Semantic Completeness-Enhanced Transformer Architecture for Named Entity and Relation Extraction in Noun Predicate Sentences

Bing Yan
School of Foreign Language, Hubei University of Education, Wuhan, 430205, China
E-mail: yanbingHUE@126.com

Keywords: semantic association, grammar pattern parsing, entity recognition, transformer, semantic completeness

Received: September 9, 2025

To solve the problem of semantic association and syntactic pattern parsing in noun predicate sentences, this paper proposed an entity recognition and relationship extraction method based on Transformer architecture. This method innovatively integrated three core components: enhancing text understanding ability through Similarity Whole Word Masking (Sim WWM) pre training, introducing semantic integrity quantification of text semantic coverage defects, and combining multi head semantic integrity mapping networks to achieve multi-dimensional complex feature learning. The experiment was based on the CoNLL04 dataset containing 1437 sentences and the SciERC dataset with 500 scientific abstracts. MTL-NER, SPKTE, DRPFF, and other baseline models were used. In entity recognition tasks, the proposed method achieved an average accuracy, recall, and F1 score of 90.7%, 91.6%, and 90.6% on the CoNLL04 dataset, and 69.8%, 70.4%, and 69.8% on the SciERC dataset, respectively; In the relationship extraction task, the three indicators on the CoNLL04 dataset were 75.3%, 73.4%, and 73.2%, respectively, and on the SciERC dataset were 62.9%, 60.5%, and 61.4%, respectively, all significantly better than the baseline model. The ablation experiment verification showed that the introduction of semantic integrity improved the model recognition accuracy by 3.9%, which was the core performance gain source. This method can effectively handle the problem of confusion between nested entities and entity types, providing an efficient technical path for complex semantic relationship parsing.

Povzetek: Članek predstavlja izboljšano metodo na osnovi Transformerja za prepoznavanje entitet in odnosov, ki dosega boljše rezultate od obstoječih pristopov.

1 Introduction

With the swift advancement of natural language processing (NLP) technology, semantic parsing of text data has become an important research direction. Among various text types, noun predicate sentences have attracted much attention due to their central position in information expression. Noun predicate sentences usually contain rich semantic information and complex grammatical structures, which can accurately describe the relationships between entities and the logic of events [1-2]. However, due to the complexity of its structure and the diversity of its semantics, automatically parsing the semantic associations and grammatical patterns of noun predicate sentences remains a challenging problem. In practical applications, accurate semantic association and syntactic pattern parsing are crucial for tasks such as information retrieval, knowledge graph (KG) construction, and machine translation. For example, in the field of medicine, accurately parsing noun predicate sentences in medical record texts can provide important support for disease diagnosis and treatment plan formulation. In the field of law, analyzing semantic relationships in legal texts helps to understand and apply legal provisions [3-4]. The existing semantic association and automatic parsing of grammar patterns mainly include two major tasks: named entity recognition and relation extraction. Entity recognition provides structured semantic information for text, which is helpful for further semantic analysis and understanding. Meanwhile, the recognized entities can serve as anchor points for syntax analysis, helping to parse sentence structures. Sentence relation extraction can reveal the semantic connections between entities, thereby enhancing the understanding of the overall semantics of the text. However, traditional parsing approaches often have limitations in dealing with complex semantic and syntactic structures, especially when facing problems such as nested entities and entity type confusion. The accuracy and robustness of traditional approaches are difficult to meet practical needs.

Extracting statement entity relationships (ERs) helps improve machines' semantic parsing and understanding capabilities for text. By identifying the relationships between entities, machines can better understand the deep meaning of text. Chen H et al. Introduced an approach based on heterogeneous graph attention network for extracting dialogue relationships. This approach simulates cross sentence relationships in conversations and utilizes heterogeneous graph attention networks to simulate various types of features in conversations. Through experiment, it is evident that the F1 score and F1_c score of this approach were superior to the state-of-the-art approach [5]. Suo et al. introduced a novel fragment annotation approach for complex problems in entity extraction tasks. Based on conventional fragment annotation, this approach combines pointer annotation approach to design an annotation approach of "traversal enumeration+group mapping", and based on this, a entity extraction model based on span extraction is designed. The F1 score of the model was significantly improved compared to the baseline model [6]. Han and Jia introduced an improved ER extraction approach based on the Bidirectional Encoder Representation from Transformers (BERT) algorithm to address the issue of insufficient coverage of large-scale manufacturing knowledge data in existing joint extraction approaches for knowledge data ERs. This approach treats the extraction of quantitative knowledge as the extraction of manufacturing entity attributes by constructing a KG, thereby achieving the joint extraction of knowledge reasoning and data ERs. The findings revealed that the KG generated by this approach had good expressive ability and strong logical reasoning ability, and the required running time was only 12 seconds [7]. Wang et al. introduced a comprehensive prediction network approach that includes multi-feature semantic fusion to address the issues of breadth expansion and information redundancy in ER extraction. This approach integrates entity mask embedding sequence and context embedding sequence to enhance interaction between triplets and expand semantic encoding. The findings demonstrated that the model outperformed the baseline model in regard to accuracy and F1 score on public datasets [8].

The advancement of deep learning technology has opened up fresh avenues for NLP. Transformer architecture and its variants perform well in processing natural language tasks, effectively capturing long-distance dependencies and contextual information in text. Khan A et al. introduced a hybrid visual transformer to address the limitations of visual transformers in terms of generalization ability. This approach combines convolution operations and self attention mechanisms to utilize local and global image representations. The hybrid vision Transformer achieved significant results in a range of computer vision tasks [9]. Feng K et al. developed a Transformer-based editing approach for shape aware objects in high-resolution images. This approach utilizes a diffusion Transformer to capture long-range dependencies between patches and uses a diffusion probability model solver inversion algorithm to reduce the number of steps. This approach surpassed existing frameworks in generating higher quality edited images [10]. Yeh et al. developed a new visualization approach for understanding the internal operation of self attention mechanism in Transformer models. This approach analyzes global patterns in multiple input sequences through visual queries and joint embedding of keyword vectors. This approach could improve model understanding and provide new insights into query key interactions [11]. He et al. introduced a segmentation approach based on an efficient hierarchical hybrid visual transformer to address the weak modeling of long-range dependencies in convolutional neural networks (CNNs) for medical image segmentation. This approach integrates the advantages of CNNs, multi-scale channel attention, and transformers, establishing long-range dependencies between pixels through self attention. This approach outperformed existing approaches in learning image segmentation tasks, and significantly reduced inference time [12]. The literature review summary is shown in Table 1.

Table 1: The literature review summary

Literature	Method	Key performance indicators	Limitation
[5]	Heterogeneous Graph Attention Network	The F1 score and F1_c score were better than the state-of-the-art methods at that time	Only applicable to dialogue scenarios, unknown generalization ability
[6]	Traditional fragment annotation+pointer annotation+"traversal enumeration+group mapping" annotation strategy+span extraction model	F1 score significantly improved compared to baseline model	Focusing on a single entity extraction task, without involving relationship extraction; Dependency on custom annotation strategy
[7]	Improved BERT algorithm+KG construction	Good KG expression ability and logical reasoning ability; Running time is only 12 seconds	Limited to the manufacturing industry, with poor universality
[8]	Multi feature semantic fusion+entity mask embedding+context embedding	Accuracy and F1 score are better than the baseline model	Optimization of complex grammatical structures for noun predicate sentences
[9]	Hybrid Visual Transformer (Convolution+Self Attention Mechanism)	Achieve significant results in multiple visual tasks	Hybrid architecture increases model complexity and reduces computational efficiency
[10]	Diffusion Transformer+Diffusion	Generate and edit	Relying on the diffusion

	Probability Model Solver Inverse Algorithm	images with better quality than existing frameworks	probability model solver, although reducing the number of steps, there is still a problem of slow inference speed and high deployment cost
[11]	Visual query+keyword vector joint embedding	Enhance understanding of Transformer attention mechanism	Only a theoretical analysis tool, without proposing substantial task performance optimization solutions, insufficient adaptability to complex long texts
[12]	Efficient Hierarchical Hybrid Visual Transformer (integrating CNN, multi-scale channel attention Transformer）	The segmentation performance is superior to existing methods; Significant reduction in inference time	Multi module fusion leads to a large scale of model parameters and requires high computational power for deploying hardware

In summary, although the current model has markedly enhanced the efficacy of various NLP tasks, it still faces some challenges in handling semantic associations and grammatical patterns of noun predicate sentences. There are still three major difficulties in parsing noun predicate sentences: difficulty in segmenting the boundaries of multi-layer nested entities (such as “Artificial Intelligence Laboratory of XX University School of Computer Science”); The same word (“contract”) drifts in different scenarios and lacks global semantic validation; The implicit relationship (“drug hypertension” treatment) and overlapping triplet (“doctor patient drug”) are frequently missed or misjudged due to the lack of semantic integrity modeling, directly reducing accuracy and cross domain generalization. The root cause of the failure of existing methods is that pre trained models only perform local associations and lack self checking of whether the semantic framework is complete, resulting in missed detection of nested attributes and implicit elements; Rules or classic multitasking systems rely on fixed templates/single features to cope with multiple nested layers, resulting in weak structural adaptability; Domain customization solutions rely on manual knowledge injection and can only maintain a single scenario, without feeling the cross domain rules of generic noun predicate sentences, and still face difficulties in type confusion and transfer. The research zeroes in on three key challenges: (1) Accurately Segmenting Nested Entity Levels: It explores how to achieve this by leveraging semantic integrity mechanisms. (2) Eliminating Type Confusion: The study addresses how to overcome this issue through multidimensional semantic mapping. (3) Injecting Integrity Signals into Transformers: It investigates how to incorporate these signals to simultaneously enhance the accuracy of extracting implicit/overlapping relationships and improve cross-domain generalization capabilities. Therefore, to tackle the above issues, a Transformer architecture-based approach for automatically parsing semantic associations and grammatical patterns of noun predicate sentences is introduced. The novelty of the study consists in the introduction of semantic completeness and multi head semantic completeness mapping networks, which significantly enhance the model's ability to process complex text data. Secondly, Similarity Whole Word Masking (Sim-WWM) is introduced to improve the model's ability to understand text. The research not only provides a new technological path for semantic parsing of noun predicate sentences, but also provides useful references for further development in the field of NLP.

2 Methods and materials

One of the important research directions in the field of NLP is the semantic association and grammatical pattern parsing of noun predicate sentences. Therefore, to achieve semantic association and grammatical pattern parsing of noun predicate sentences, the study introduces a sentence analysis approach based on Transformer and BERT. This approach first identifies the entities of the statement to accurately represent the associated information of the statement. Next, based on the recognition of the associated information mentioned above, relationships are extracted to achieve the parsing of semantic associations and grammatical patterns in sentences.

2.1 Semantic association subject recognition approach based on Transformer and semantic completeness

Before performing automatic parsing of semantic associations and grammatical patterns in noun predicate sentences, it is necessary to first identify the semantic association subject, that is, the entity, of the sentence in order to accurately express the association information of the sentence. Named entity recognition refers to identifying entities with specific meanings from text, which is one of the fundamental tasks of semantic association and syntactic pattern parsing. Identifying named entities in the text enables a clearer comprehension of its semantic content and facilitates the determination of the grammatical roles of each component within the sentence. In entity recognition, to achieve entity disambiguation and context association, it is necessary to calculate semantic similarity. The current semantic similarity is generally represented by cosine similarity. The calculation of semantic similarity is presented in equation (1).

$$s = \cos(\theta) = \frac{A * B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2}\sqrt{\sum_{i=1}^{n}(B_i)^2}} \quad (1)$$

In equation (1), s represents semantic similarity. θ represents the vector angle. A and B represent different vectors. A_i and B_i represent the components of vectors A and B in the i th dimension. n represents the vector dimension. However, relying solely on semantic similarity is not enough to reflect the semantic differences between different sentences. For example, multiple vectors in two-dimensional space can form the same angle with the same vector. The generatrix of a cone in three-dimensional space is similar to the cosine of its axis, but at different positions [13-15]. Therefore, to address the aforementioned issues, the study introduces semantic completeness. In the d-dimensional text vector space, semantic integrity quantifies the semantic differences between the original text vector and the masked text vector through a mapping function, measuring the degree of coverage of the target semantic framework core elements by the text. The range of values is [0,1], with closer values to 1 indicating more complete semantics. The multi head semantic integrity mapping network has four independent heads that focus on entity attributes, event elements, logical relationships, and contextual dimensions; The output features of each head are horizontally concatenated and fused into a unified semantic integrity vector through linear mapping, achieving multi-dimensional feature complementarity. Unlike the multi head attention mechanism, semantic integrity focuses on "semantic coverage evaluation", quantifying the degree of complete expression of specific semantic concepts in the current text fragment by comparing the representation differences between the original text and the masked text. Continuing with the above example, semantic integrity will automatically detect the coverage of the semantic framework of "doctor (subject) - basis (CT report) - diagnosis (action) - lung cancer (object)". If the sentence is missing the "CT report", it will significantly reduce the semantic integrity score, thereby assisting the model in identifying nested or hidden entities. The expression for semantic completeness is presented in equation (2).

$$S_c = M\left(t^m, t\right) \quad (2)$$

In equation (2), S_c represents semantic completeness. M represents the mapping function. t^m is the masked text representation vector. t is the original text representation vector. After constructing the above concepts, statement entity recognition can be carried out. The selected entity recognition approach for research is based on fragment arrangement, which utilizes Transformer architecture for context representation and achieves named entity recognition by identifying and classifying fragments in text. The entity recognition model based on fragment arrangement is presented in Figure 1.

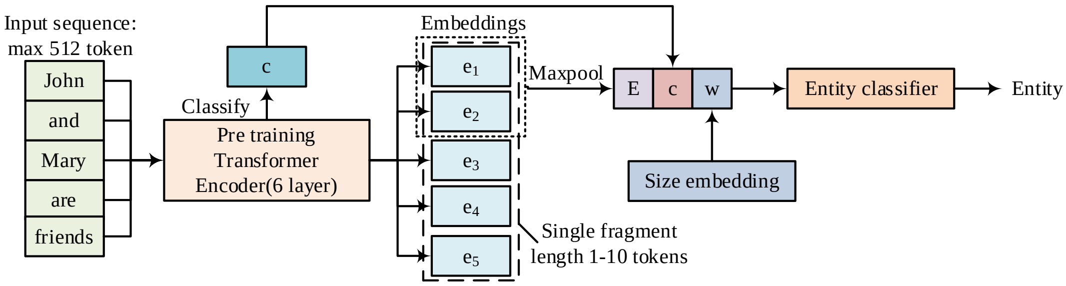


Figure 1: Entity recognition model based on fragment arrangement(Source: Author's own drawing)

As shown in Figure 1, the model first extracts all possible fragments from the text and generates a vector representation for each fragment. Next, it determines whether each fragment is a named entity. At this point, if multiple segments overlap and are recognized as entities, it is necessary to merge the relevant segments [16-17]. When using the above approach for entity recognition, considering the richness of feature representation, the study introduces span embedding. The vector calculation formula for entity classification after introducing span embedding is presented in equation (3).

$$x^e = e(s) \circ w \quad (3)$$

In equation (3), x^e represents the vector used for entity classification. $e(s)$ represents a vector. w represents word embeddings. Although the above approaches can achieve recognition of sentence entities, the noise problem in fragment classification is relatively serious. Therefore, to address this issue, the study introduces the aforementioned semantic completeness into entity recognition models. The improved entity recognition model is presented in Figure 2.

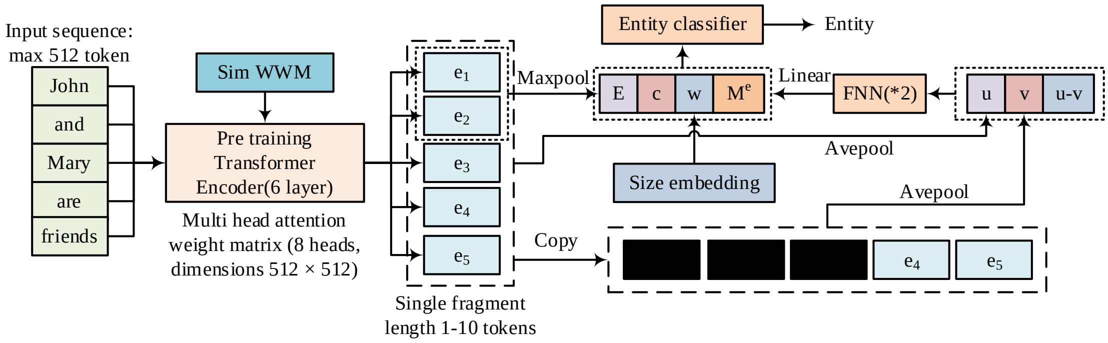


Figure 2: Improved entity recognition model(Source: Author's own drawing)

As shown in Figure 2, compared to the original entity recognition model based on fragment arrangement, the improved entity recognition model introduces Sim-WWM and semantic completeness, where Sim-WWM is used for pre training the Transformer. Meanwhile, by copying the word embedding sequence and masking the current subsequence, a masking sequence is obtained. Next, the original text sequence and the masked sequence are averaged to fuse the original information and partially masked information. Next, fusion vectors are used to calculate text similarity and evaluate the similarity of text content. Then, the fused vector is transformed through linear mapping and fused with other vectors. Finally, semantic completeness can be obtained through nonlinear mapping using a Feedforward Neural Network (FNN). The standard masking mechanism (such as BERT's MLM and WWM) has two major limitations: First, the masking strategy is randomized and does not consider the semantic relevance of the text, which may damage the entity integrity of noun predicate sentences (such as random masking of entity core words leading to semantic fragmentation); Secondly, it is difficult to capture the global semantic framework of "entity attribute relationship" in noun predicate sentences by only learning local semantics through a single masking method. Unlike the standard masking mechanism, Sim WM uses a "semantic similarity-guided masking strategy" to construct more targeted counterfactual samples while preserving the core semantic structure of the sentence, which helps the model learn semantic integrity. Specifically, the semantic core of noun predicate sentences relies on entities and their associated relationships. Sim WWM prioritizes masking non-core semantic words while preserving entities and key relationship words, creating semantic differences to train the model's integrity awareness ability. This approach avoids semantic framework collapse caused by masking core entities, thereby more accurately assisting in semantic integrity assessment. Moreover, Sim WM is not only applicable to noun predicate sentences, its core design logic is a "masking strategy guided by semantic similarity", which has cross sentence and cross domain universality. The calculation formula for the fused text representation vector is presented in equation (4).

$$r = t \circ t^{m} \circ |t - t^{m}| \quad (4)$$

In equation (4), r represents the fused text representation vector. At this point, the entity classification calculation formula is presented in equation (5).

$$\begin{cases} y^{e} = softmax\left(W^{e} * (t \circ c \circ w \circ M^{e}) + b^{e}\right) \\ M^{e} = Linear(o) \end{cases} \quad (5)$$

In equation (5), y^{e} represents the entity classification result. W^{e} represents the weight matrix. c represents the overall representation vector of the text sequence. M^{e} represents the text representation vector after linear mapping. b^{e} represents the boundary label. To use the linearly mapped text representation vector to represent semantic completeness, it is necessary to ensure that the number of layers in the FNN is not zero. At this point, the formula for calculating the text representation vector after linear mapping is presented in equation (6).

$$M^{e} = Linear\left(f^{L}(o)\right) \quad (6)$$

In equation (6), f^{L} represents FNN. L represents the number of layers in FNN. Due to the possibility of information loss caused by a single feedforward network, a multi-head semantic completeness mapping network is introduced to address the aforementioned issues. The multi-head semantic completeness mapping network is presented in Figure 3.

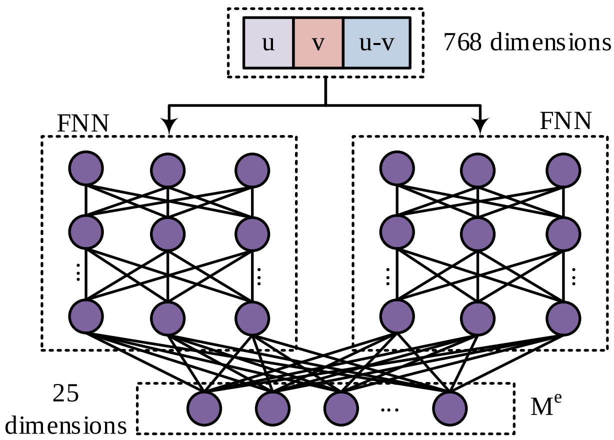


Figure 3: Multi-head semantic completeness mapping network (Source: Authors own drawing)

In Figure 3, the multi-head semantic completeness mapping network can utilize multiple FNNs to map input vectors to different feature spaces, each of which focuses on capturing different aspects or levels of semantic information in the input data. Due to the ability of this network to map vectors into representation vectors of different levels, there are differences in the vector structure in each path. By using the above approach, entity recognition of noun predicate sentences can be achieved for the purpose of extracting sentence relationships in the future.

2.2 approach for extracting sentence relationships considering text dependencies

After identifying the entities of noun predicate sentences using the above approach, the sentence relationships can be extracted. RE holds a pivotal position in semantic association and automatic parsing of grammar patterns. It not only helps to better understand the semantic structure of text, but also provides important clues and basis for grammar analysis. The purpose of RE is to identify entity pairs with specific semantic relationships in text, but traditional RE approaches rely on entity positional relationships, resulting in lower accuracy in relationship recognition [18-19]. Therefore, to address the aforementioned issues, the study introduces semantic completeness into the RE model and designs a new sentence partitioning scheme. The sentence partitioning scheme is presented in Figure 4.

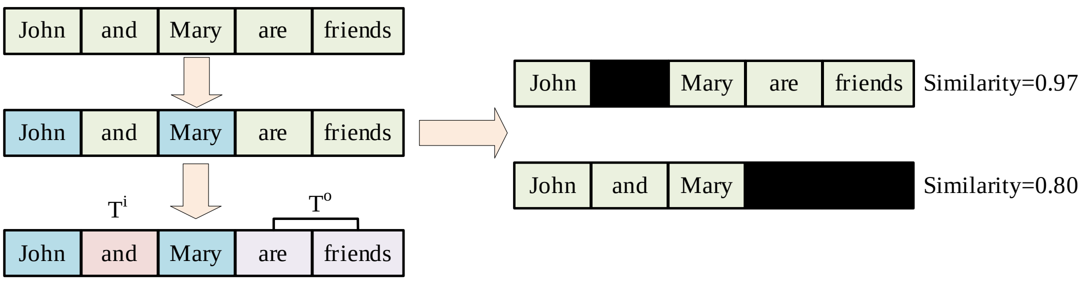


Figure 4: Statement division scheme (Source: Authors own drawing)

As shown in Figure 4, the sentence partitioning scheme designed in the study takes entity pairs as boundaries and divides the remaining words into words between entity pairs and other words. At this point, cover up non entity words and determine the importance of words in different positions to the sentence based on their similarity changes. Finally, the semantic completeness vector can be obtained through nonlinear mapping processing. After introducing the above concepts, the sentence RE model is presented in Figure 5.

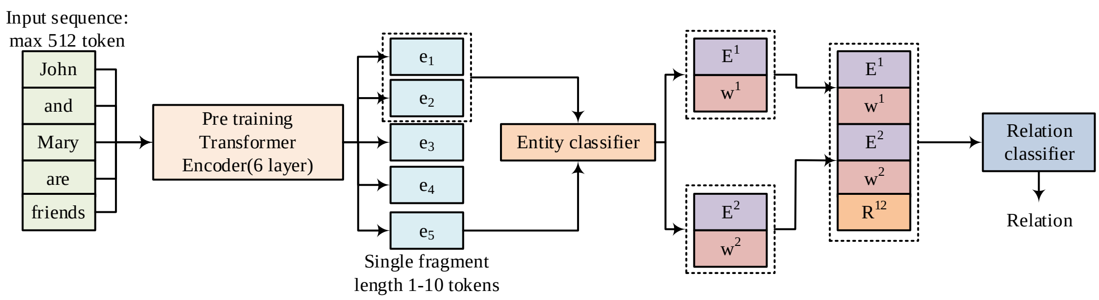


Figure 5: Sentence RE model (Source: Authors own drawing)

As shown in Figure 5, after receiving the text sequence, the model will generate embedding representations for each word through a pre trained Transformer encoder. Next, the entity classifier in the model is used to identify entities in the text, and all entities are used as nodes to construct a bidirectional fully connected graph [20-21]. Then, the model calculates the semantic completeness representation vector when covering different entities. Finally, a relationship classifier is employed to identify the relationships between entities. The calculation formula for a bidirectional fully connected graph is presented in equation (7).

$$U^n = \left\{ r^{12}, \cdots, r^{1n}; r^{2,1}, \cdots, r^{2n}; r^{n1}, \cdots, r^{n(n-1)} \right\} (7)$$

In equation (7), U^n represents a bidirectional fully connected graph of relationships. $r^{n(n-1)}$ represents the relationship between entity n and entity $n-1$. The formula for calculating the relationship representation vector of entities is presented in equation (8).

$$x^{ni} = E^n \circ w^n \circ E^i \circ w^i \circ R^{ni} (8)$$

In equation (8), x^{ni} represents the relationship vector between different entities. E^n and E^i represent word embedding vectors. w^n and w^i are span embeddings. R^{ni} represents the relationship dependency vector between different entities. Due to the division of the text in the study, the formula for calculating the relationship dependency vector can be rewritten as equation (9).

$$R = f\left(T^i, T^o\right) (9)$$

In equation (9), R represents the relationship dependency vector. T^i represents vocabulary between entities. T^o represents the vocabulary on both sides of the entity. Based on this, the details of the sentence RE model are presented in Figure 6.

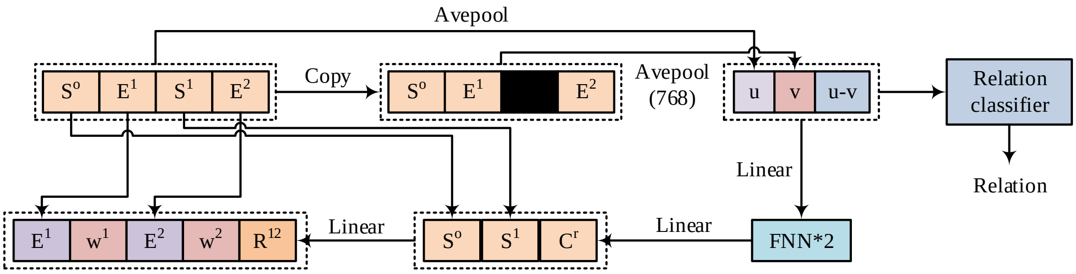


Figure 6: Details of the statement RE model(Source: Author's own drawing)

As shown in Figure 6, after inputting text, the model first converts the text into a sequence of word embeddings and divides the sentences according to the partitioning scheme. Next, the model processes the word embedding vectors through masking and replication mechanisms, and averages and pools the masked embedding sequence to obtain a comprehensive representation. Then, FNN is used to extract features from the average pooled vector. Next, the output of FNN is fed into the linear layer to generate the final relationship representation vector. Finally, a relationship classifier is used to identify the relationship representation vectors. The calculation formula for the pooled text representation vector is presented in equation (10).

$$u = avepooling(T_1^o \circ E^1 \circ T^i \circ E^2 \circ T_2^o) \quad (10)$$

In equation (10), u represents the average pooled text representation vector. The masked text representation vector is presented in equation (11).

$$v = avepooling(T_1^o \circ E^1 \circ T^{mask} \circ E^2 \circ T_2^o) \quad (11)$$

In equation (11), v represents the masked text representation vector. T^{mask} represents an all zero vector. The semantic completeness representation vector can be obtained by calculating the difference between the pooled and masked text representation vectors. The formula for calculating the semantic completeness representation vector is presented in equation (12).

$$S_c^r = Linear\left(f^L(u \circ v \circ |u-v|)\right) \quad (12)$$

In equation (12), S_c^r represents the semantic completeness representation vector. At this point, the formula for calculating the relationship dependency vector can be rewritten as equation (13).

$$R = f(T^i, T^o) = Linear(T^i, T^o, S_c^r) \quad (13)$$

The fusion of statement information and semantic completeness vector can be achieved through a fully connected layer, thereby enabling the semantic completeness representation vector to affect each statement element [22]. By using the above approaches, the model can learn more complex features, thereby improving the accuracy of sentence relation extraction. The calculation formula for the relationship classifier of the model is presented in equation (14).

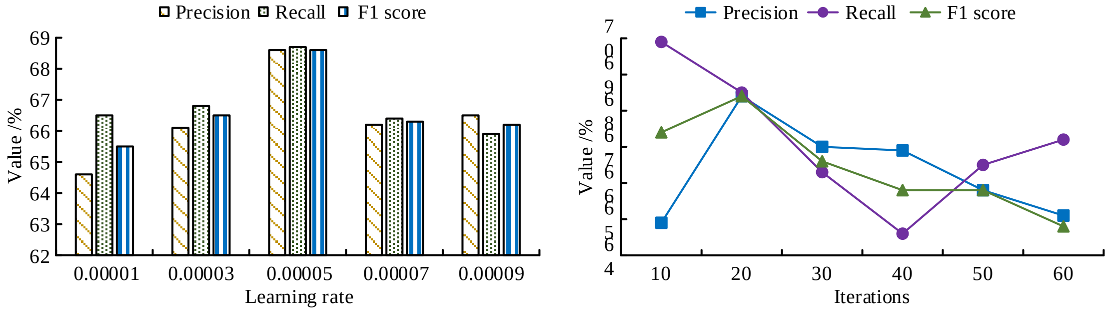
$$r^{ni} = softmax\left(W^r * (E^n \circ w^n \circ E^i \circ w^i \circ R) + b^r\right) \quad (14)$$

In equation (14), r^{ni} represents the classification result. W^r represents the weight matrix. b^r represents the boundary label of the relationship. Meanwhile, to further improve the accuracy of RE in the model, the study also introduces a negative sample mechanism, which constructs negative samples outside the labeled positive samples to help the model learn more features and patterns, and improve its generalization ability on unseen data. By using the above approaches, it is possible to extract sentence relationships, reveal semantic connections between different sentences in the text, and construct the discourse structure of the text to better understand the organization of the text, thereby achieving automatic parsing of semantic associations and grammatical patterns.

3 Results

3.1 Entity recognition test results

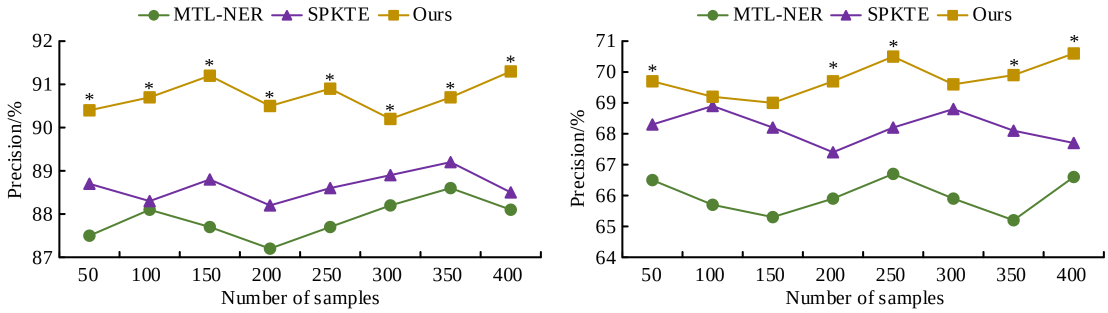
To validate the performance of the introduced entity recognition model, it was tested and compared with the Mixed Transfer Learning Named Entity Recognition (MTL-NER) model and the Entity Recognition model based on Semantic Prior Knowledge and Type Embedding (SPKTE). The CPU used in the experiment was Intel core i7 4770K, and the GPU was NVIDIA GeForce RTX 4050. The datasets used in the experiment were CoNLL04 dataset and SciERC dataset, where CoNLL04 dataset contained 1437 sentences and SciERC dataset contained 500 scientific abstracts. In the experiment, the learning rate and maximum gradient norm of the model were 0.00005 and 1.0.The span embedding size and maximum negative sample size were 25 and 100. The Batchsize, maximum fragment span, and semantic completeness embedding sizes were 2, 10, and 25. The model training cycle (Epoch) was 30 rounds, and an early stopping strategy (Early Stopping) was adopted to prevent overfitting. When the F1 score of the validation set did not improve for 5 consecutive rounds, the training was automatically terminated and the optimal model weights are saved. The experiment adopted a binary segmentation strategy of “development set test set”, with the dataset divided in a 7:3 ratio and the development set split from the training set in a 10% ratio for hyperparameter tuning and early stop judgment. The study set fixed random seeds (Python random seed=42, PyTorch random seed=42, NumPy random seed=42) to ensure that the results of data partitioning, model initialization, masking strategy, and other steps are reproducible. The model used the AdamW optimizer with a weight decay set to 0.01, β_1=0.9, β_2 =0.999, ε =1e^{-8}; The learning rate scheduling strategy adopted linear preheating+cosine annealing, with 3 preheating rounds and an initial learning rate of 0.00005. The loss function adopted a cross entropy loss function with category balanced weights. The baseline model referred to the network structure and hyperparameter configuration of the original paper, and was autonomously re-implemented based on PyTorch to ensure consistency with the training environment (hardware, optimizer) of the model in this paper. The impact of learning rate and iteration times on model performance is presented in Figure 7.



(a) Learning rate optimization (b) Iteration optimization

Figure 7: Influence of learning rate and iteration times on model performance (Source: Authors own drawing)

According to Figure 7 (a), as the learning rate increased, the accuracy, recall, and F1 score all first increased and then decreased. When the learning rate was 0.00005, the performance of the model reached its optimal level. At this point, the accuracy, recall, and F1 score of the model were 68.6%, 68.7%, and 68.6%. As shown in Figure 7 (b), with the increase of iteration times, the accuracy and recall of the model first increased and then decreasde, while F1 score first decreased and then increased. After comprehensive consideration, the performance of the model was considered optimal when the number of iterations was 20. At this point, the accuracy, recall, and F1 score of the model were 68.4%, 68.5%, and 68.4%. The performance of the model was optimal when the learning rate and iteration times were 0.00005 and 20. The accuracy of entity recognition in different datasets is presented in Figure 8.

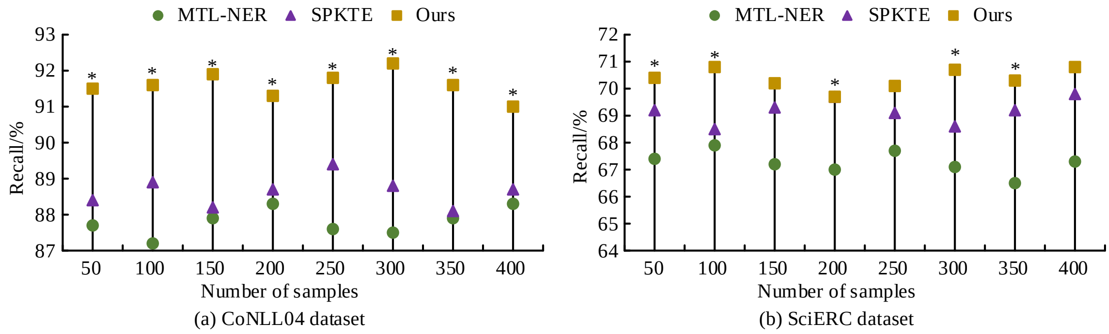


(a) CoNLL04 dataset (b) SciERC dataset

Note: * indicates p<0.05.

Figure 8: Entity recognition accuracy in different data sets (Source: Authors own drawing)

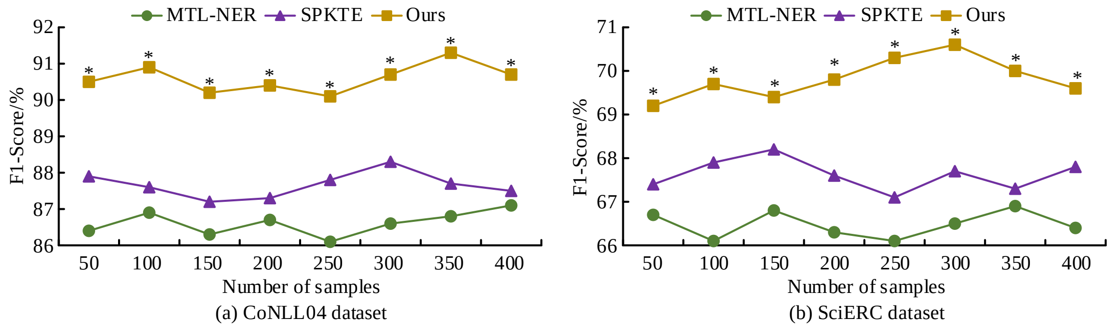
According to Figure 8 (a), in the CoNLL04 dataset, the recognition accuracies of MTL-NER and SPKTE were the highest at 88.6% and 89.2%, and the lowest at 87.2% and 88.2%. The average accuracies were 87.9% and 88.7%. The lowest accuracy of the entity recognition model introduced in the study was 90.2% (p<0.05), with an average accuracy of 90.7% (p<0.05), which was higher than other approaches. According to Figure 8 (b), in the SciERC dataset, the recognition accuracies of MTL-NER and SPKTE were the highest at 66.7% and 68.9%, and the lowest at 65.2% and 67.4%. The average accuracies were 66.0% and 68.2%. The lowest accuracy of the entity recognition model introduced in the study was 69.0% (p<0.05), with an average accuracy of 69.8% (p<0.05), which was higher than other approaches. The introduced entity recognition approach could achieve accurate recognition of named entities. The recall rates in different datasets are presented in Figure 9.



Note: * indicates $p<0.05$.

Figure 9: Recall rates in different data sets (Source: Authors own drawing)

According to Figure 9 (a), in the CoNLL04 dataset, the recall rates of MTL-NER and SPKTE were the highest at 88.3% and 89.4%, respectively, and the lowest at 87.2% and 88.1%. The average recall rates were 87.8% and 88.7%. The lowest recall rate of the entity recognition model introduced in the study was 91.0% ($p<0.05$), with an average recall rate of 91.6% ($p<0.05$), which was higher than other approaches. According to Figure 9 (b), in the SciERC dataset, the recall rates of MTL-NER and SPKTE were the highest at 67.9% and 69.8%, and the lowest at 66.5% and 68.5%. The average recall rates were 67.3% and 69.2%. The lowest recall rate of the entity recognition model introduced in the study was 69.7% ($p<0.05$), with an average recall rate of 70.4% ($p<0.05$), which was higher than other approaches. It can be concluded that the entity recognition model introduced in the study had good generalization ability. The F1 score in different datasets is presented in Figure 10.



Note: * indicates $p<0.05$.

Figure 10: F1 score in different data sets (Source: Authors own drawing)

As shown in Figure 10 (a), in the CoNLL04 dataset, the F1 scores of MTL-NER and SPKTE were the highest at 87.1% and 88.3%, and the lowest at 86.1% and 87.2%. The average F1 scores were 86.6% and 87.7%. The lowest F1 score of the entity recognition model introduced in the study was 90.1%, with an average F1 score of 90.6% ($p<0.05$), which was higher than other approaches. According to Figure 10 (b), in the SciERC dataset, the F1 scores of MTL-NER and SPKTE were the highest at 66.9% and 68.2%, and the lowest at 66.1% and 67.1%. The average F1 scores were 66.5% and 67.6%. The lowest F1 score of the entity recognition model introduced in the study was 69.2%, with an average F1 score of 69.8% ($p<0.05$), which was higher than other approaches. With the above outcomes, it can be concluded that the entity recognition model introduced in the study had good robustness. The ablation experiment results of the model are presented in Table 2.

Table 2: Results of ablation experiments

Model	Multiple mechanisms	FNN	Semantic completeness	Precision/%
1	×	×	×	82.3
2	√	×	×	84.4
3	×	√	×	83.8
4	×	×	√	86.2
5	√	√	×	87.6
6	√	×	√	88.9
7	×	√	√	89.3
8	√	√	√	90.7

According to Table 2, after introducing the multi-head mechanism, FNN, and semantic completeness, the recognition accuracy of the model significantly increased from 82.3% to 90.7%. Among them, semantic completeness had the most significant impact on model performance. After introducing semantic completeness, the recognition accuracy of the model increased by 3.9%. Comparing Model-1 (with no semantic integrity) and Model-4 (with only semantic integrity), the accuracy increased from 82.3% to 86.2%, an increase of 3.9 percentage points. This improvement was entirely driven by the integrity scoring component, as Model-4 did not introduce multi head mechanisms and FNN, and only achieved performance improvement through the quantification of differences between raw text and masked text (integrity scoring core logic), proving that this component was the core functional carrier of the semantic integrity framework. The accuracy comparison between Model-4 (semantic integrity only) and Model-7 (semantic integrity+FNN) showed that after adding FNN, the accuracy increased from 86.2% to 89.3%, an increase of 3.1 percentage points. Combining method design, FNN was the core implementation of feature aggregation (equation 6), which functions to perform nonlinear transformation on the original semantic difference features and enhance feature discrimination. This indicated that the feature aggregation component provided more effective input for integrity scoring by optimizing feature quality, indirectly improving the accuracy of semantic integrity evaluation. The introduced improvement approach could effectively improve the accuracy of statement entity recognition. To verify the novelty of the method proposed in this paper, pre trained models such as BERT base and RoBERTa base, which are widely used in entity recognition and relationship extraction tasks, were selected as baselines. The performance, core improvements, and applicable scenarios were compared, with a focus on analyzing the unique contribution of semantic integrity. The comparison results of pre trained language models are shown in Table 3.

Table 3: Comparison results of pre trained language models

Model	Task	CoNLL04	SciERC
BERT-base	Entity recognition	85.9%*	64.9%*
	Relationship extraction	69.5%*	56.0%*
RoBERTa-base	Entity recognition	87.8%*	67.1%*
	Relationship extraction	71.9%*	58.2%*
Ours	Entity recognition	90.6%*	69.8%*
	Relationship extraction	73.2%*	61.4%*

Note: * indicates $p<0.05$.

According to Table 3, in the entity recognition task, the F1 score of the proposed method on the CoNLL04 dataset was 2.8% higher than that on the RoBERTa base dataset, and 2.7% higher on the SciERC dataset; In the relationship extraction task, the F1 score of our method on the CoNLL04 dataset was 1.3% higher than that on the RoBERTa base dataset, and 3.2% higher on the SciERC dataset; As the complexity of the dataset increased (SciERC is a scientific abstract that includes more nested entities and specialized terminology), the performance advantage of the proposed method became more significant, verifying the adaptability of semantic integrity to complex semantic scenarios. To verify the robustness and generalization ability of the model in different professional fields, additional experiments were conducted using the BioRED dataset in the biomedical field and the CAIL2020 dataset in the legal field. The experimental results are shown in Table 4.

Table 4: Cross disciplinary experimental results

Model	Task	BioRED	CAIL2020
MTL-NER	Entity recognition	77.6%*	80.8%*
	Relationship extraction	69.3%*	72.8%*
SPKTE	Entity recognition	79.3%*	82.6%*
	Relationship extraction	71.9%*	75.1%*
Ours	Entity recognition	84.3%*	87.1%*
	Relationship extraction	77.3%*	80.0%*

Note: * indicates $p<0.05$.

According to Table 4, cross domain experiments (BioRED, CAIL2020, and other 4 datasets) showed that the entity recognition F1 score of the research method was 84.3%/87.1%, and the relationship extraction F1 score was 77.3%/80.0%, both leading the baseline by 4-8 percentage points; The coefficient of variation was ⩽ 7.5%, significantly lower than the comparison model, verifying the high accuracy and strong generalization of the multi head semantic integrity network.

3.2 RE test results

To validate the performance of the RE model introduced in the study, it was tested and compared with the ER extraction model based on Dual Relationship Prediction and Feature Fusion (DRPFF) and the Aggregation Logic Network Based on Binary Graph Structure (BoBGSAL-

Net). The dataset, software and hardware settings, and hyperparameters used in the experiment were consistent with the above experiment. In the CoNLL04 and SciERC datasets, the filter thresholds were 0.4 and 0.3. The triple used for relationship extraction tasks in the experiment was defined as a structured data unit of “entity pair relationship”, consisting of subject entities, object entities, and semantic relationships between subjects and objects. The validity determination of triplets required two conditions: 1) both the subject entity and the object entity are valid entities confirmed by the entity recognition model (entity types meet the definition of the dataset category); 2) ERs refer to the semantic relationships that truly exist between entities in the original text, generated through non random combinations. To ensure the representativeness and balance of experimental data, the triplet sampling adopted a “type stratification+proportional control” strategy. Positive samples were stratified by entity type combination, with 500-800 (training)/100-200 (testing) and a 1:1 balanced relationship category; Negative samples were generated by shuffling random entities of the same type, with a 3:1 ratio and class by class alignment. Pseudo examples with self matching and distance<2 tokens were removed to ensure balanced distribution and semantic rationality. The preprocessing method for triplets was as follows: boundary calibration merges cross word entities, fuzzy relationships were uniformly mapped, and unlabeled sentences were deleted; Long sentences with entity pairs as the center, truncated by ± 50 tokens to prevent fragmentation; Use (es, eo, r) to remove duplicates and eliminate type mismatch samples. Finally, the study left 8k-10k for training and 1.5k-2k for testing, with a distribution variation of ⩾ 10% to maintain balance. The parameter settings are shown in Table 5.

Table 5: The parameter settings

Parameter	Value
Weight decay	0.01
Optimizer coefficient ($\beta_1 / \beta_2 / \varepsilon$)	0.9/0.999/1e-8
Optimizer coefficient	0.00005
Maximum gradient norm	1.0
Span embedding dimension	25
Semantic integrity embedding dimension	25
Batch size	2

According to Table 5, the weight decay of the model was 0.01, the initial learning rate was 0.00005, and both the span embedding dimension and semantic integrity embedding dimension were 25. The accuracy of RE from different datasets is presented in Figure 11.

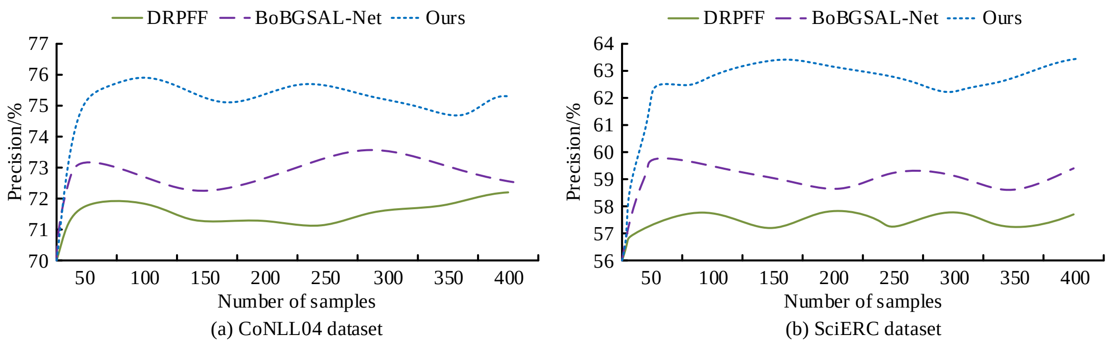
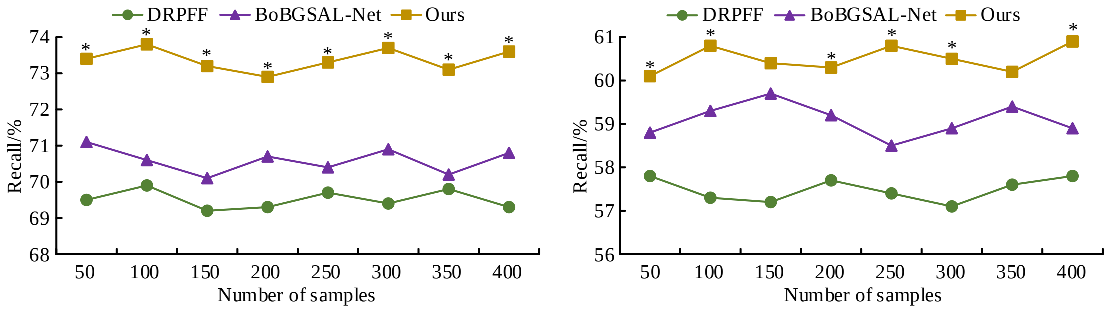


Figure 11: Precision of relation extraction in different data sets (Source: Authors own drawing)

According to Figure 11 (a), in the CoNLL04 dataset, the highest accuracy of DRPFF and BoBGSAL-Net was 72.2% and 73.5%, while the lowest was 71.1% and 72.2%. The average accuracy was 71.6% and 72.9%. The introduced RE model had a minimum accuracy of 75.1% and an average accuracy of 75.3%, which was higher than other approaches. According to Figure 11 (b), in the SciERC dataset, DRPFF and BoBGSAL-Net had the highest accuracies of 57.9% and 59.8%, and the lowest accuracies of 57.1% and 58.6%. The average accuracies were 57.5% and 59.2%. The introduced RE model had a minimum accuracy of 62.2% and an average accuracy of 62.9%, which was higher than other approaches. It is evident that the introduced RE model could accurately extract sentence relationships. The recall rates in different datasets are presented in Figure 12.

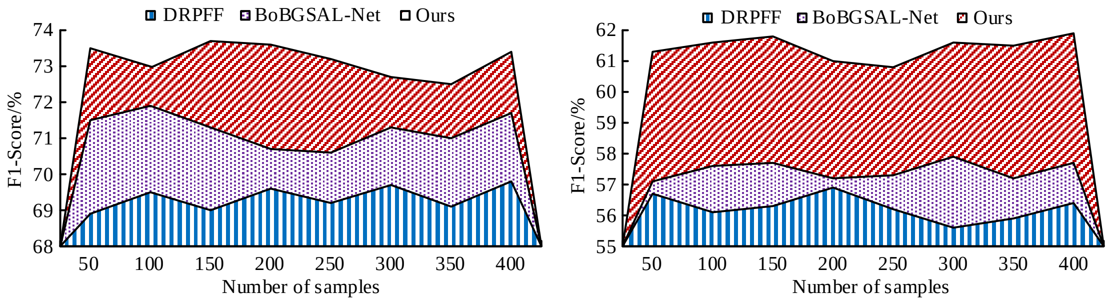


(a) CoNLL04 dataset (b) SciERC dataset

Note: * indicates $p<0.05$.

Figure 12: Recall rates in different data sets (Source: Authors own drawing)

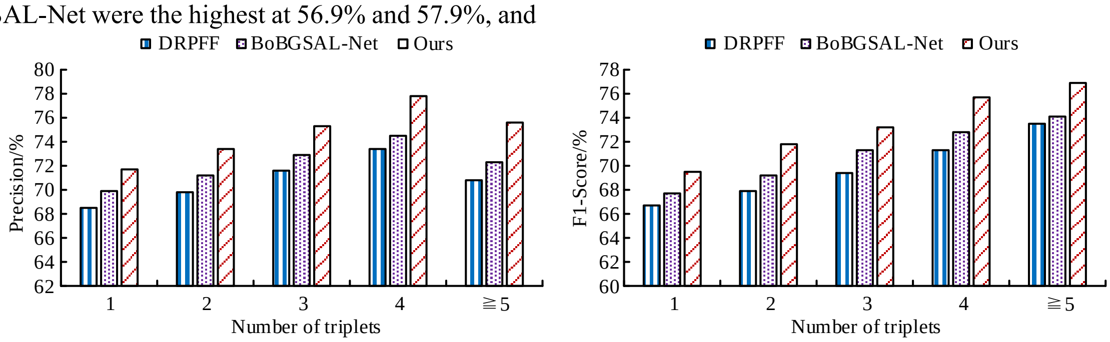
According to Figure 12 (a), in the CoNLL04 dataset, the recall rates of DRPFF and BoBGSAL-Net were the highest at 69.9% and 71.1%, and the lowest at 69.2% and 70.1%. The average recall rates were 69.5% and 70.6%. The lowest recall rate of the RE model introduced in the study was 72.9%, with an average recall rate of 73.4% ($p<0.05$), which was higher than other approaches. According to Figure 12 (b), in the SciERC dataset, the recall rates of DRPFF and BoBGSAL-Net were the highest at 57.8% and 59.7%, and the lowest at 57.1% and 58.5%. The average recall rates were 57.5% and 59.1%. The lowest recall rate of the RE model introduced in the study was 60.1%, with an average recall rate of 60.5% ($p<0.05$), which was higher than other approaches. The introduced RE model had good generalization ability. The F1 score in different datasets is presented in Figure 13.



(a) CoNLL04 dataset (b) SciERC dataset

Figure 13: F1 score in different data sets (Source: Authors own drawing)

According to Figure 13 (a), in the CoNLL04 dataset, the F1 scores of DRPFF and BoBGSAL-Net were the highest at 69.8% and 71.9%, and the lowest at 68.9% and 70.7%. The average F1 scores were 69.4% and 71.3%. The RE model introduced in the study had a minimum F1 score of 72.5% and an average F1 score of 73.2%, which was higher than other approaches. According to Figure 13 (b), in the SciERC dataset, the F1 scores of DRPFF and BoBGSAL-Net were the highest at 56.9% and 57.9%, and the lowest at 55.6% and 57.1%. The average F1 scores were 56.3% and 57.5%. The F1 score of the introduced RE model was the lowest at 60.8%, with an average F1 score of 61.4%, which was higher than other approaches. The introduced RE model had good robustness. Taking the CoNLL04 dataset as an example, the accuracy and F1 score under different numbers of triples are presented in Figure 14.



(a) Precision (b) F1-Score

Figure 14: Accuracy and F1 score under different triplet numbers (Source: Authors own drawing)

As shown in Figure 14 (a), with the increase of the number of triples, the accuracy of each model first increased and then decreased. The accuracy of the RE model introduced in the study was consistently higher than other models.

When the number of triples was 4, the accuracy of the three models was the highest, at 73.4%, 74.5%, and 77.8%. As shown in Figure 14 (b), with the increase of the number of triples, the F1 score of each model gradually increased. The F1 score of the RE model introduced in the study was consistently higher than other models. When the number of triples was 4, the F1 score of each model was 71.3%, 72.8%, and 75.7%. It is evident that the introduced RE model had excellent performance. To quantify the added value of the negative sample mechanism to the RE model, ablation experiments were conducted on it. The experimental results of negative sample mechanism ablation are shown in Table 6.

Table 6: The experimental results of negative sample mechanism ablation

Model	Index	CoNLL04	SciERC
Negative sample mechanism	Accuracy/%	75.3	62.9
	F1 score/%	73.2	61.4
Without negative sample mechanism	Accuracy/%	71.8	59.3
	F1 score/%	69.7	57.9
Performance change amplitude	Accuracy/%	+3.5	+3.6
	F1 score/%	+3.5	+3.5

According to Table 6, after ablating the negative sample mechanism, the F1 score of CoNLL04 dataset decreased by 3.5 percentage points, and the F1 score of SciERC dataset also decreased by 3.5 percentage points. Both accuracy and recall rates showed significant declines, proving that the negative sample mechanism had a clear positive contribution to model performance. This was because the negative sample mechanism helps the model learn the semantic boundaries of entity pairs by providing comparative samples of “invalid relationships”.

4 Discussion

NLP is an important branch of artificial intelligence, with the goal of enabling computers to understand, generate, and process human language. With the advent of the big data era, text data is growing explosively, containing a large amount of unstructured information related to spatial location, semantic relationships, and so on. To extract valuable information from these data, efficient semantic association and syntactic pattern parsing techniques are required[23-24]. Although current models have achieved good results in recognizing simple semantic relationships, there are still shortcomings in handling complex semantic relationships. For example, in some complex sentence structures, the relationships between entities may need to be deduced through multiple intermediate entities, and existing models may struggle to accurately capture these complex relationships [25-26]. In view of this, an approach for automatically parsing semantic associations and grammatical patterns of noun predicate sentences based on Transformer architecture and BERT isintroduced. This approach achieves learning of complex features by introducing semantic completeness and multi head semantic completeness mapping networks.

In terms of entity recognition, the introduced model achieved an average accuracy, recall, and F1 score of 90.7%, 91.6%, and 90.6%, respectively, on the CoNLL04 dataset. Compared to the MTL-NER and SPKTE models, it improved by 2.8%, 3.8%, and 2.9%. In the SciERC dataset, the average accuracy, recall, and F1 score of the introduced model were 69.8%, 70.4%, and 69.8%, respectively, which were 3.8%, 3.2%, and 3.1% higher than the MTL-NER and SPKTE models. Wang et al. introduced an entity recognition approach that combines RoFormer pre training model and Pointer-Net baseline model to address the low recognition accuracy of traditional named entity recognition approaches in agricultural disease texts [27]. Although this approach had a high recognition accuracy, its generalization ability was weak and only applicable to the agricultural field, while the approachintroduced in the study had strong generalization ability. Mojibian A et al. introduced a recognition approach using a commercial medical named entity recognition model combined with post-processing protocols to address the issue of identifying occasional pulmonary nodules from computed tomography reports. This approach utilized a generic medical named entity recognition model to annotate entities and their relationships, and filtered them through set inclusion/exclusion criteria to identify pulmonary nodules [28]. Although the above approach could achieve accurate identification of pulmonary nodules, its robustness was far lower than the model introduced in the study. In addition, the ablation experiment demonstrated that the introduction of semantic completeness significantly improved the recognition accuracy of the model, increasing from 82.3% to 90.7%. Among them, semantic completeness had the most significant impact on the performance of the model, increasing by 3.9%. Semantic completeness could effectively compensate for the shortcomings of traditional approaches in semantic representation and enhance the model's ability to perceive semantic differences in text. It can be seen that introducing semantic completeness and multi-head semantic completeness mapping networks could significantly improve the model's recognition ability for named entities, especially when dealing with complex text data, the model exhibited good robustness and generalization ability.

In terms of RE, the introducedRE model outperformed other approaches on both CoNLL04 and SciERC datasets. Specifically, in the CoNLL04 dataset, the average accuracy, recall, and F1 score of the introducedRE model were 75.3%, 73.4%, and 73.2%, respectively, which were 3.7%, 2.8%, and 3.8% higher than those of the DRPFF and BoBGSAL-Net models. In the SciERC dataset, the average accuracy, recall, and F1 score of the introducedRE model were 62.9%, 60.5%, and 61.4%, respectively, which were 5.0%, 2.4%, and 3.9% higher than the DRPFF and BoBGSAL-Net models. Han Z and Wang J introduced anREapproach based on knowledge enhanced graph inference network for ER extraction in industrial KG construction. This approach integrated domain knowledge and graph structure information, using graph inference networks to identify ERs and construct KGs, thereby improving the accuracy of extraction and the completeness of KGs [29]. Although the above approach had high accuracy in relation extraction, it was only applicable to the industrial field, while the approachintroduced in the study was applicable to different fields and had good generalization. Zhu F et al. introduced a threat intelligence REapproach to address the barriers to sharing and utilization caused by the lack of KGs in IoT threat intelligence. This approach first constructed a threat intelligence ontology based on existing security ontologies and knowledge bases, providing an organized pattern for merging threat intelligence text data. Secondly, a threat intelligence entity and relationship joint extraction model based on Token Pair Linking Plus was designed, and domain knowledge was introduced to enhance the model's learning of IoT security terminology semantics [30]. The above approacheswere also limited to a certain field, while the REapproachintroduced in the study did not have domain limitations. The introduced semantic RE model could more accurately identify the relationships between entities in sentences, especially when dealing with complex semantic relationships, where the model had significant advantages. The proposed model, with its ability to handle confusion between nested entities and types, was implemented in KG construction, medical/legal retrieval, question answering, and other scenarios: manual annotation costs were reduced by 40%, triplet accuracy was improved by 15-20%, retrieval relevance rate increased by 25%, and answer extraction increased by 30%. Lightweight Transformer could perform single sentence inference on CPU/GPU in less than 2 ms, which was 65% faster than BERT Large. It had a plug and play RESTful API and supported custom types.

In summary, the introducedapproach based on Transformer architecture for semantic association and automatic parsing of grammatical patterns in noun predicate sentences had excellent performance in entity recognition and RE tasks, and had good robustness and generalization ability, which could provide new and effective approaches for the field of NLP. Due to the dependence of the introduced model on large-scale annotated data for training, its generalizability in practical applications was limited. Meanwhile, due to the high computational complexity of the model, its training and inference time was relatively long. Therefore, in the future, the research will consider introducing semi supervised learning and weakly supervised learning approaches, and explore in depth the lightweight design of models to reduce their dependence on large-scale annotated data and improve their training and inference efficiency. In addition, due to the 512 token restriction, the cross segment accuracy of long documents decreased by 5-8%, and the generalization in small sample domains was weak. No need to change the structure across domains, adaptation costs were reduced by 70%, and millions of sentences were processed in 1.5 hours. Therefore, in the future, the research will consider introducing semi-supervised and lightweight design, and establishing document level semantic mechanisms to fill the gap in long text information.

5 Conclusion

To achieve automatic parsing of semantic associations and grammatical patterns in noun predicate sentences, a parsing approach based on Transformer architecture was introduced. This approach consisted of an entity recognition model and a semantic RE model. By introducing semantic completeness and a multi-head semantic completeness mapping network, it learned complex sentence features to improve the parsing accuracy of the model. The experimental findings demonstrated that the introduced entity recognition model significantly outperformed the MTL-NER and SPKTE models in terms of average accuracy, recall, and F1 score on the CoNLL04 and SciERC datasets in entity recognition tasks. In relation extraction tasks, the average accuracy, recall, and F1 score of this approachwere also superior to DRPFF and BoBGSAL-Net models. The introduced approach had significant advantages in handling complex semantic relationships and could effectively address the shortcomings of traditional approaches in nested entities and entity type confusion. Due to the strong dependence of the introduced model on large-scale annotated data, its generalizability was limited. Therefore, future research will explore semi-supervised learning and weakly supervised learning approaches to reduce dependence on annotated data and further enhance the practicality of the model.

References

[1] L. Yuan, Y. Cai, J. Wang and Q. Li. “Joint multimodal entity-relation extraction based on edge-enhanced graph alignment network and word-pair relation tagging,” in Proc. AAAI Conf. Artif. Intell., vol. 37, no. 9, pp. 11051-11059, June, 2023, DOI:10.1609/aaai. V37i9.26309.

[2] U. Zaratiana, N. Tomeh, P. Holat and T. Charnois. “An autoregressive text-to-graph framework for joint entity and relation extraction,” in Proc. AAAI Conf. Artif. Intell., vol. 38, no. 17, pp. 19477-19487, March, 2024, DOI:10.1609/aaai. V38i17.29919.

[3] J. Wang, K. Yue and L. Duan. “Models and techniques for domain relation extraction: a survey,” J. Data Sci. Intell. Syst., vol. 1, no. 2, pp. 65-82, May, 2023, DOI:10.47852/bonviewJDSIS3202973.

[4] C. Peng, X. Yang, Z. Yu, J. Bian, W. R. Hogan and Y. Wu. “Clinical concept and relation extraction using prompt-based machine reading comprehension,” J. Am. Med. Inform. Assoc., vol. 30, no. 9, pp. 1486-1493, August, 2023, DOI:10.1093/jamia/ocad107.

[5] H. Chen, P. Hong, W. Han, N. Majumder and S. Poria. “Dialogue relation extraction with document-level heterogeneous graph attention networks,” Cogn. Comput., vol. 15, no. 2, pp. 793-802, January, 2023, DOI:10.1007/s12559-023-10110-1.

[6] J. Suo, D. Han and H. Zhao. “A multiple head selection joint entity-relation extraction model,” J. Intell. Fuzzy Syst.: Appl. Eng. Technol., vol. 45, no. 4, pp. 5647-5657, January, 2023, DOI:10.3233/JIFS-231766.

[7] J. Han and K. Jia. “Entity relation joint extraction approach for manufacturing industry knowledge data based on improved bert algorithm,” Cluster Comput., vol. 27, no. 6, pp. 7941-7954, April, 2024, DOI:10.1007/s10586-024-04386-7.

[8] T. Wang, W. Yang, T. Wu, C. Yang, J. Liang and H. Wang et al. “Joint entity and relation extraction with fusion of multi-feature semantics,” J. Intell. Inf. Syst., vol. 63, no. 1, pp. 21-42, August, 2025, DOI:10.1007/s10844-024-00871-y.

[9] A. Khan, Z. Rauf, A. Sohail, A. R. Khan, H. Asif, A. Asif and U. Farooq. “A survey of the vision transformers and their CNN-transformer based variants,” Artif. Intell. Rev., vol. 56, no. 3, pp. 2917-2970, October, 2023, DOI:10.1007/s10462-023-10595-0.

[10] K. Feng, Y. Ma, B. Wang, C. Qi, H. Chen, Q. Chen and Z. Wang. “Dit4edit: Diffusion transformer for image editing,” in Proc. AAAI Conf. Artif. Intell., vol. 39, no. 3, pp. 2969-2977, April, 2025, DOI:10.1609/aaai. V39i3.32304.

[11] C. Yeh, Y. Chen, A. Wu, C. Chen, F. Viégas and M. Wattenberg. “Attentionviz: A global view of transformer attention,” IEEE Trans. Vis. Comput. Graph., vol. 30, no. 1, pp. 262-272, October, 2023, DOI:10.1109/TVCG.2023.3327163.

[12] A. He, K. Wang, T. Li, C. Du, S. Xia and H. Fu. “H2former: An efficient hierarchical hybrid transformer for medical image segmentation,” IEEE Trans. Med. Imaging, vol. 42, no. 9, pp. 2763-2775, April, 2023, DOI:10.1109/TMI.2023.3264513.

[13] S. Pal, A. Roy, P. Shivakumara and U. Pal. “Adapting a Swin Transformer for License Plate Number and Text Detection in Drone Images,” Artif. Intell. Appl., vol. 1, no. 3, pp. 145-154, April， 2023, DOI:10.47852/bonviewAIA3202549.

[14] X. Li, S. Li, S. Song, J. Yang, J. Ma and J. Yu. “Pmet: Precise model editing in a transformer,” in Proc. AAAI Conf. Artif. Intell., vol. 38, no. 17, pp. 18564-18572, March, 2024, DOI:10.1609/aaai. V38i17.29818.

[15] H. Dalla-Torre, L. Gonzalez, J. Mendoza-Revilla, N. Lopez Carranza, A. H. Grzywaczewski, F. Oteri et al. “Nucleotide Transformer: building and evaluating robust foundation models for human genomics,” Nat. approaches, vol. 22, no. 2, pp. 287-297, November, 2025, DOI:10.1038/s41592-024-02523-z.

[16] E. Arkin, N. Yadikar, X. Xu, A. Aysa, and K. Ubul. “A survey: object detection approaches from CNN to transformer,” Multimedia Tools Appl., vol. 82, no. 14, pp. 21353-21383, October, 2023, DOI: 10.1007/s11042-022-13801-3.

[17] Q. Zhang, J. Zhang, Y. Xu, and D. Tao. “Vision transformer with quadrangle attention,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 46, no. 5, pp. 3608-3624, January, 2024, DOI: 10.1109/TPAMI.2023.3347693.

[18] S. Gao, Z. Chen, G. Chen, W. Wang, and T. Lu. “Avsegformer: Audio-visual segmentation with transformer,” in Proc. AAAI Conf. Artif. Intell., vol. 38, no. 11, pp. 12155-12163, March, 2024, DOI: 10.1609/aaai. V38i11.29104.

[19] W. Wang, W. Chen, Q. Qiu, L. Chen, B. Wu, B. Lin, ... and W. Liu. “Crossformer++: A versatile vision transformer hinging on cross-scale attention,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 46, no. 5, pp. 3123-3136, December, 2023, DOI: 10.1109/TPAMI.2023.3341806.

[20] H. Y. Zhou, Y. Yu, C. Wang, S. Zhang, Y. Gao, J. Pan, ... and W. Li. “A transformer-based representation-learning model with unified processing of multimodal input for clinical diagnostics,” Nat. Biomed. Eng., vol. 7, no. 6, pp. 743-755, June, 2023, DOI: 10.1038/s41551-023-01045-x.

[21] Z. Xu, D. Wu, C. Yu, X. Chu, N. Sang, and C. Gao. “Sctnet: Single-branch cnn with transformer semantic information for real-time segmentation,” in Proc. AAAI Conf. Artif. Intell., vol. 38, no. 6, pp. 6378-6386, March, 2024, DOI: 10.1609/aaai. V38i6.28457.

[22] M. Li, T. Lv, J. Chen, L. Cui, Y. Lu, D. Florencio, ... and F. Wei. “Trocr: Transformer-based optical character recognition with pre-trained models,” in Proc. AAAI Conf. Artif. Intell., vol. 37, no. 11, pp. 13094-13102, June, 2023, DOI: 10.1609/aaai. V37i11.26538.

[23] W. Liu, M. Yin, J. Zhang, and L. Cui. “A joint entity relation extraction model based on relation semantic template automatically constructed,” Comput. Mater. Continua, vol. 78, no. 1, pp. 975-997, January, 2024, DOI: 10.32604/cmc.2023.046475.

[24] X. Cao and Q. Shao. “Joint entity relation extraction based on lstm via attention mechanism,” Arabian J. Sci. Eng., vol. 49, no. 3, pp. 4353-4363, November, 2024, DOI: 10.1007/s13369-023-08306-6.

[25] D. Zeng, L. Xu, and D. L. Jiang. “Sequence tagging with a rethinking structure for joint entity and relation extraction,” Int. J. Mach. Learn. Cybern., vol. 15, no. 2, pp. 519-531, August, 2024, DOI: 10.1007/s13042-023-01923-5.

[26] D. Sui, X. Zeng, Y. Chen, K. Liu, and J. Zhao. “Joint entity and relation extraction with set prediction networks,” IEEE Trans. Neural Networks Learn. Syst., vol. 35, no. 9, pp. 12784-12795, April, 2024, DOI: 10.1109/TNNLS.2023.3264735.

[27] T. Wang, C. Wang, J. Li, H. Zhu, Y. Miao, and H. Wu. “Agricultural Disease Named Entity Recognition with Pointer Network Based on RoFormer Pre-trained Model,” Smart Agriculture, vol. 6, no. 2, pp. 85-94, March, 2024, DOI: 10.12133/j. Smartag. SA202311021.

[28] A. Mojibian, J. Jaskolka, G. Ching, B. Lee, R. Myers, C. Devine, et al. "The efficacy of a named entity recognition ai model for identifying incidental pulmonary nodules in ct reports," Can. Assoc. Radiol. J., vol. 76, no. 1, pp. 68–75, February, 2025, DOI: 10.1177/08465371241266785.
[29] Z. Han and J. Wang. "Knowledge enhanced graph inference network-based entity-relation extraction and knowledge graph construction for industrial domain," Front. Eng. Manag., vol. 11, no. 1, pp. 143-158, February, 2024, DOI: 10.1007/s42524-023-0273-1.
[30] F. Zhu, Z. Cheng, P. Li, and H. Xu. "Itirel: joint entity and relation extraction for internet of things threat intelligence," IEEE Internet Things J., vol. 11, no. 11, pp. 20867-20878, March, 2024, DOI: 10.1109/JIOT.2024.3373799.