-training model, the hardware requirements for training equipment are relatively high. However, human-machine dialogue systems are often deployed in embedded devices with poor hardware environments, and human-machine dialogue systems have high requirements for real-time performance. It is not a good choice to directly use pre-training models as models for intent recognition. Through model compression, such as knowledge distillation, the inference speed of the pre-trained model can be effectively improved, so that the model with high accuracy can be better implemented in the actual industrial sceneHinton et al. (2015).

In a power business hall, users usually know what their needs are, but they don't know the specific types of business hall business corresponding to the needs. In this paper, by cooperating with the electric power business office, we collected data sets of intention recognition in real scenarios, and deeply analyzed the application of deep learning technology to improve the effect of intention recognition.

In summary, under the background that the public's demand for the service industry is increasing year by year, the use of pre-trained models and model compression can effectively improve the accuracy of intent recognition in the human-machine dialogue system to solve people's needs for travel. Communication needs. It is of far-reaching significance for reducing the employment cost of enterprises and merchants and improving the user experience of customers.

2 Related Work

2.1 Convolutional neural network

Convolutional neural network was originally proposed from the research of computer vision, and it is an end-to-end model of deep learning. The idea of convolutional neural network is to operate the input and convolution kernel. The convolution kernel can only pay attention to the local area. This process is called feature extraction, and the extracted features are called feature maps. Convolutional neural networks are the first to shine in the image field, such as handwriting recognitionPoznanski & Wolf (2016); Abdel-Hamid et al. (2012), object detectionParkhi et al. (2015), etc. In 2014, Kim et al. proposed the Text-CNN model, which used the convolution technology in the direction of computer vision in the task of natural language processing, and achieved very good results. It has avoided the cumbersome manual extraction of features, simple structure, and calculation The advantage of lower complexity. The Text-CNN model architecture is shown in figure 1.

2.2 Recurrent neural network

Recurrent neural network is a network that can remember information for a certain period of time. The structure of the cyclic neural network is very simple, and it mainly propagates forward with the continuous input of the sequence. The parameters of the recurrent neural network are updated by a backpropagation algorithm with time called **?** which is improved by the backpropagation algorithm. The algorithm transmits the error information to the previous neuron step by step in the reverse order of the time step. When the length of the text sequence to be learned is relatively large, the value of the gradient will be empty or huge. The core structure of RNN is shown in Figure 2.

2.3 Transformer

Transformer is a seq2seq model structure Vaswani et al. (2017) proposed by Google in 2017, which is mainly used to solve machine translation problems. Compared with previous deep learning meth-

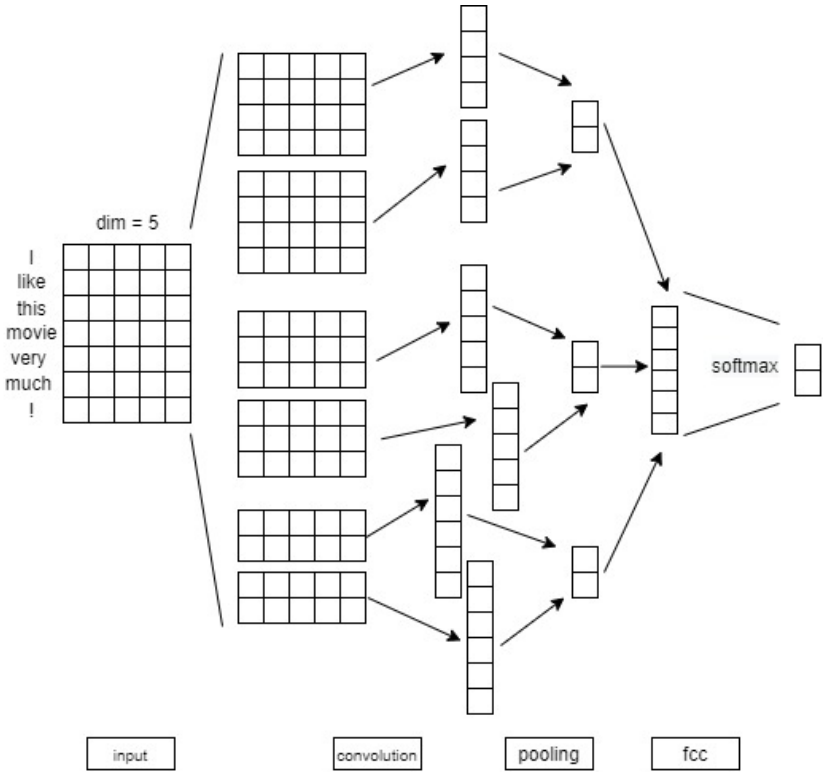

Figure 1: Text-CNN model architecture

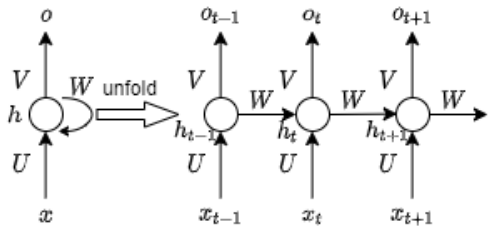

Figure 2: RNN's core structure

ods, Transformer completely abandons traditional feature extractors, such as CNN and RNN, and uses a self-attention mechanism. The specific structure diagram is shown in figure 3.

The essential structure of Transformer is a sequence-to-sequence model, so it can be divided into encoder and decoder. In the specific training process, the output of the last layer of the encoder part is sent to the decoder as an input, and combined with each layer of the decoder to calculate the final result.

In Transformer, position coding is used to encode the position of each word in the training data. Position coding allows Transformer to learn the word order information of each word in a sentence. The specific method is to encode the word order with sine and cosine functions. Such an encoding method makes the position periodic and therefore produces a relative position relationship. So for an offset n, the value of $PE[pos + n]$ is equal to $PE[pos]$. After the position vector is obtained, it is superimposed with the word representation vector. The calculation equation of the position code is shown in equations (1) and (2):

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}}) \tag{1}$$

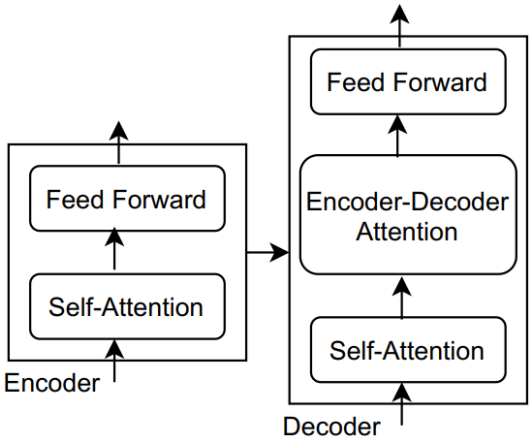

Figure 3: Transformer network structure

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}}) \tag{2}$$

Self-attention mechanism and multi-headed attention machine are the main "heritors" for feature extraction in Transformer. They can obtain long-distance dependence information between words in the text. The multi-head attention mechanism is the integration of multiple self-attention structures, using multiple "heads" to learn different characteristics. The calculation process of the attention mechanism is expressed as mapping $query$ and key and $value$ pairs to the output. In actual use, there are multiple queries. In order to perform attention operations on these queries at the same time, these queries are assembled into a matrix Q. In the same way, we can get K and V from keys and values. Because it is self-attention, here Q, K, and V are all vectors in the same input, and the final output is the weighted sum of all values. The specific calculation process is shown in equation (3).

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

In addition, compared with the general attention operation method, there is an additional zoom operation-the dot product of query and key is divided by $\sqrt{d_k}$**?**, the purpose is to avoid dot product The result is too large and enters the saturated area of softmax. d_k represents the dimensions of the Q and K vectors. The execution process of the zoom dot product attention mechanism is shown in Figure 4.

3 TARGET TASK IMPROVEMENT BASED ON CONTINUED PRE-TRAINING

Aiming at the problems of BERT's pre-training goal that NSP is not necessarily effective, pre-training data and downstream task data are inconsistent, two new pre-training goals are designed, namely the language model task and the natural language inference task.

Continue pre-training refers to continuing pre-training on the model that has been pre-trained. Continued pre-training on the fine-tuning task data set can often further improve the effect of the model during fine-tuning. In this paper, the open source pre-training model is continued to be pre-trained on the downstream power intent data. The pre-training target tasks used in the original BERT are the masking language model MLM and the next sentence prediction task NSP. Unlike BERT, this article proposes two new pre-training targets, including language model targets and natural language inference. the goal. The following describes the pre-training target tasks used in this article.

3.1 LANGUAGE MODEL TARGET TASK

This paper uses the idea of language model to design a target task for continuing pre-training. The target task is to use the language model to calculate the probability of the next word, optimize the model during training through the log-likelihood function, and let the model do the language model task in the continued pre-training. The objective function is shown in equation (4).

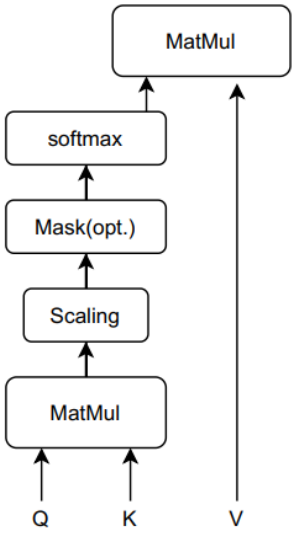

Figure 4: Attention Mechanism

$$\sum_{k=1}^{n} \log p(w_k \mid w_i, \ldots, w_{k-1}; \Theta_{BERT}). \tag{4}$$

Where Θ_{BERT} represents the parameters of the BERT model, and $w_1, w_2, w_3, w_4, ..., w_n$ represent a sequence of words with a number of n. The derivation process of Equation (4) is as follows.

First introduce the language model. A model that allows word sequences to have probabilities is called a language model (Language Model, LM). Specifically, define a word order $w_1, w_2, w_3, w_4, ..., w_n$, the model that can calculate the probability of the word order is the language model. In addition, using previous words to predict the probability of the next word is also called a language model, for example, predicting the probability of w_{n+1} appearing when the above word order is known.

The language model provides a way to assign a probability to a sentence, and to calculate the probability of the next word in the sequence from the determined part of the sequence. The following example illustrates how to calculate the probability of the next word. For example, the word order "I am learning to walk", then it is necessary to calculate $P(I, am, learning, to, walk)$ or $P(walk|I, am, learning, to)$.

uses the chain rule in probability theory to calculate the joint probability of $P(I, am, learning, to, walk)$, in the case of n word sequences, as shown in equation (5).

$$P(w_1, w_2, w_3, ..., w_n) = \prod_i P(w_i|w_1, w_2, w_3, ..., w_{i-1}) \tag{5}$$

Then the right part of the equal sign in equation (5) can be calculated through statistics in a specified corpus, as shown in equation (6).

$$P(walk|I, am, learning, to) = \frac{\text{count}(I, am, learning, to, walk)}{\text{count}(I, am, learning, to)} \tag{6}$$

If is the word that may appear after the prediction sequence "I am learning to", the probability calculation equation is shown in (7).

$$P(w|I, am, learning, to) = \frac{\text{count}(I, am, learning, to, w)}{\text{count}(I, am, learning, to)} \tag{7}$$

The specific calculation process of equation (7) is to first count the number of occurrences of the sentence "I am learning to" in the corpus, and then count the number of occurrences of the sentence "I am learning to w", where w means "I am learning to" For the words that may appear later, the probability can be calculated at the end. There are two main problems in this calculation process. The first is the sparse problem. In equation (7), if "I am learning to w" does not exist in the corpus at all, then it is obvious that $P(walk|I, am, learning, to)$ is 0; the second One is the storage problem. In equation (7), all the numbers of "I am learning to w" in the corpus need to be stored, which puts pressure on the storage space.

In order to solve the two problems mentioned above, certain assumptions are usually made in language models: the probability of each word in the sequence is only related to a certain number of words in the sequence before the word. We usually call this hypothesis Markov hypothesis. The formal definition of Markov hypothesis is shown in Equation (8).

$$P\left(w, w_2, \ldots, w_n\right) \approx \prod_i P\left(w_i | w_{i-k}, \ldots, w_{i-1}\right) \tag{8}$$

Combined with the above chain rule, equation (9) is obtained.

$$P\left(w_i | w_1, w_2, \ldots, w_{i-1}\right) \approx P\left(w_i | w_{i-k}, \ldots, w_{i-1}\right) = \frac{P\left(w_i, w_{i-k}, \ldots, w_{i-1}\right)}{P\left(w_{i-k}, \ldots, w_{i-1}\right)} \tag{9}$$

Then we need to predict the probability of the next word. We originally needed the information of all the previous words, but now we only need k. The above probability $P(walk|I, am, learning, to)$ can be simplified into $P(walk|am, learning, to)$ or $P(thousand|walk|learning, to)$, respectively, as As shown in equations (10) and (11).

$$P(walk|I, am, learning, to) \approx P(walk|am, learning, to) \tag{10}$$

$$P(walk|I, am, learning, to) \approx P(walk|learning, to) \tag{11}$$

Therefore, we can get $p(w_k \mid w_i, \ldots, w_{k-1})$ in equation (4), and then use the log likelihood function and take the logarithm to get equation (4) The complete expression.

3.2 NATURAL LANGUAGE INFERENCE TARGET TASK

This paper proposes a natural language inference training framework as the target task of continuing pre-training, that is, let the pre-training model use power data for natural language inference tasks in the pre-training stage to achieve the purpose of training the BERT model. The method is described below.

In natural language processing tasks, natural language inference has always been the focus of everyone's research. Natural Language Inference (NLI) uses two levels of learning, semantic understanding and semantic representation, such as generating very effective sentence representation vectors, which can be transferred to other natural language processing tasks. Natural language inference is mainly to judge the semantic connection between two given sentences. These two sentences are called premises and hypotheses respectively. In order to ensure that the network model can learn the semantics between sentences, the natural language inference task can be regarded as a text classification task.

Learning sentence-level general word embeddings by using natural language inference tasks was first proposed by Alexis Conneau et al. Conneau et al. (2018) in 2017. In previous work, unsupervised learning methods were used to learn sentence-level semantic representations. Alexis Conneau et al. used a supervised NLI task to learn semantic representation on annotated data sets, and achieved better results than previous unsupervised methods.

In this paper, inspired by the work of Alexis Conneau and others, the proposed method of training text sentence encoder is introduced as follows, where the encoder is the BERT pre-training model. First, the semantic vector representations of premise and hypothesis are obtained through the text sentence encoder, which are denoted as u and v respectively. On this basis, three methods are used to construct the connection between u and v. The first is splicing. The vectors of u and v are

directly spliced to obtain (u, v). The second One is the dot product, multiplying the values in the corresponding dimensions of u and v to get a new representation $u * v$, the last one is the absolute value after subtraction, that is, pressing u and v Subtract the values on the corresponding elements to get $|u-v|$. Finally, the obtained final representation is sent to a three-category classifier composed of a fully connected layer and a softmax function. The output of the softmax function is the probability distribution of the three-category labels. The three categories are entailment, contradiction and neutral. The training framework for learning text sentence representation through NLI is shown in Figure 5.

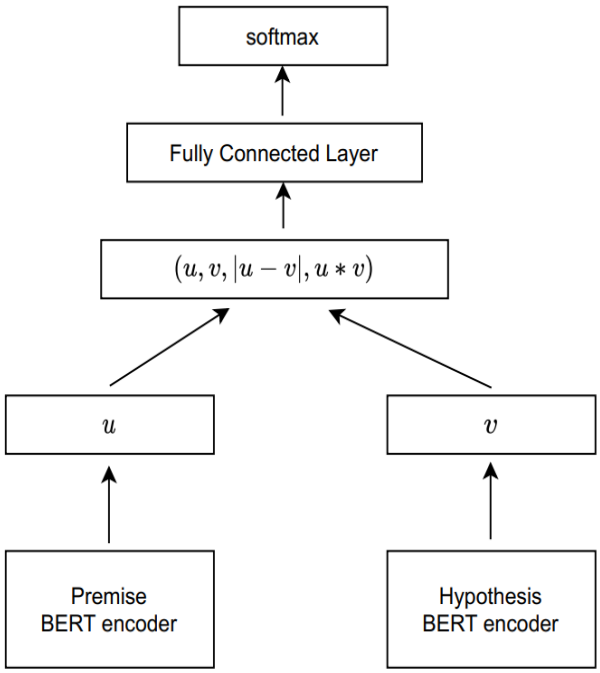

Figure 5: NLI Training Framework

4 KNOWLEDGE DISTILLATION BASED ON MULTIPLE TEACHERS

Considering that human teachers teach students, it is impossible for a teacher to be proficient in everything, so it is often necessary for different teachers to teach different subjects, so that every student can learn the specialties of each teacher. This article is inspired by the teaching of students by multiple human teachers, and proposes a knowledge distillation based on multiple teachers. Therefore, in the knowledge distillation, the knowledge of a single teacher model is often limited. Through multiple teacher models to "teach" students together, the student model can learn the strengths of the teacher model, and the generalization effect of the student model is better.

The "teaching" abilities of the three teacher models BERT, RoBERTa and ERNIE selected in this article are different. First of all, the data that the pre-training model "sees" during pre-training often determines that it will be more familiar with the similar data that it "sees" during pre-training in downstream tasks, thereby producing better results in downstream tasks. . The pre-training data of BERT and RoBERTa are mainly selected from Wikipedia and other neat and formal corpus, while ERNIE mainly selects data generated on platforms such as Baidu Tieba and Weibo, so the format is more random and the content is more diverse. . Secondly, as an improved model of BERT, RoBERTa and ERNIE can learn different information on the data through the difference in network structure. RoBERTa can learn more diverse data dependencies by canceling the NSP pre-training task and using dynamic masking. ERNIE can learn such information as phrases, names and place names through a multi-stage masking method.

Specifically, this article first trains a number of different teacher models, and then uses the teacher model to generate soft labels for the data in the training set. Different from the single teacher model, here we introduce the corresponding weight of each teacher model. The performance of each teacher model on the downstream data is not exactly the same, so let the more capable teacher model have more weight, and the less capable teacher model has a lower weight. The soft labels of multiple teacher models are weighted and averaged according to their weights, and the student model can learn the comprehensive soft labels of multiple teacher models. Finally, like standard knowledge distillation, the student model learns real labels and synthetic soft labels. The knowledge distillation process based on the multi-teacher model is shown in Figure 6.

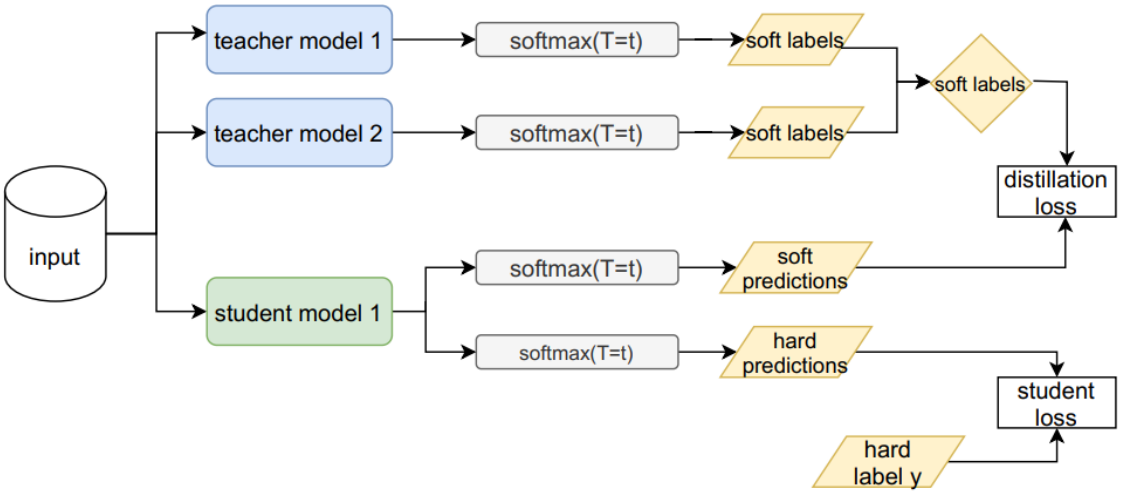

Figure 6: Multi-teacher knowledge distillation flowchart

5 Experimental Results

5.1 Dataset Introduction

5.1.1 Data Sources

This paper collects data under real scenarios in a project cooperating with an electric power business office. The members of the project team used the voice recorder to record the dialogue in the real scene offline. Therefore, this data set has very high authenticity. At the same time, the original recorded voice data contains a lot of noise, such as car whistle outside the business hall, printer noise in the business hall, and the original voice data also contains a lot of meaningless modal particles, excessively long pauses, etc. This requires further processing of the original data.

5.1.2 Speech to Text

Since the original data is stored in the form of speech, the operation of converting speech to text is required to perform the next step of intent recognition. The speech-to-text processing method in this article is implemented using the speech recognition interface of iFlytek.

5.1.3 Manually Organize

At present, most speech-to-text tools are difficult to achieve 100% accuracy. At the same time, because the original speech data contains a lot of meaningless modal particles, long pauses and a lot of external noise, it is necessary to perform the conversion results. Manual verification, remove irrelevant words.

For example, in the result obtained after converting voice to text, ”Well, please, please, please, XXX, what do our company base want, uh, what to do with the electric meter, oh”, this sentence contains

some dialect vocabulary And the modal particles, after sorting out: "Our company base is going to be relocated, what should I do with the electric meter?"

5.1.4 DESENSITIZATION

The data occurs in a real scene, so certain sensitive information is involved, which needs to be desensitized and filtered, such as personal names, electricity account numbers, ID numbers, home addresses, and so on. The purpose of desensitization is to protect user privacy from leaking, and also to make the intention recognition model more universal and universal. The data set used as an intent classification task is mainly for business guidance. Therefore, the content of the data set is actually a description of business needs, such as "The capacity of the electric meter is always at the top, what should I do", "My house is moving, and the electric meter is not needed. So, what should I do", so the amount of sensitive information contained in the actual data is not much.

5.1.5 DATA ANNOTATION

After processing the data in the above multiple steps, a clean and tidy text data is obtained. The members of this project team mark the sorted data set according to the business category specification and knowledge base provided by the electric power business hall company. Each piece of data corresponds to a power business category. There are a total of 35 categories, such as common businesses such as residents applying for electricity meters and time-of-use tariffs, as well as rare businesses such as high and low voltage definitions.

5.1.6 DATA EXPANSION

Preliminary observation of the statistical data set shows that the amount of data for common businesses is relatively large, while the amount of data for rare and unpopular businesses is relatively small. For example, businesses oriented to residents' daily life, such as payment of resident electricity bills and household application for household splitting, account for more businesses, while businesses oriented to business users, such as the expansion of commercial electricity meters and business relocation, account for a relatively low percentage of the data set. Therefore, in order to ensure the balance of the number of label categories and avoid the problem of unbalanced categories, the project team members perform manual transcription and expansion based on the collected real data. The expansion method is to use different narrative methods for the original sentence.

5.2 INTRODUCTION TO EVALUATION INDEX

In the process of training, the model needs to be tested with a validation set to verify the accuracy of the current model's prediction. Therefore, we need a suitable evaluation method to evaluate the classification results, which is an indispensable part of the intent classification process , Choosing appropriate evaluation indicators to train the model in the correct direction is helpful to the improvement and enhancement of the model. The evaluation index used in this article is the F1 value. The evaluation index used in this article is introduced below. Suppose that in the case of two classifications, the categories are 0 and 1, respectively, category 1 represents a positive example, and category 0 represents a negative example.

Accuracy, also known as precision, refers to the ratio between the category of the text in the verification set classified by the algorithm and its correct category. That is, in all cases where the predicted category is 1, how many of the original category is 1, mainly for the result of predicted category 1. The calculation formula is shown in formula 12.

$$P = \frac{TP}{TP + FP} \tag{12}$$

Among them, TP represents the number of samples with label category 1 classified as category 1 by the model.

Recall rate, also known as recall rate, refers to the proportion of all samples with category 1 labels in the verification set that are classified as category 1 by the algorithm, mainly for samples with real labels as category 1. The recall rate reflects the completeness of the classification model. The

larger the value, the higher the completeness of the algorithm. The calculation formula is shown in formula 13.

$$R = \frac{TP}{TP + FN} \tag{13}$$

The precision and recall rates mentioned above mainly refer to the two classification tasks. In this article, we need to perform multi-classification tasks. Then every combination of two categories corresponds to a pair of precision and precision. Full rate. At this time we can use an indicator called Macro F1. The calculation of the macro F1 index is as follows: first calculate the confusion matrix in each two-category task, and calculate the precision rate and recall rate respectively, then calculate the arithmetic mean of the precision rate and the arithmetic mean of the recall rate, and record them respectively For macro-P and macro-R, calculate the harmonic average of macro-P and macro-R to obtain macro F1. The specific calculation formula is shown in formula 14.

$$macro\text{-}F1 = \frac{2 \times macro\text{-}P \times macro\text{-}R}{macro\text{-}P + macro\text{-}R} = \frac{2}{\frac{1}{macro\text{-}P} + \frac{1}{macro\text{-}R}} \tag{14}$$

5.3 Experiments for target task improvement

5.3.1 The impact of continuing pre-training targets on model performance

First download the open source pre-training model from the Internet. The models all contain twelve-layer Transformer blocks, and then continue pre-training on the downstream power intent data set. After the continued pre-training is over, the model obtained after the continued pre-training is fine-tuned on the labeled power intent data to obtain the F1 value effect of the model on the test set. The fine-tuned models are all simple BERT connected fully connected layers and softmax functions. In order to explore whether continued pre-training can improve the effect of the fine-tuned model, a control group that has not undergone continuous pre-training is set for each model. The experimental results are shown in the table 1.

Table 1: F1 value results of different pre-training models on different pre-training targets

model \ training target	MLM+NSP	MLM	LM	NLI
BERT	88.78%	-	-	-
RoBERTa	-	90.29%	-	-
ERNIE	89.26%	-	-	-
BERT+ continues pre-training	89.32%	90.45%	89.92%	92.89%
RoBERTa+ continue pre-training	89.82%	90.88%	90.15%	94.32%
ERNIE+ continue pre-training	89.15%	90.48%	89.83%	93.31%

The table1 shows the comparison of the effects of different pre-training models BERT, RoBERTa and ERNIE on different pre-training targets. Each result in the table is the result of fine-tuning the F1 value of each model on the electric power intention data. Since the BERT without continuous pre-training uses only the pre-training targets of MLM and NSP in the initial pre-training, it has no experimental results in MLM, LM and NLI. The same is true for RoBERTa and ERNIE without further pre-training.

It can be known from the table that the effect of the model after continuing pre-training is better than that of the model without continuing pre-training in most cases. Among the models that have been continuously pre-trained, the RoBERTa model has the best effect among all pre-training models. In RoBERTa, the effect is best when the pre-training target is NLI, with an F1 value of 94.32%. After a simple analysis, it can be seen that RoBERTa uses more data in the initial pre-training process than BERT and ERNIE, and does not use the NSP pre-training target. The performance of the models trained on the NLI task is often very superior. The three pre-trained models perform very well on the NLI target. The reason may be that NLI is an inter-sentence reasoning task that requires a very good understanding of the semantics of the data. Therefore, after using NLI as the pre-training target, when the model is fine-tuned on the next downstream tasks, such as simpler classification tasks, the effect improvement will be more obvious compared to other training targets.

5.3.2 EXPERIMENT OF KNOWLEDGE DISTILLATION BASED ON MULTIPLE TEACHERS

From the experimental results of the effect of the continued pre-training target on the model performance, it can be seen that the model after the continued pre-training performs better than the model without the continued pre-training. From the ablation experiment results of fine-tuning the network structure, it can be seen that the network structure improvement effect is the best when the output of the last three layers of Transformer blocks and the CLS vector are selected for splicing. Therefore, the experiments in this section are carried out on the basis of the above two experiments. First, the three models are continuously pre-trained on the power intent data, and then the fine-tuned model uses the network model improved in this paper. The experimental results are shown in the table 2.

Table 2: Experimental results of F1 value of different models on the improved fine-tuning network model

model \ training target	MLM+NSP	MLM	LM	NLI
BERT+ continues pre-training	89.43%	90.89%	90.13%	93.11%
RoBERTa+ continue pre-training	90.12%	91.10%	90.40%	95.81%
ERNIE+ continue pre-training	89.50%	90.79%	90.10%	93.56%

It can be seen that the improved fine-tuning network model proposed in this paper improves The fine-tuning effect of each model proves that the improvement of the BERT fine-tuning network structure in this article is effective. When the model is selected to continue pre-training, the target is RoBERTa with NLI, and the fine-tuning effect on the improved network model proposed in this paper is the best, and the F1 value of the best effect is 95.81%.

5.4 EXPERIMENT OF KNOWLEDGE DISTILLATION BASED ON MULTIPLE TEACHERS

5.4.1 SINGLE TEACHER MODEL KNOWLEDGE DISTILLATION EXPERIMENT

The single-teacher knowledge distillation experiment distills different single pre-training models BERT, RoBERTa and ERNIE onto CNN and LSTM. The pre-training model selects the best performing model in the previous chapter, that is, the model that has been continuously pre-trained and the target is NLI. The fine-tuning network structure uses the output of the last three layers of Transformer blocks and the improvement of the CLS vector. structure. In order to explore the role of continuous pre-training in knowledge distillation, the model without continuous pre-training was used as a comparison reference group for distillation experiment. The experimental results of knowledge distillation based on a single teacher are shown in the table 3.

Table 3: Single teacher model knowledge distillation F1 value result

teacher model \ student model	CNN	LSTM
BERT	85.78%	85.30%
RoBERTa	87.60%	87.04%
ERNIE	85.66%	85.33%
BERT+ continues pre-training	85.89%	85.39%
RoBERTa+ continue pre-training	88.74%	88.41%
ERNIE+ continue pre-training	85.96%	86.28%

From the table 3, it can be seen that the performance of the student model will be better after the teacher model that has been continuously pre-trained is distilled onto the student model. Among these continuously pre-trained models, RoBERTa's distillation performed best. From the comparison of the effect of the student model CNN and LSTM, it can be seen that the performance of CNN is slightly better than LSTM. After a simple analysis, it can be seen that the model learns a certain amount of knowledge in the downstream data when it continues to pre-train, and the NLI training target is a more difficult target for the model, so the knowledge learned by the teacher model in the continued pre-training Will be better taught to students in the knowledge distillation model.

5.4.2 MULTI-TEACHER MODEL KNOWLEDGE DISTILLATION EXPERIMENT

This paper proposes a knowledge distillation method based on the multi-teacher model. This experiment will explore the effect of the multi-teacher model on the student model on the test dataset. The experimental teacher model is selected from the single-teacher model knowledge distillation experiment, and the selected models are those with better effects that have been continuously pre-trained. The experimental results are shown in table 4.

Table 4: Result of knowledge distillation F1 value of multi-teacher model

multi-teacher model \ student model	CNN	LSTM
BERT + RoBERTa	86.78%	86.37%
BERT + ERNIE	86.23%	86.30%
RoBERTa + ERNIE	88.83%	88.47%

The table 4 is the knowledge distillation training result of the multi-teacher model. This table shows the effect comparison of the joint distillation of multiple teacher models to the student model CNN and LSTM. It can be seen from the table that when the teacher model chooses RoBERTa and ERNIE in the knowledge distillation of the multi-teacher model, the student model performs best, and is better than the effect of the single teacher model RoBERTa and ERNIE after continuing pre-training and then performing distillation training. It is proved that the distillation based on the multi-teacher model can make the student model more effective. When the multi-teacher model selects BERT and RoBERTa separately, it may lead to a situation where the effect is worse than that of a single teacher model. This paper analyzes that because BERT and RoBERTa use similar training data in the initial large-scale pre-training, and because the effect of BERT and RoBERTa on the single-teacher model is large, the two models are teaching the student model at the same time. The time cannot play a complementary role, so the effect is worse. Therefore, in the knowledge distillation based on multiple teachers, the selection of teacher model is very important. The greater the difference between multiple models, the better the distilling effect will generally be.

6 CONCLUSION

With the continuous improvement of deep learning technology in artificial intelligence, there are more and more smart products around us that can interact with us. These dialogue robots can not only communicate emotionally with humans, but also help humans increase productivity. As an important part of the dialogue system, intention recognition requires a better and faster understanding of the user's intention expression, and then the next step is to feedback the user the correct information.

This paper is based on the pre-training model for intent recognition, using the electric power intent recognition data obtained in the real scene, and by continuing to pre-train on the data, the effect of intent recognition is further improved. At the same time, in view of the problem of too large pre-training model volume and too long loading time, the knowledge distillation method in model compression is used to reduce the amount of model parameters and loading time, making it possible to realize a model with high accuracy and small size. The main work of this paper is summarized as follows:

(1) Constructed a power-based intent recognition data set. Through cooperation with the electric power business office, 9577 intent identification data sets containing 35 categories were collected. The data set is mainly collected by the project team members in the real dialogue scene of the electric power business hall, filling the gap of the intention recognition data set in the electric power industry.

(2) In the intent recognition module, in view of the problems that the NSP in the pre-training target of BERT is not necessarily effective, the pre-training data and the downstream task data are inconsistent, etc., the improved RoBERTa and ERNIE based on BERT are proposed to continue pre-training on the power intent data. Training and proposed two new pre-training goals. In order to further improve the recognition effect of the model, this paper improves the network structure of the

fine-tuning stage based on BERT. Through experiments with different pre-training targets, including MLM+NSP, MLM, LM, NLI, the experimental results show that the use of power intent data and NLI pre-training targets in the continued pre-training enables fine-tuning of the improved network structure on downstream classification tasks The F1 value reached the best 95.81%.

(3) In the model compression module, due to the large size of the pre-training model, the large amount of parameters, and the long loading time, better hardware deployment conditions in the actual scenario are required. In the real application scenario, whether it is a separate The intention recognition system is still a module in the human-machine dialogue, and it is difficult to guarantee a good enough hardware environment. Based on the previous pre-training work, this article distills and compresses the two improved BERTs and compresses them to CNN and LSTM respectively, reducing the amount of model parameters from more than 100 million to 500,000. The student model after a single teacher model is compressed The F1 value is best to reach 88.74%. This paper also proposes a knowledge distillation based on the multi-teacher model to improve the performance of the student model. The F1 value of the student model compressed based on the multi-teacher model can best achieve 88.83%.

(4) Designed and implemented the entire intention recognition system. The system realizes the automatic recognition of the user's intention, and displays the recognition result to the user through a graphical interface.

The intention recognition system implemented in this paper encountered some unpredictable problems in actual application, and gradually discovered some areas that can be further improved. This system has not considered the situation when the user has multiple intentions in one sentence. Although in most scenarios, the user expresses a single intention, in a few practical scenarios, the user sometimes expresses multiple needs in one sentence, and this system only realizes the recognition of a single intention. In the follow-up work, we can consider the use of multi-label classification methods for multi-intent recognition. In the mode compression module, although the student model used in this article is small in size and fast in reasoning, it also affects the accuracy of the original teacher model. There is still room for improvement in the accuracy of the student model. In the follow-up work, we can consider digging out more difficult to classify and accurate samples for analysis, and further improve the accuracy of student model recognition. At the same time, we can also try to design a better loss function to carry out the model distillation experiment.

REFERENCES

Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, and Gerald Penn. Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. In *2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)*, pp. 4277–4280. IEEE, 2012.

J.L. Austin, J.L. Austin, J.O. Urmson, J.O. Urmson, and M. Sbisà. *How to Do Things with Words: Second Edition*. Harvard paperback. Harvard University Press, 1975. ISBN 9780674411524. URL https://books.google.com/books?id=V43VS07TGEMC.

A. Celikyilmaz, D. Hakkani-Tur, G. Tur, A. Fidler, and D. Hillard. Exploiting distance based similarity in topic models for user intent detection. In *2011 IEEE Workshop on Automatic Speech Recognition Understanding*, pp. 425–430, 2011. doi: 10.1109/ASRU.2011.6163969.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. A survey on dialogue systems: Recent advances and new frontiers. *ACM SIGKDD Explorations Newsletter*, 19, 11 2017. doi: 10.1145/3166054.3166058.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

National Bureau of Statistics. http://data.stats.gov.cn/.

Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. 2015.

Arik Poznanski and Lior Wolf. Cnn-n-gram for handwriting word recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2305–2314, 2016.

Gokhan Tur and Renato De Mori. Spoken language understanding: Systems for extracting semantic information from speech. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, 03 2011. doi: 10.1002/9781119992691.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

J. D. Williams. Spoken dialogue systems: Challenges, and opportunities for research. In *2009 IEEE Workshop on Automatic Speech Recognition Understanding*, pp. 25–25, 2009. doi: 10.1109/ASRU.2009.5372951.