Preprints.org

Article

Research on Machine Reading Comprehension Integrating Longdistance Semantic Relations and Linguistic Features

Ning Ma [*] and Wu Di

Article

Research on Machine Reading Comprehension Integrating Longdistance Semantic Relations and Linguistic Features

MA Ning[1,2],* and Wu Di [1,2]

[1] Key Laboratory of China's Ethnic Languages and Information Technology of Ministry of Education, Northwest Minzu University, Lanzhou 730030, China
[2] Key Laboratory of China's Ethnic Languages and Intelligent Processing of Gansu Province, Northwest Minzu University, Lanzhou 730030, China
* Correspondence: maning8162@163.com

Abstract: Machine reading comprehension is a crucial area of research in natural language processing, aiming to enable machines to read and understand text as humans do, and to answer questions related to the text content. The pre-trained language model, represented by BERT, has achieved superior results compared to traditional models in many NLP tasks, leading to the rise of the pre-trained paradigm in the field of natural language processing. This paper addresses the issue that the pre-trained language model lacks the ability to extract long-distance semantic relations and make efficient use of linguistic features. Firstly, the recent developments of the pre-trained language model are described. Then, two feature maps are used to intuitively express the structured long-distance semantic correlation features, and the traditional sequence structure features are integrated. The influence of different feature map construction methods on machine reading comprehension is compared. Finally, the application of the pre-trained language model in machine reading comprehension is summarized and prospected. By fusing graph structure features, the model can learn more abundant language knowledge, thus further improving its reasoning ability.

Index Terms: machine reading comprehension; long-distance semantic relations; linguistic features; pre-trained language models

I. Introduction

With the rapid development of information technology, Machine Reading Comprehension (MRC), as one of the core tasks in the field of natural language processing, has received extensive attention and research. Machine reading comprehension aims to understand a text in the form of a question and answer, with the input being a question in natural language form and an article containing evidence to support the answer to the question, and the output being the answer to the question[1]. However, in practical applications, machine reading comprehension tasks still face many challenges, one of which is how to effectively deal with long-distance semantic relations in text and linguistic features that are often ignored during the process of model understanding. Traditional machine reading comprehension methods tend to focus on processing text semantic information on a word-by-word basis, with models pre-trained on large-scale unstructured text to obtain a generic language representation, and then fine-tuned in specific downstream tasks to adapt to different tasks[2]. For example, after the Transformer model is proposed, it can effectively solve the problem of long-distance dependence in text through self-attention mechanism[3], providing a new solution for machine reading comprehension. Although some researches have made some progress, how to integrate long-distance semantic relations more effectively to improve the accuracy and depth of machine reading comprehension is still an important research direction in this field. At the same time, for machine reading comprehension, linguistic features are also crucial to text comprehension,

providing information about text structure, semantic roles, etc., which helps models grasp the meaning of text more accurately. Although pre-trained language models have achieved promising results in many NLP tasks, they neglect to incorporate structured knowledge into language understanding tasks and are often limited by tacit knowledge representation[4]. Therefore, the integration of long-distance semantic relations and linguistic features becomes the key to improve the performance of machine reading comprehension.

The purpose of this paper is to explore a machine reading comprehension method that integrates long-distance semantic relations and linguistic features. Firstly, this paper analyzes the shortcomings of existing machine reading comprehension methods in dealing with long-distance semantic relations and linguistic features, and puts forward corresponding improvement ideas. Then, we propose a machine reading comprehension model integrating linguistic features, which can effectively capture long-distance semantic relationships in text through graph convolutional neural networks. At the same time, combined with linguistic features, the model's ability to understand the text is further enhanced. This study not only helps to promote the development of machine reading comprehension, but also provides a valuable reference for other NLP tasks. Through in-depth analysis and experimental verification, it is expected that the proposed method can achieve remarkable results in improving the performance of machine reading comprehension, and provide strong support for practical applications.

II. Related Work

A. Pre-Trained Language Models

Pre-trained language models obtain generic language representations by pre-training on large-scale open domain corpora, and then fine-tune them in specific downstream tasks to learn domain-specific knowledge[5]. The introduction of Transformer[6] marks the further improvement of model understanding and generalization ability, more accurate semantic feature extraction, and the ability of machines to learn the relationship between problems and answers through multi-layer neural networks. Traditional four-layer network models such as QANet[7] and BiDAF[8] have gradually faded from researchers' attention. In 2018, the Google research team launched the groundbreaking pre-trained language model BERT[9], an innovative initiative that inspired a lot of natural language processing research based on pre-trained language models, and also triggered the rise of pre-trained paradigms in the field of natural language processing[10]. Joint Laboratory of HIT and iFLYTEK Research has released a variety of Chinese pre-training model resources and related supporting tools. "Pre-training+fine-tuning" technology can solve different languages and different NLP tasks with one set of technologies, effectively improving the development efficiency. These include Chinese MacBERT[11], Chinese ELECTRA[12], Chinese XLNet[13], etc. In the process of model input, three parts of information need to be fused, the first is the Token Embedding, each lexicon has a corresponding index in the vocabulary, BERT maps these indexes through a learnable word embedding matrix into a continuous vector representation, these vectors capture the semantic information of the lexicon. To distinguish between different sections of input text, such as context and questions in question answering tasks, BERT introduces Segment Embedding. These embeddings are learnable vectors that are added to the representation of each tag in each paragraph. Since the Transformer architecture itself does not have the ability to capture sequence order, BERT adds Positional Embedding to the input representation to provide information about where the tags are in the sequence. Finally, all embeddings are added up as the final input, and the input form is shown in the Figure 1
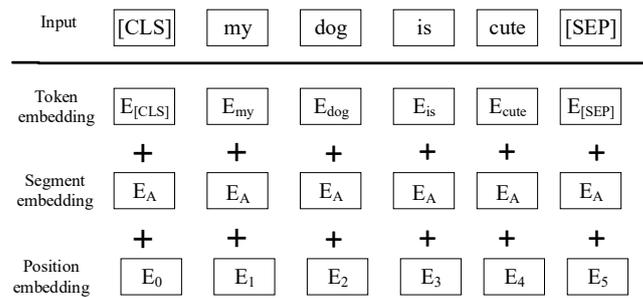
3

Figure 1. Model input form.

The pre-trained language model utilizes the transformer architecture to handle long-distance dependencies through a self-attention mechanism, which is critical for understanding complex logic and reasoning in text. At the same time, the Mask Language Model (MLM) and the Next Sentence Prediction (NSP) can further optimize the semantic features of the text, and improve the generalization ability and adaptability of the model. The structure of the pre-trained language model represented by BERT is shown in the figure below.
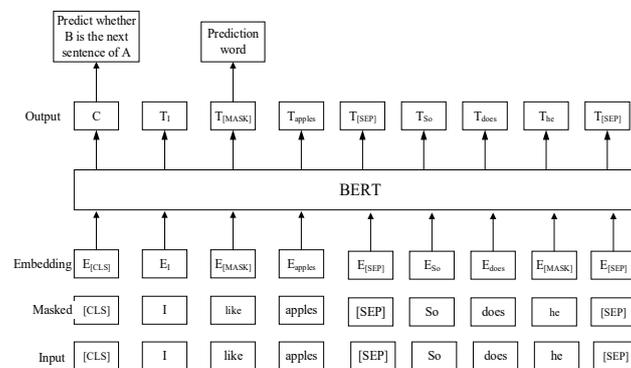

Figure 2. Model structure.

B. Graph Convolutional Network

Traditional four-layer network model and pre-trained language model can only process data in Euclidian space. In the field of natural language processing, text structure, syntax and even the sentence itself exist in the form of graph data[14]. To seamlessly integrate structured data into a language model, you need to provide a comprehensive view of each node in the graph, including its neighbors, relationships to itself, and a clear description of the node[15]. Graph neural networks provide the ability to model non-European spatial data, which further expands the application range of deep neural networks[16]. Graph convolutional neural network[17] generates a vector representation by aggregating information of nodes and their neighbors, so that it can process graph data with complex structural relationships. Its core lies in graph convolution operation, which can capture local and global features in graph structure and optimize model performance through forward and backpropagation processes.

Graph convolutional neural networks have shown strong capabilities in many fields, such as social network analysis, recommendation system, traffic prediction and so on. The graph structure is represented by G=(V, E), where V represents the set of nodes and E represents the set of edges. n represents the number of nodes and m represents the number of edges. In GCN, only undirected graphs are usually considered. In this paper, the relationship between the named entities of the text is constructed as a graph, and its vector representation is obtained based on it. The convolutional neural network structure of the graph is shown in the figure.
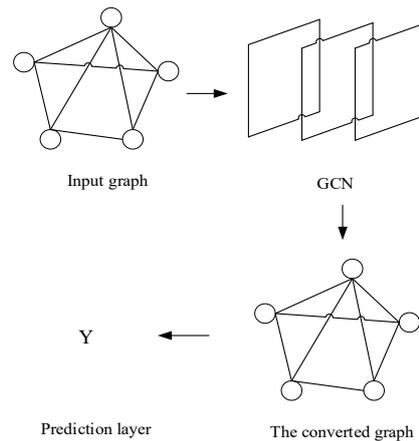
Figure 3. Network structure.

III. 3 Semantic Fusion

A. Named Entity Extraction

Named entities, such as personal names, place names, organization names, dates, times, etc., are specific and identifiable important information units in a text. In machine reading comprehension, the system needs to accurately identify these entities so that it can accurately locate relevant information when answering questions or summarizing text. As can be seen from the following example, named entities occupy a core position in the semantics of paragraphs and there is a lot of repetition, which provides the possibility to collect entity information to enrich the semantic representation of text.

Cleaner is a free system optimization and privacy protection tool. CCleaner is used to clean Windows. Unpack junk files that are no longer used to free up more hard disk space. Another function is to clear the user's Internet history. CCleanerBT is small in size and runs extremely fast, which can clean up garbage from folders, history recycle bins, etc. The registry is scanned and cleaned for garbage items. Includes software uninstall function. It also supports free use of E and Firefox. It does not contain any spyware and junk programs, and supports 26 language interfaces including simplified Chinese.

Figure 4. Example of a text segment.

The resulting named entity recognition result contains the entity name, the start index in the text, the end index, and the entity type. The purpose of the start and end indexes is to pinpoint the position of the entity within the text or question, making it easier to combine with word granularity information in the original text. In this paper, Stanford open source toolkit Stanza[18] is used to extract named entity information. Stanza is a powerful multi-language named entity extraction tool, which provides a convenient natural language processing interface for Python users and supports text analysis in more than 60 languages. Including part-of-speech tagging, named entity recognition, dependency parsing and other NLP functions.

B. Construction of Entity Co-Occurrence Graph Based on Sentence Co-Occurrence Relation

As for the co-occurrence relationship of sentences, it can be understood that a paragraph is divided according to the unit of sentences, and the number of simultaneous occurrences of two named entities in the same sentence is counted respectively, which is used to measure the correlation strength between two entities. If two named entities appear frequently in the same sentence, the number of co-occurrences between them will increase, indicating that they are more relevant. If two entities appear n times in the same sentence, the edges in their co-occurrence graph are weighted to

n. The paragraphs understood by reading and questions to be answered are joined together as the model input, and the entity co-occurrence diagram based on the sentence co-occurrence relationship is obtained, as shown in the figure below.
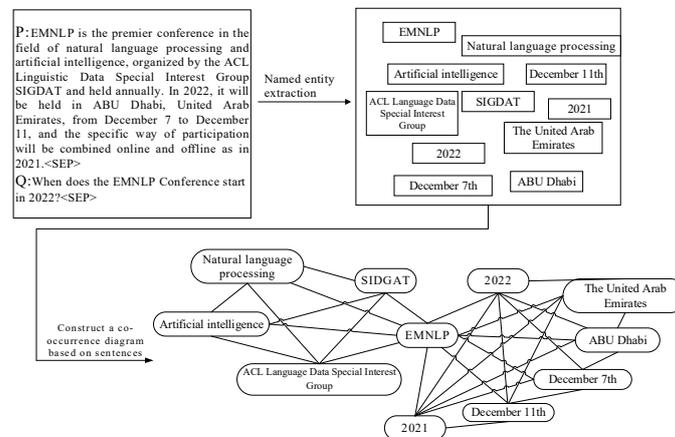

Figure 5. Graph based on the intersentence co-occurrence relationship.

C. Construction of Entity Co-Occurrence Graph Based on Sliding Window

During the construction of entity co-occurrence graph based on sliding window, the weight of the connected edge between two nodes is no longer just the number of times the two entities co-occur in the sentence, but the point mutual information value (PMI)[19] is adopted as the measurement standard.

The PMI value is standardized to the interval of [-1, 1], where the PMI limit value -1 indicates that the two entities have never appeared at the same time within the text range under investigation; when the PMI value is 0, it indicates that the two entities are independent in appearance and have no correlation; when the PMI value is 1, it means that the two entities always occur at the same time completely. The introduction of standardized PMI enables more precise quantification of the strength of co-occurrence relationships between entities. The PMI value is calculated by the co-occurrence relationship of two words in the sliding window. Set a sliding window with a length of 10, and add 1 to the weight of two entities when they appear in the same document. The calculation formula for the PMI value is as follows.

$$PMI\ (i, j) = \log \frac{p(i, j)}{p(i)\,p(j)} \tag{1}$$

$$P(i, j) = \frac{\#W(i, j)}{\#W} \tag{2}$$

$$p(i) = \frac{\#W(i)}{\#W} \tag{3}$$

Where, #W (i) is the number of sliding Windows in the corpus containing word i, #W (i, j) is the number of sliding Windows containing both words i and j, and #W is the total number of sliding Windows in the corpus. This method can adjust the window size and move step according to the text length, and dynamically generate the feature map, but it ignores the sentence as the original text unit, which may lose part of the original semantic information. The entity co-occurrence diagram based on sliding window is shown in the figure below.
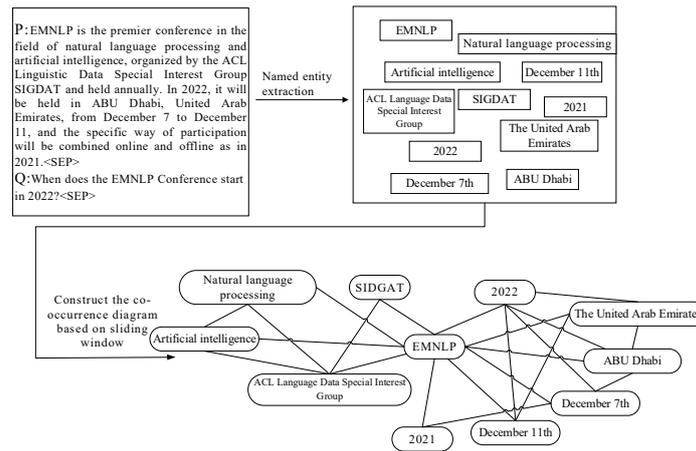
Figure 6. Graph based on slding window co-occurrence relationship.

D. Named Entity Feature Generation

When the entity cooccurrence graph is input into the graph convolutional neural network, information consolidation and feature update are performed for each node in the hidden layer. This process involves two aspects. The first is neighbor information aggregation. For each node in the graph, the model will consider the characteristics or information of its neighbor nodes and aggregate the information. The second is node feature update, which usually combines the original node feature with the aggregated neighbor information through an update function to generate a new, richer node representation. This process involves two aspects. The first is neighbor information aggregation. For each node in the graph, the model will consider the characteristics or information of its neighbor nodes and aggregate the information. The second is node feature update, which usually combines the original node feature with the aggregated neighbor information through an update function to generate a new, richer node representation. As shown in the figure below, each node in the figure represents the named entity of the text segment, and its adjacency matrix is A. Assuming that a, b, c, d, and e are named entities, the adjacency matrix and degree matrix examples are shown in the figure below.
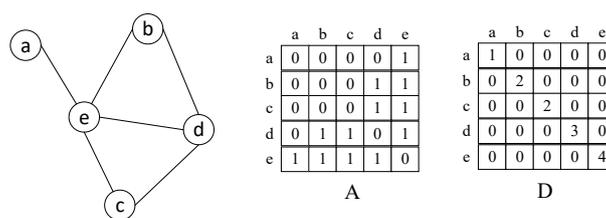

Figure 7. Examples of adjacency matrix and degree matrix.

The matrix after aggregating the node features of the graph is shown in the figure below.
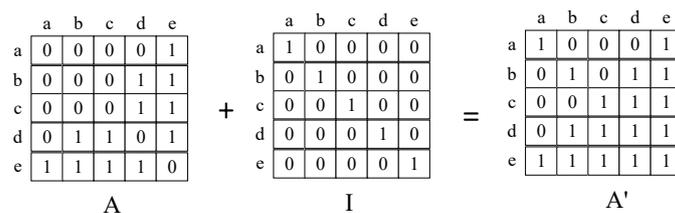

Figure 8. Example of information aggregation matrix.

Through the stacking of multi-layer GCN, higher-level structure information of the graph can be gradually captured, thus generating a richer node representation. The specific propagation mode is shown as follows.

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{4}$$

GCN is A multi-layer graph convolutional neural network, each convolutional layer only processes first-order neighborhood information, and multi-order neighborhood information transfer can be realized by superimposing several convolutional layers. Each layer of GCN is multiplied by adjacency matrix A and feature matrix H(l) to obtain a summary of each vertex neighbor feature, and then multiplied by a parameter matrix W(l). The matrix H(l+1) of the features of the aggregated adjacent vertices is obtained by a nonlinear transformation of the activation function. Assuming a two-layer GCN is constructed and the activation functions are ReLU and Softmax respectively, the overall forward propagation formula is as follows:

$$Z = f(X, A) = soft\max(A^{'}\operatorname{Re}LU(A^{'}XW^{(0)})W^{(1)}) \tag{5}$$

By stacking multiple GCN layers, features can be propagated over a wider area of the graph. Each layer receives the output of the previous layer as input and repeats the process of feature propagation, transformation, and activation. After multi-layer feature propagation, each node will obtain an updated feature representation, which encodes the structure and feature information of the node in the graph and can be used to obtain the final named entity feature representation. The propagation process of node features is shown in the figure below.
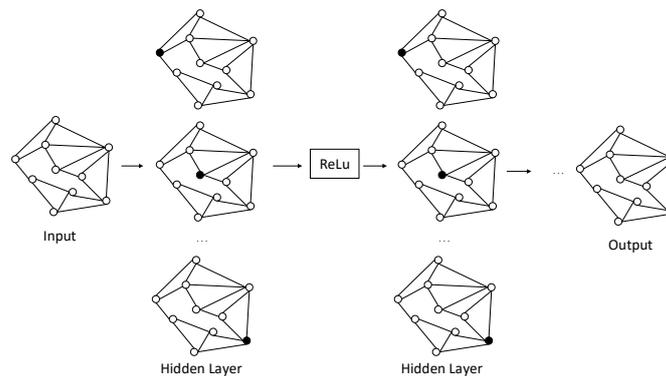

Figure 9. Feature propagation process.

E. Feature Fusion

Structured entity knowledge can be effectively integrated into unstructured original text information by combining context embedding vector based on pre-trained language model and entity node embedding vector based on graph structure learning. This process helps the model to further capture deep linguistic features, thus improving the model's ability in semantic extraction and entity information perception. This fusion strategy enables the model to better understand and utilize the complex semantic relationships in the text, thus improving the overall performance. The overall model structure is shown in the figure.
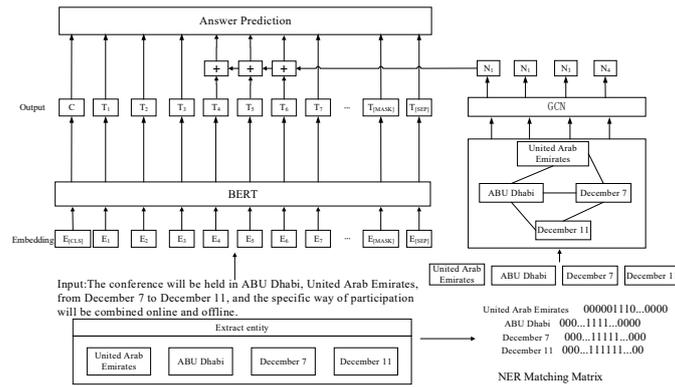
Figure 10. Model architecture.

Suppose that in a machine reading comprehension task, the input text segment is represented by $c = (c_1, c_2, c_3, ... c_{k_c})$, where kc represents the number of characters in the text segment, once the named entities are extracted, an entity matching matrix M is used to mark the position of each entity in the text segment, so M is a kc×kn matrix, where each element is calculated as follows.

$$m_{i,j} = \begin{cases} 1, & c_i \in n_j \\ 0, & c_i \notin n_j \end{cases} \tag{6}$$

Where kn represents the number of named entities extracted in paragraph c, and nj represents the NTH named entity extracted. An example M for input text is shown at the bottom of Figure 11. After the sequence-based embedding obtained by the pre-trained model and the entity embedding obtained by the graph structure, the two need to be fused to obtain a unified embedding representation combining the two features, and then the answer prediction is carried out. Here, a_i is used to represent the word in the text paragraph, and e_j is used to represent the named entity in the paragraph. When feature fusion is carried out, the word embeddings, that is, the embeddings obtained by the pre-trained model, are represented by $E(a_i)$; the named entity embeddings, that is, the embeddings obtained by the graph structure, are represented by $E(e_j)$, and the fusion mode is as follows.

$$E(a_i) = \begin{cases} E(a_i), & a_i \notin e_i \\ E(a_i) + E(e_j), & a_i \in e_i \end{cases} \tag{7}$$

In the fusion process, $v_i^{(l)}$ is used to represent the i th character of layer l, and $\mu_{i,k}^{(l)}$ is used to represent the k th named entity related to $v_i^{(l)}$. The specific calculation formula for each character is as follows.

$$v_i^{(l)*} = v_i^{(l)} + \sum_k \mu_{i,k}^{(l)} \tag{8}$$

Extending to the whole layer, the calculation formula is as follows.

$$V^{(l)*} = V^{(l)} + M \times U^{(l)} \tag{9}$$

Where V(l) is the embedding matrix of all characters, and its association with U (l) can be done directly through M.

IV. Experiment

A. Dataset

Dureader was adopted as the basic data set of the experiment. Dureader is a large-scale Chinese machine reading comprehension data set, which contains a large number of question and answer pairs under real scenarios, and aims to evaluate the model's ability to understand natural language text and answer questions. It includes three types of data, namely entity type, description type and non-correct type. Each type is also divided into fact and opinion, and the data contains 200k questions, 1000k original text and 420k answers. In addition, a part of question and answer data is extracted from the Dureader data set as a supplement, and the data format is normalized by means of data cleaning and regularization, as shown in the figure below.

```
{
    'documents': [
        {
            'is_selected': True or Flase,
            'title': ' String',
            'paragraphs':[ ' String', ' String', ' String'      ],
            'segmented_title': [fenci],
            'most_related_para': 0 or 1 or 2···..
            'segmented_paragraphs':[ [fenci], [fenci], [fenci] ]
        },
        ..........
    ]
}

    'answers': ['String', 'String', 'String'],
    'question':   ‘String’ ,
    'question_type': DESCRIPTION or ENTITY or YESNO,
    'fact_or_opinion': FACT or OPINION,
    'question_id': int,
```

Figure 11. Data format.

B. Experimental Setup

In this experiment, GPU was used for training, Python was the development language, Pytorch was the deep learning framework, and model training was carried out on RTX 3090. Experimental parameters are shown in Table 1.

Table 1. Experimental parameter.

parameter	value
Epoch	10
Batch Size	16
Learning rate	5e-5
Dropout	0.1
Longest input sequence	512

The base BERT Fusion model is a Bert-base model provided by Google. The model consists of 12 layers, 768 hidden layers and 16 self-attention mechanism heads. In the training process, the text length and problem sequence length are defined as 512 and 60 respectively, and the batch size is 8. In order to make efficient use of cache space, gradient_accumulation_steps is set to 8.

The optimization function uses Adam, where the learning rate is set to 5e-5 and the warmup is 0.1.

C. Evaluation Index

Rouge-L is a recall ration-based assessment that focuses on the longest common subsequence(LCS) between the original answer and the predicted answer. By calculating the length of the LCS, Rouge-L is able to measure the similarity between two answers, a method that is particularly effective for evaluating machine reading comprehension tasks because it is able to capture key units of information in the answer and consider the relative position of these units in the

original answer and the predicted answer. Therefore, a higher Rouge-L score usually means that the predicted answer overlaps more with the original answer, and the quality of the generation is better.

Where LCS (X, Y) is the length of the longest common subsequence of X and Y, m and n represent the original answer length and predicted answer length respectively, Rlcs and Plcs represent the recall rate and accuracy rate respectively, and finally Flcs is the ROUGE-L value, the calculation method is shown in the following formula.

$$R_{lcs} = \frac{LCS(X,Y)}{m} \tag{10}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \tag{11}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{12}$$

BLEU-4 is an evaluation method based on accuracy rate, which focuses on how often the 4-tuple in the predicted answer appears in the original answer. The higher the BLEU-4 score, the more tuples of the original answer contained in the predicted answer, reflecting the greater similarity between the predicted answer and the original answer. BLEU-4 is widely used in machine translation tasks because of its ability to capture word choice and phrase structure in translated results. The value range of BLEU-4 is between 0 and 1, the closer the model score is to 1, the higher the quality is, the number of consecutive words is set to 4, the machine reading comprehension task question and answer text is longer, the higher-order BLEU value can be used to measure the fluency of the answer at the same time, lr represents the predicted answer length, lc represents the original answer length, the calculation method is shown in the formula.

$$BLUE = BP \times \exp(\sum_{n=1}^{N} w_n \times \log P_n) \tag{13}$$

$$BP = \{ \begin{array}{ll} 1 & lc > lr \\ \exp(1-lr/lc) & lc \leq lr \end{array} \tag{14}$$

D. Experimental Results and Analysis

In view of the designed model structure, this paper conducts experiments on three basic pre-trained language models, namely BERT, PERT based on the prediction task of similar words and MacBERT based on the prediction task of similar words, by comparing and analyzing the improvement of the model effect after integrating long-distance semantic and linguistic information.

As shown in Table 2, the results of the three pre-trained language models have been improved to a certain extent after the integration of the features based on the co-presence graphs in sentences. BERT model BLEU-4 and ROUGE-L are 45.52 and 46.87, respectively, which are 1.52% and 1.06% higher than the original results. The two indexes of PERT model were 45.28 and 48.55, respectively, which increased by 0.32% and 0.78% on the basis of the original results, among which MacBERT had the best effect, BLEU-4 and ROUGE-L reached 51.32 and 52.98, respectively, increasing by 1.54% and 2.62%.

Table 2. The experimental results of co-occurrence graphs within sentences.

Model structure	Raw result		Result	
	BLEU-4	ROUGE-L	BLEU-4	ROUGE-L
BERT	44.0	45.81	45.52	46.87
PERT	44.96	47.77	45.28	48.55
MacBERT	49.78	50.36	**51.32**	**52.98**

As shown in Table 3, the results of the three pre-trained language models were also improved to a certain extent after integrating the features of the sliding window-based co-appearance map. Through experiments, BERT model BLEU-4 and ROUGE-L were 45.15 and 47.11, respectively, which increased by 1.15% and 1.30% on the basis of the original results. BLEU-4 and ROUGE-L in PERT model were 45.44 and 47.29, respectively, which increased by 0.48% and 0.02% on the basis of the original results, while BLEU-4 and ROUGE-L in MacBERT model were 50.72 and 51.98, respectively. The results were improved by 2.17% and 3.26% respectively.

Table 3. The experimental results based on sliding window co-occurrence map.

Model structure	Raw result		Result	
	BLEU-4	ROUGE-L	BLEU-4	ROUGE-L
BERT	44.0	45.81	45.15	47.11
PERT	44.96	47.77	45.44	47.79
MacBERT	49.78	50.36	**51.95**	**53.62**

It can be seen that the model effect has been improved to some extent by integrating the entity features constructed by long-distance semantic relations and different feature maps. Comparing the two different co-occurrence map construction methods, it can be seen that the overall effect of co-occurrence map based on sliding window is better. First, the construction method based on sentence co-occurrence relationship can better reflect the semantics of the original text, but the feature maps may be too sparse. It can't even form a complete connected graph. The construction method based on sliding window can dynamically generate the feature map by adjusting the window size and moving step size according to the text length, which can control the sparsity of the map to a certain extent. On the other hand, the weights of the edges between nodes are defined by the mutual information values of points during the construction of the sliding window-based cooccurrence graph, which can more accurately represent the relationship between entities.

V. Summary and Prospect

This paper focuses on the research of machine reading comprehension integrating long-distance semantic relations and linguistic features. By introducing the graph structure entity association relationship, the sequence structure features and graph structure features are fused to significantly improve the model performance. The experimental results show that this fusion strategy can not only help the model to capture the deep semantic information in the text, but also effectively process the complex language structure, thus improving the accuracy and robustness of the model.

In view of the development status of machine reading comprehension, first of all, we will continue to optimize the feature fusion strategy, explore more efficient feature fusion methods, reduce the computational complexity and improve the model performance. Secondly, more advanced model structures and technologies, such as knowledge graphs, multimodal information and large language models, will be explored to further expand the application range of machine reading comprehension. Finally, we will focus on the interpretability and reliability of the model, and strengthen the research on the internal mechanism of the model in order to better understand its working principle and performance.

Author Contributions: N.M.: Conceptualization, Methodology, Software; D.W.: Data curation, Writing-Original draft preparation. N.M.: Visualization, Investigation; D.W.: Supervision,Software and Validation; N.M.: Writing- Reviewing and Editing. All authors read and approved the final manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

References

1. YAN Weihong, Li Shaobo, Shan Lili, Sun Chengjie, Liu Bingquan. An extractive machine reading comprehension model with explicit lexical and syntactic features [J]. Applications of Computer Systems,2022,31(09):352-359.

2. Bian R T. Application of pre-training model integrating knowledge in reading comprehension [D]. Harbin Institute of Technology, 2021.

3. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Proceedings of   Advances in Neural Information Processing Systems，2017: 5998-6008.

4. [4]Lu, Yinquan et al. KELM: Knowledge Enhanced Pre-Trained Language Representations with Message Passing on Hierarchical Relational Graphs[J]. arXiv preprint arXiv: 2109.04223, 2021.

5. Liu, Weijie , et al. "K-BERT: Enabling Language Representation with Knowledge Graph." Proceedings of the AAAI Conference on Artificial Intelligence 34.3(2020):2901-2908.

6. Liu, Shanshan et al. Neural Machine Reading Comprehension: Methods and Trends[J]. arXiv preprint arXiv: 1907. 01118, 2019.

7. Seo M, Kembhavi A, Farhadi A, et al. Bidirectional attention flow for machine comprehension[J]. Computer Science, arXiv: 1611.01603, 2016.

8. Yu A W, Dohan D, Luong M, et al. QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension[C]//Proceedings of the International Conference on Learning Representations, 2018 : 13-17.

9. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.

10. Zhang Qi, GUI Tao, Zheng Rui, Huang Xuanjing. Large-scale Language Model: From Theory to Practice [M]. Publishing House of Electronics Industry, 2024.

11. Cui, Yiming et al. Revisiting Pre-Trained Models for Chinese Natural Language Processing[J]. arXiv preprint arXiv: 2004.13922, 2020.

12. Clark, Kevin, et al. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators[J]. arXiv preprint arXiv: 2003.10555, 2020.

13. Yang, Zhilin, et al. XLNet: Generalized Autoregressive Pretraining for Language Understanding[C]// Advances in Neural Information Processing Systems, 2019.

14. Chen Yulong, Fu Qiankun, Zhang Yue. Application of graph neural networks in natural language processing [J]. Journal of Chinese Information Processing, 2021,35(3):23.

15. Yichong Xu, Chenguang Zhu, Ruochen Xu, Yang Liu, Michael Zeng, and Xuedong Huang. 2021. Fusing context into knowledge graph for commonsense question answering. In Findings of ACL-IJCNLP 2021, pages 1201-1207.

16. Zhang S, Tong H, Xu J, et al. Graph convolutional networks: A comprehensive review[J]. Computational Social Networks, 2019, 6(1): 11.

17. Kipf T N , Welling M . Semi-supervised Classification with Graph Convolutional Networks[J]. arXiv preprint arXiv:1609.02907, 2016.

18. Qi P, Zhang Y, Zhang Y, et al.Stanza: A Python Natural Language Processing Toolkit for Many Human Languages[J]. 2020. acl-demos.14.

19. Yao L, MaoC, Luo Y. Graph convolutional networks for text classification[C]//Proceedings of the AAAI Conference on Artificial Intelligence，2019, (33): .7370-7377.

13

Short Biography of Authors

FIRST A. Ma Ning PhD, Professor, Doctoral Supervisor of Language Intelligence and Cultural Computing at Northwest University for Nationalities, Master's Supervisor of Computer Science and Technology, Computer Technology, and Artificial Intelligence, Distinguished Professor of Feitian Scholars in Gansu Province, Director of Key Laboratory of Language and Cultural Computing of the Ministry of Education, Leader of Master's Supervisor Group of Artificial Intelligence, Director of the Chinese Information Society, Deputy Secretary General of the Ethnic Language and Text Information Professional Committee, Member and Executive Secretary of the Computational Linguistics Special Committee, and Member of the 11th Academic Committee, Teaching Guidance Special Committee, and Academic Ethics Special Committee of Northwest University for Nationalities. Engaged in teaching and research in the field of machine learning and natural language processing. Published 1 monograph, published over 60 academic papers, participated in and completed 2 national science and technology support program projects and 1 major science and technology special project in Gansu Province as a core scientific research backbone, led and completed 1 National Natural Science Foundation project and 5 provincial and ministerial level projects, applied for 3 patents, 6 software copyrights, won 1 second prize of Qian Weichang's Chinese Information Processing Science and Technology Award, 1 third prize of Gansu Province's Higher Education Science and Technology Progress Award, and 1 first prize of Gansu Province's Department of Education's Science and Technology Progress Award. Engaged in teaching and research in the field of natural language processing in artificial intelligence, with a focus on text generation, knowledge Q&A, and secure alignment related to multimodal large models.

SECOND B. Wu Di born in February 1999 in Tai'an, Shandong, obtained a bachelor's degree from Shandong University in 2021 and a master's degree from Northwest University for Nationalities in 2024. During my master's degree, my main research direction was natural language processing, specifically in the field of machine reading comprehension. I conducted research on methods to enhance the semantic understanding ability of model texts and improve the machine reading comprehension ability of models. I have published two papers.In 2022 and 2023, he won the first and third academic scholarships of Northwest University for Nationalities, and the first prize of the "Internet plus" College Students Innovation and Entrepreneurship Competition.