ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Research on the Application of Deep Learning in Biology Knowledge Graph Construction

Yao Song[1,*]

[1] College of Life Sciences, Capital Normal University, Beijing 100048, Beijing, China

SUMMARY: *In order to support the construction of knowledge graphs in heterogeneous biological texts and databases, a deep learning framework combining entity recognition, standardized mapping, relation extraction and graph fusion was proposed. The experimental data are collected from PubMed abstracts, biological terminological databases and structured relational databases, and 182000 entity mentions, 396000 candidate relations and 128000 triplet candidates are obtained after cleaning. In the first stage, the entity coding network based on BioBERT extracted the context semantic features and stably completed the term standardization mapping. In the second stage, the relation classification module identified the semantic types between entities and generated candidate triples. In the third stage, the fusion building module performs node merging, relation alignment and structure writing. Experimental results show that the framework achieves 93.2% entity recognition accuracy, 91.3% standardized accuracy, and 89.8% macro average F1 value of relation classification. The node repetition rate is controlled to 4.1%, the effective write success rate is 88.4%, and the single-hop query response time is 84 ms.*

KEYWORDS: *Deep learning; Biological knowledge graph; Entity recognition; Relation extraction*

1 Introduction

Biological knowledge graph organizes the semantic relationships among genes, proteins, diseases, pathways and drugs in a graph structure, and can transform the knowledge scattered in databases, literature and terminology into a searchable, computable and reasonable unified representation. With the continuous expansion of computing tasks for precision medicine, biological information retrieval and mechanism discovery, knowledge graph is no longer just a knowledge management tool, but an important technical carrier connecting natural language processing, graph representation learning and heterogeneous data fusion. At present, the biological data that can be used for graph construction contain text descriptions, standard terms, structured database tables and cross-database identifiers. The data have obvious differences in data form, complex naming methods, and fine semantic boundaries, so it is difficult to support high-density knowledge update by relying on manual sorting alone. The introduction of deep learning into entity recognition, term normalization, relation extraction and graph fusion can gradually transform discrete knowledge into stable graph structure objects, so as to make the construction of biological knowledge graph have stronger automation and scale capabilities. At the computer implementation level, entity boundary detection, namesake abbreviation disambiguation, cross-database identity mapping and

*songyao0126@126.com
https://doi.org/10.65102/is2026297
Published: 30 April, 2026

relationship type determination jointly determine the quality of graph nodes and edges, and the quality of graph directly affects the downstream retrieval efficiency, similar entity matching results and representation propagation effect on graph neural networks.

The existing research has provided a solid technical foundation for this direction. Sousa et al. embedded the knowledge graph recommendation mechanism into the biomedical relation extraction process, and strengthened the utilization of semantic associations between entities in the relation recognition stage [1]. Alshahrani et al. jointly model biomedical knowledge graph and text representation, and use unified computing framework to deal with drug-target and drug-indication prediction, showing the feasibility of collaborative learning of structural knowledge and contextual semantics [2]. Ren et al. constructed a deep neural network combining local features and global features for drug-drug interaction prediction, indicating that the graph representation can provide a stable semantic basis for subsequent discrimination tasks [3]. Yang et al. further identified interpretable paths from multi-layer biological networks, making graph structure learning move from static association representation to mechanism characterization [4].

At the level of data organization and engineering implementation, Gao et al. proposed KG-Predict framework to integrate multi-source biomedical entities and relationships into a unified graph structure and complete drug relocation calculation [5]. Erdengasileng et al. used pre-trained models, data augmentation, and ensemble learning to carry out biomedical information extraction, and provided a transferable text-side scheme for high-quality entity and relation generation [6]. Bang et al. focused on the learning of multi-layer biomedical knowledge graph, and improved the drug relocation representation by extending the culpability association propagation path to enhance the relationship propagation ability in heterogeneous graph [7]. Chandak et al. constructed PrimeKG to incorporate diseases, drugs, genes and phenotypes into a unified framework, providing a clear data organization paradigm for large-scale biological knowledge graph construction [8]. These studies have promoted biomedical knowledge computing from the perspectives of relation extraction, graph learning, graph structure inference and knowledge organization, respectively. However, there is still room for further refinement of the deep learning methods that focus on the complete construction link of "entity recognition, relation generation and graph fusion". From the perspective of technical journal writing, the introduction should not only explain the application scenario, but also explain the method entry clearly, that is, how entity recognition and normalization on the text side are connected with relation modeling and fusion writing on the graph side. Only when this computational link is spelled out will the model design, experimental metrics, and system results in subsequent sections have a unified interpretation basis. This is the reason why this paper adopts a phased design when organizing the methodology. The computational goal is more explicit. Clearer layers

Based on this, this paper focuses on the construction process of biological knowledge graph itself, and organizes the method process around three core objects: entities, relationships and graph structures. In this paper, we propose a deep learning construction method for heterogeneous biological data. The method consists of three consecutive stages. In the first stage, the entity recognition is completed through the biological entity semantic coding network, and the normalized mapping is realized by combining the standard vocabulary. In the second stage, the deep relation extraction model is used to identify the interaction types between entities and form structured triples. The third stage uses the fusion strategy to write the structural information in the literature, database and terminology resources into the unified graph. The remainder of this paper is organized as follows: Section II presents related research, Section III describes the proposed method, Section IV discusses experimental results, and Section V gives conclusions and future work.

2 Related Research

2.1 Research on biological entity recognition and Terminology Normalization

Biological entity recognition and terminology normalization constitute the initial link of knowledge graph construction. Whether the nodes in the graph remain stable, consistent and computable depends on whether the gene, protein, disease, drug and phenotype names in the text can be accurately identified and further corresponded to a unified terminology system and database identifier. Different from general domain named entity recognition, entities in biological texts are often accompanied by a large number of abbreviations, aliases, nested phrases and cross-base coding differences. The same term may also correspond to different objects in different contexts. Therefore, related research has gradually shifted from rule matching to the technical path of deep semantic representation, context discrimination and standardized mapping.

Morris et al. carried out research on SPOKE map, constructed a large-scale precision medicine knowledge framework covering diseases, drugs, genes and phenotypes, and proposed the implementation method of organizing nodes and identity mapping based on multi-ontology [9]. In this study, entity normalization is directly incorporated into the underlying graph organization process, which provides a unified foundation for node naming, semantic merging, and cross-source connection. Murali et al. systematically reviewed the construction of medical knowledge graph driven by electronic medical records, and proposed an analysis framework from entity representation, term alignment to collaborative organization of knowledge completion process [10]. This study emphasizes that entity recognition is not an independent text task, and the standardization results will directly affect the quality of graph node generation, relationship linking, and subsequent knowledge completion. Lyu et al. studied the construction of causal knowledge graph in clinical decision support for diabetic nephropathy, and proposed a graph representation method combining clinical concept organization and causal relationship expression [11]. In this process, the determination of entity boundaries and the unification of concept names are not only related to node generation, but also to semantic accuracy when expressing causal paths. Lin et al. studied a graph multi-modal learning method for disease relation extraction, and proposed a relation recognition framework that jointly encoded graph structure information and linguistic information [12]. Although this research focuses on relation extraction, the premise is that disease entities can be stably identified and represented, which also indicates that entity normalization has formed a close connection with the subsequent graph learning process.

Based on the existing research, it can be seen that biological entity recognition and term normalization are shifting from dictionary dependence and local rule constraints to a computational mode involving pre-trained representation, structural constraints and cross-source mapping. The normalized entity results are no longer just text extraction output, but directly become the unified entry for knowledge graph node construction, relationship organization, and graph learning training. From the perspective of computer implementation, the key to this direction is not only recognizing the entity name itself, but also putting the character-level morphological features, contextual semantic features and standard terminology library constraints into the same representation space, so that the model can take into account both semantic discrimination and identity normalization in the recognition stage. The generated node input is more suitable for the subsequent relation extraction and graph fusion process, and the whole construction link will be more coherent. Therefore, the focus of research in this field has shifted from pure entity recognition to the overall computational goal

of collaborative promotion of recognition accuracy, standardization consistency and node in-graph stability, which forms a more complete method chain and enhances the expansion ability of the model in graph construction scenarios.

2.2 Research on relation extraction driven by Deep representation learning

This category of research is centered around computational representations of relational semantics. The basic idea is to map entity pairs, context sentences, dependency structures and cross-sentence evidence into trainable vectors, and then use the deep model to determine the relationship type. Compared with rule-based or shallow feature-based relation extraction, deep representation learning can absorb lexical semantic, syntactic dependency and graph structure information at the same time, so it is more suitable for dealing with the situations of dense relation expression, scattered evidence and large term span in biological texts. In the process of knowledge graph construction, relation extraction is not only to find out whether there is a connection between two entities from a sentence, but also to generate structured relations with clear semantic labels and direction constraints, which provides stable input for subsequent triple organization and graph fusion.

Wang et al. studied the pre-training framework of biomedical knowledge graph and proposed PT-KGNN method, which used graph neural network for knowledge graph pre-training representation learning [13]. This method models the relation semantics by node context and graph structure together, so that the relation representation does not rely on a single text fragment, but can absorb the adjacency information and topological characteristics in the graph. He et al. studied prompt tuning methods in biomedical relation extraction and proposed to introduce task-related prompts on pre-trained language models to enhance the ability of relation recognition [14]. This approach reduces the cost of large-scale parameter updates while enabling the model to more centrally exploit discriminative signals in the entity context. Li et al. studied an ensemble pre-trained language model for knowledge extraction from biomedical literature, and proposed to improve the consistency and stability of relation extraction through multi-model collaborative output [15]. This study shows that the representations of relational semantics in different encoders are complementary, and they are more suitable for knowledge extraction tasks in complex literature scenarios after integration. Yuan et al. studied document-level biomedical relation extraction and proposed a method combining hierarchical treemap and relation segmentation module [16]. This method integrates the intra-sentence information and cross-sentence information into the dendrogram structure, so that the model can capture the potential semantic links between entities in the range of longer texts.

From the existing research, the relation extraction driven by deep representation learning has gradually shifted from single sentence discrimination to a computing mode involving graph structure perception, prompt adaptation and document-level inference. At the level of computer implementation, relation extraction models usually need to process entity location, context window, dependency path, and paragraphlevel evidence propagation synchronously to avoid misclassifying surface co-occurrences as real biological relationships. For tasks with fine semantic boundaries such as protein interactions, gene regulation and drug action mechanisms, the design of the representation layer often directly determines whether the relation labels can be accurately inserted into the graph. Therefore, related research no longer regards relation extraction as an independent classification step, but as an important computational step connecting text understanding, structure representation and graph construction. The higher the stability of the model output, the easier it is to control the error propagation in the triplet organization, graph fusion, and subsequent graph learning training,

so that the relation extraction can more naturally connect the knowledge graph construction process, and enhance the stability of the whole computing link in engineering deployment.

2.3 Research on heterogeneous biological data Fusion and knowledge graph Construction

The core of research on heterogeneous biological data fusion and knowledge graph construction is no longer local knowledge extraction from a single text, but how to organize literature sentences, database entries, terminology systems and graph structural constraints into a unified computational object. For biological scenarios, the sources of entities are scattered, the evidence levels of relationships are different, and the identification system is not completely consistent. Therefore, the construction of graph usually requires multi-source alignment at the presentation layer, relationship organization at the structure layer, and unified writing of nodes and edges at the storage layer. The introduction of deep learning methods has gradually shifted this process from rule concatenation to the technical path of collaborative promotion of representation learning, structure fusion and automatic composition.

Zhang et al. studied location-enhanced syntactic knowledge in biomedical relation extraction, and proposed a method to jointly introduce positional information and syntactic structure into relation representation [17]. Although this study focuses on relation recognition, its results provide a more stable semantic boundary for heterogeneous textual evidence to enter the unified graph structure. Jia et al. studied a biomedical relation extraction method based on ensemble learning and attention mechanism, and proposed to improve the consistency of relation recognition through multi-model collaboration [18]. This idea enables relation signals from different corpora and different encoders to maintain good complementarity in the fusion stage. Schafer et al. studied the process of automatically constructing knowledge graphs from biomedical literature and proposed the BioKGrapher system to evaluate the initial effect of automatic composition [19]. This work shows that after structured extraction, standardized mapping and relational organization of literature knowledge, a graph framework with computational value can be formed. Ivanisenko et al. studied the method of extracting knowledge from scientific literature by using structured ontology model, graph neural network and large language model, and proposed the implementation path of integrating multi-class models to complete knowledge extraction and graph generation [20]. This approach connects text understanding, graph representation and knowledge organization, making graph construction closer to the end-to-end computing process.

The methodological characteristics of related studies are shown in Table 1.

Table 1: Comparison of heterogeneous biological data fusion and knowledge graph construction studies

References	Core Object	Main Method
[17]	Relation Evidence Fusion	Position-Enhanced Syntactic Representation
[18]	Multi-Model Relation Integration	Ensemble Learning + Attention
[19]	Automatic Literature Graph Construction	Normalization Mapping and System Evaluation
[20]	Multi-Source Knowledge Generation	Ontology + GNN + LLM Fusion

It can be seen from the existing research that heterogeneous biological data fusion and knowledge graph construction have shifted from sequential processing to a computing mode

that is jointly promoted by multi-source representation collaboration, structural constraint participation and automatic composition. The stability of the graph construction results depends not only on the performance of a single extraction module, but also on the fusion quality of cross-source data in a unified semantic space. At the computer implementation level, only when text representation, database identification, graph structure constraints and fusion writing strategies are incorporated into the same computing link, node duplication, relationship splitting and semantic drift can be more easily controlled, and subsequent graph learning training can also obtain a more stable input basis. The resulting construction process is more complete in structure organization and more stable in engineering implementation.

3 Proposed method

This section revolves around the overall computational flow of biological knowledge graph construction. Instead of separating entity recognition, relation extraction and graph writing into separate processing steps, we integrate semantic representation, relation discrimination and structure fusion into a unified deep learning framework. The system input consists of biomedical literature, a glossary of terms, and structured database records. After cleaning, clauses, entity positioning and matching standard identifiers, the original data is transformed into entity representations that contain both contextual semantic information and library table constraint information. On this basis, the relation encoding module combines the context attention mechanism, syntactic dependency information and graph structure features to identify the interaction types between entities and generate relation triples that can be directly inserted into the graph. Then, the fusion building module aliges, deduplicates and writes consistently the nodes and edges from different sources, and finally forms a queryable and extensible biological knowledge graph. The overall process is shown in Fig. 1. The graph presents data input layer, entity recognition layer, relation extraction layer and graph fusion layer in turn, and the results of each layer are continuously passed in the same computing link, so as to ensure the consistency of semantic expression and structural organization of the construction results. In the concrete implementation, the entity recognition layer uses the context coding and term standardization coordination mechanism, so that synonyms, abbreviations and cross-database aliases can be stably mapped to the unified node. The relation extraction layer reduces the generation of repeated edges and weak edges by constraining the relation direction and semantic label through multi-granularity representation. The graph results generated through this link can not only support subsequent graph learning training, but also facilitate computing tasks such as retrieval, statistical analysis and biological relationship discovery.

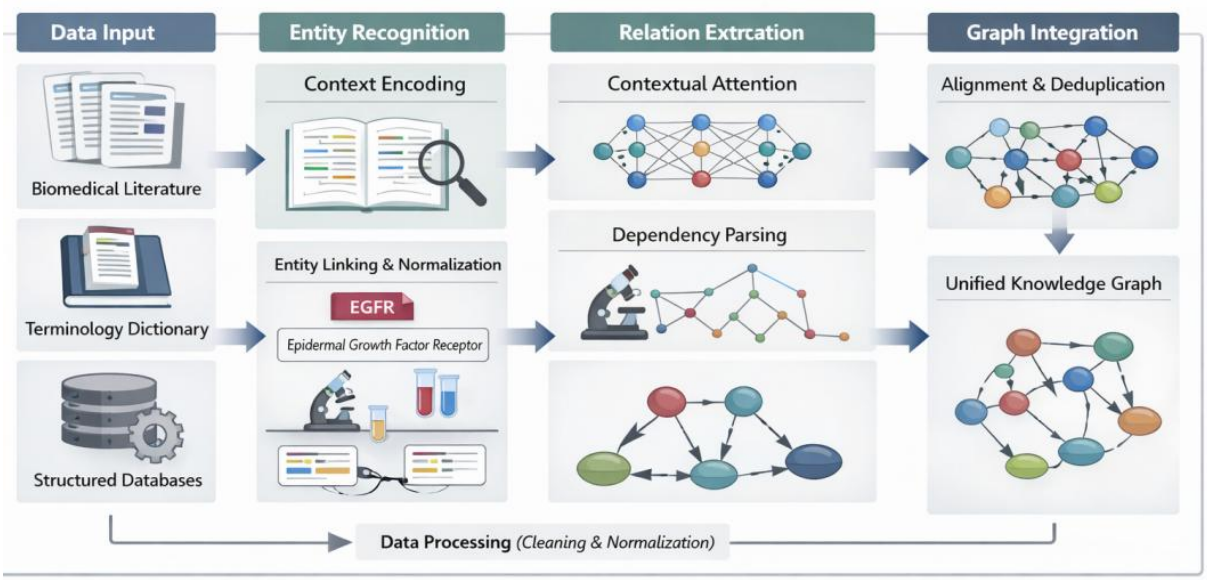

Figure 1: Deep learning driven biological knowledge graph construction process

3.1 Entity Recognition mechanism based on deep learning

3.1.1 Biological entity semantic encoding network

The role of biological entity semantic encoding network is not to simply encode the word form, but to transform the gene, protein, disease and drug names in the literature into a unified semantic representation that can participate in the subsequent standardized mapping and relationship calculation. Considering the characteristics of dense abbreviations, coexistence of aliases and frequent occurrence of compound phrases in biomedical texts, the network input is composed of word segment semantics, relative position, entity type and term vocabulary matching results. The relevant input composition is shown in Table II.

Table 2: Input composition of the semantic encoding network for biological entities

Input Item	Function	Representation Method
Token Sequence	Preserve contextual semantics	Index Embedding
Position Sequence	Mark entity boundaries	Relative Position Encoding
Type Sequence	Distinguish entity categories	Type Embedding
Lexicon Matching Sequence	Introduce terminology priors	Gated Vector

To unify information from different sources, the network first constructs an initial semantic representation, which is calculated as follows.

$$S_i = LN(X_i + P_i + T_i + \lambda_i K_i) \tag{1}$$

Here, S_i represents the initial encoding of the i word slice, X_i represents the semantic embedding of the word slice, P_i represents the position encoding, T_i represents the entity type embedding, K_i represents the term vocabulary matching vector, λ_i represents the matching strength coefficient, and LN represents the layer normalization operation. Equation (1) is used to map text semantic information and biological term constraints into the same representation space, so that the encoding results have clear biological entity pointing before entering the context modeling stage.

In the context modeling phase, the network uses a location-aware attention mechanism to aggregate effective semantics near entity boundaries, which is calculated as follows.

$$\alpha_{ij} = \frac{\exp\left((Q_i \cdot K_j)/\sqrt{d} + \beta B_{ij}\right)}{\sum_{t=1}^{n} \exp\left((Q_i \cdot K_t)/\sqrt{d} + \beta B_{it}\right)} \tag{2}$$

$$C_i = \sum_{j=1}^{n} \alpha_{ij} V_j \tag{3}$$

Here, α_{ij} represents the attention weight from the i position to the j position; Q_i, K_j and V_j represent the query, key and value vectors respectively; d represents the vector dimension; B_{ij} represents the relative distance bias; β represents the distance regulation coefficient; and C_i represents the context aggregation result. Equations (2) and (3) are jointly used to enforce local dependencies and long-distance semantic connections near entity boundaries, enabling more stable contextual expression of compound phrases, abbreviated forms, and cross-word slice entities.

In the entity decision stage, the network further maps the context results together with the candidate segment features into label probabilities, which are calculated as follows.

$$y_i = \mathrm{Softmax}(U \tanh(MC_i + \gamma G_i + b_1) + b_2) \tag{4}$$

Here, G_i represents the boundary features of the candidate segments, M and U represent the linear transformation parameters, γ represents the boundary feature adjustment coefficient, b_1 and b_2 represent the bias term, and y_i represents the probability distribution of each class of entity labels. Equation (4) is used to complete the final discrimination of entity categories such as genes, proteins, diseases and drugs. After this encoding link, abbreviations, aliases and compound biological phrases can obtain more stable semantic aggregation in a unified representation, which provides consistent input for subsequent standardized mapping and node writing.

After the above encoding process, the entity fragments in the original biomedical text are transformed into a unified representation with both contextual semantics, boundary features and term constraints. This representation not only improves the stability of entity recognition results, but also provides a reliable semantic basis for subsequent standardized mapping and graph node construction.

3.1.2 Biological entity standardized mapping method

The function of biological entity standardized mapping is not to simply replace the identification results, but to stably correspond the gene, protein, disease and drug names in the text to the standard nodes in the unified terminology base. Biomedical corpora also have abbreviation forms, alias forms, old name forms and cross-database coding differences. The same entity may also present different representation granularity in different documents. Therefore, if we only rely on exact string matching, it is easy to cause problems such as synonymous names being split and written, different entities being wrongly merged, and node identification being unable to be unified. In order to make the subsequent relation extraction, triple organization and graph writing consistent, this paper introduces four consecutive steps of candidate retrieval, semantic alignment, context correction and threshold determination on the basis of the entity semantic encoding results to form a standardized mapping link for knowledge graph construction.

In the candidate retrieval phase, the system first generates the candidate node set according to the entity surface form, the term lexicon index and the cross-database alias table. Let the identified entity fragment be denoted as a_i, and the j candidate node in the terminology base be denoted as b_j. First, calculate the base matching score between them:

$$r_{ij} = \eta \cdot \cos(a_i, b_j) + (1 - \eta) \cdot \frac{a_i^{\top} D b_j}{\|a_i\| \|D b_j\|} \tag{5}$$

Here, r_{ij} represents the basic similarity between the entity fragment and the candidate node, η represents the balance coefficient of the two types of similarity, D represents the learnable mapping matrix, and $\cos(\cdot)$ represents the cosine similarity. Equation (5) simultaneously describes the semantic proximity relationship in the original coding space and the deep semantic consistency in the mapping space, so that the candidate ranking is no longer dependent on the single word plane coincidence, but is based on the joint participation of semantic representation and mapping representation.

After the basic matching score is obtained, the system does not directly select the maximum value node, but further introduces context correction to distinguish the entities with the same name and the entities with similar hyponymy concepts. For different references of the same name in different sentence segments, the network needs to use context semantics to determine which kind of standard node the current entity corresponds to. It is calculated as follows:

$$c_{ij} = \sigma\left(u^{\top}[a_i; b_j; q_i] + b\right) \tag{6}$$

where c_{ij} represents the context consistency score, q_i represents the context vector of the sentence segment in which the entity is located, $[a_i; b_j; q_i]$ represents vector concatenation, u and b are learnable parameters, and σ represents the Sigmoid function. Equation (6) is used to measure the degree of consistency between the semantics of the candidate node and the context of the current sentence segment. If the entity segment itself is highly close to the candidate node in surface form, but the function words, action objects or biological processes in the sentence segment are inconsistent, the score of this item will be depressed, thus reducing the probability of wrong mapping.

On this basis, the system jointly normalizes the base similarity and context consistency to obtain the final standardized mapping probability:

$$p_{ij} = \frac{\exp(r_{ij} + \mu c_{ij})}{\sum_{k=1}^{m} \exp(r_{ik} + \mu c_{ik})} \tag{7}$$

Here, p_{ij} represents the probability that an entity fragment maps to candidate node j, μ represents the regulation coefficient of the context correction term, and m represents the number of candidate nodes. Equation (7) is used to complete the final selection among multiple candidate nodes and output a standardized result with ranking information. Compared with the simple maximum matching strategy, this joint normalization method can maintain a more stable decision boundary in the case of dense aliases, repeated abbreviations and overlapping nodes across databases.

When the maximum probability was higher than the preset threshold, the system directly mapped the entity to the corresponding canonical node. When the probability of multiple candidate nodes is close, the system retains the candidate ranking results and passes them to the subsequent relation extraction module to continue verification. The reason for this design

is that the standardization of partial entities cannot be completed only by local names, but still needs to be further confirmed by combining relationship direction, action object and contextual biological process. After this mapping process, abbreviated entities, alias entities and cross-database entities can enter the unified node domain, the phenomena of repeated writing, semantic splitting and node drift in the construction of the graph are significantly reduced, and the node writing process can maintain more stable consistency and traceability. This standardized mapping method provides a reliable node foundation for subsequent relation extraction and knowledge graph fusion construction, so that the entity layer expression and the graph layer organization are continuously connected.

3.2 Relation Extraction and Triplet Generation Based on Deep learning

The goal of this step is to further determine the semantic relationship between entities after completing entity recognition and standardized mapping, and generate structured triples that can be directly added to the graph. The input consists of normalized entity nodes, sentence segment context representation and dependency structure features. Different from the entity encoding stage, this step no longer emphasizes the semantic expression of a single entity, but focuses on entity pair combination, relationship direction discrimination and triple confidence estimation. The system first constructs a direction-sensitive relation representation for any candidate entity pair, which is calculated as follows.

$$z_{ij} = \tanh\left(A_1 e_i + A_2 e_j + A_3 c_{ij} + b_1\right) \tag{8}$$

Here, z_{ij} represents the relation representation vector from entity i to entity j, e_i and e_j represent the normalized coding results of the two entities, c_{ij} represents the context semantic vector between the two, A_1, A_2 and A_3 are linear transformation parameters, b_1 is the bias term. Equation (8) is used to jointly compress the head entity, tail entity and local context into the same relation representation space, and explicitly preserve the direction information.

After obtaining the relation representation, the system further uses the relation matrix to score different semantic types, so as to distinguish between activation, inhibition, action, dependence and other relations. It is calculated as follows:

$$g_{ij}^{(r)} = z_{ij}^{\top} R_r z_{ij} + \rho_r^{\top} d_{ij} \tag{9}$$

Here, $g_{ij}^{(r)}$ represents the original score of entity pair (i, j) on relation type r, R_r represents the bilinear discriminant matrix corresponding to relation r, d_{ij} represents the structural feature vector composed of dependency path and syntactic distance, and ρ_r represents the structural feature parameters of relation r. Equation (9) is used to jointly incorporate semantic representation and syntactic structure into relation determination, so that relation scoring not only depends on context co-occurrence, but also considers directionality and structure.

The final probability of relation labels is given by the multi-relation normalization procedure, which is calculated as follows.

$$p_{ij}^{(r)} = \frac{\exp\left(g_{ij}^{(r)}\right)}{\sum_{s=1}^{K} \exp\left(g_{ij}^{(s)}\right)} \tag{10}$$

Here, $p_{ij}^{(r)}$ represents the probability that the entity pair (i, j) is judged to be of relation type r, and K represents the total number of candidate relation categories. Equation (10) is used to complete the normalized selection between multi-class relation labels and output the most probable relation type.

The overall process of relation extraction and triple generation is shown in Fig. 2. In the figure, four consecutive steps are given in turn: candidate entity pair construction, relation representation generation, relation label determination and triple screening. The first two links are responsible for transforming the node-level representation into the relation-level representation, and the last two links are responsible for completing the label determination and structure output, so that the relationship level results can be directly connected with the standardized node results of the previous article.

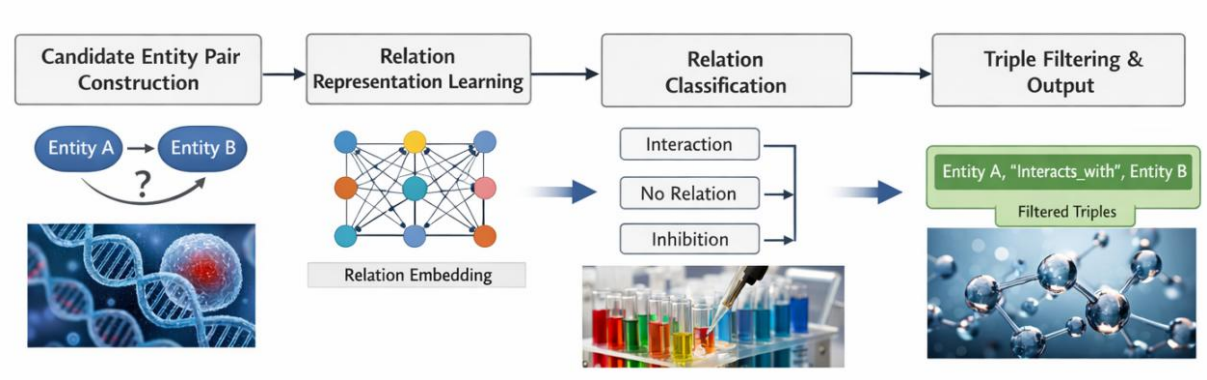

Figure 2: Relation extraction and triple generation process

After the relation labels are determined, the system further performs confidence estimation on the candidate triples to screen out weak associations and unstable relations. It is calculated as follows:

$$q_{ijr} = \sigma\left(\theta_1 p_{ij}^{(r)} + \theta_2 m_{ij} + \theta_3 n_{ij} + b_2\right) \tag{11}$$

Here, q_{ijr} represents the candidate triple confidence consisting of head entity i, relation r, and tail entity j, m_{ij} represents entity co-occurrence strength, n_{ij} represents cross-sentence evidence consistency, θ_1, θ_2, and θ_3 are fusion coefficients, $b_2$is bias term, and σ represents Sigmoid function. Equation (11) is used to integrate relation probability, co-occurrence evidence and context consistency, and only high-confidence triples that can support graph writing are retained.

After the above calculation, the system can transform the potential links between normalized entity nodes into structured triples with directions, labels and confidence. Such output no longer stays at the sentence-level relation recognition level, but can directly enter the subsequent graph fusion and structure writing process. Because the relationship type determination, direction constraint and confidence screening are completed in the same link, the generated triples have good stability in semantic expression and structural organization, and also provide a continuous and reliable relationship basis for subsequent knowledge graph construction.

3.3 Knowledge Graph Fusion Construction Based on Deep Learning

After entity recognition, standardized mapping, relation extraction and triple generation, the

system needs to write nodes and relations from literature, term base and structured database into a unified graph. At this time, the focus of processing is no longer the determination of a single entity or a single relationship, but how to complete the node merging, relationship alignment, conflict screening and structure writing in the existing graph structure. To this end, we construct a fusion module on the initial graph to jointly determine the candidate node pairs and candidate relation pairs, so that the writing process can maintain both semantic consistency and structural stability.

For any two candidate nodes a and b, the system first calculates the node fusion score, which is used to determine whether they should be merged into the same graph node. It is calculated as follows:

$$m_{ab} = \sigma(x_a^{\top} H x_b + \phi^{\top} |x_a - x_b| + b_1) \tag{12}$$

Here, m_{ab} represents the fusion score of node a and node b, x_a and x_b represent the semantic representation vectors of the two nodes, H represents the billinear mapping matrix, ϕ represents the difference term parameter, $|x_a - x_b|$ represents the absolute representation of the node vector difference, b_1 is the bias term, and σ represents the Sigmoid function. Equation (12) is used to simultaneously characterize the proximity of two nodes in terms of semantic similarity and representation difference, thus avoiding the repeated writing of synonymous nodes.

After the node fusion decision, the system also needs to make the relationship structure consistent. For candidate relations r between the same pair of nodes, the system further computes the relation alignment score, which is calculated as follows.

$$o_{ab}^{(r)} = \tanh\left(\psi_1 g_{ab}^{(r)} + \psi_2 c_{ab}^{(r)} + \psi_3 l_{ab}^{(r)}\right) \tag{13}$$

Here, $o_{ab}^{(r)}$ represents the alignment score of node pair (a, b) on relation type r, $g_{ab}^{(r)}$ represents the relation extraction score, $c_{ab}^{(r)}$ represents the cross-source evidence consistency, $l_{ab}^{(r)}$ represents the matching degree between relation label and terminology base definition, and ψ_1, ψ_2 and ψ_3 are the fusion weight parameters. Equation (13) is used to jointly compress textual evidence, structural evidence, and term constraints into a unified relation alignment space.

The overall processing flow of knowledge graph fusion construction is shown in Fig. 3. The graph gives four consecutive steps in turn: candidate node merging, relationship alignment, conflict screening and structure writing. The first two steps are responsible for the unification of semantic level and relation level, the third step is responsible for removing duplicate edges and low-confidence edges, and the last step is responsible for writing the nodes and relations that pass the determination into the main structure of the graph. The processing links thus formed enable the entity nodes and relation triples obtained in the previous section to naturally transition to the graph layer expression.

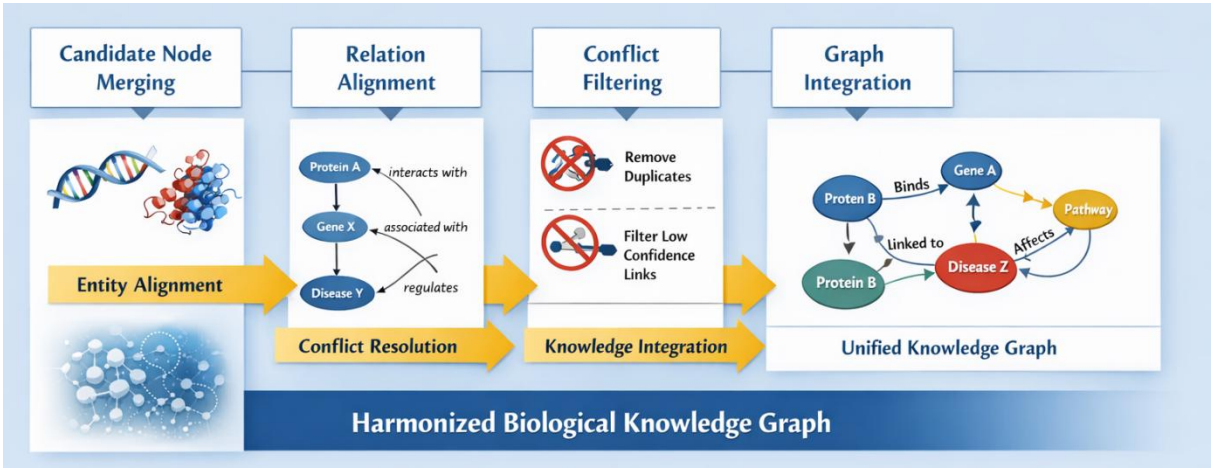

Figure 3: Knowledge graph fusion construction process

After node fusion and relationship alignment are completed, the system unifies them to obtain the final structure written weight:

$$w_{ab}^{(r)} = \frac{\exp(\alpha m_{ab} + \beta o_{ab}^{(r)})}{\sum_{k=1}^{R} \exp(\alpha m_{ab} + \beta o_{ab}^{(k)})} \tag{14}$$

Here, $w_{ab}^{(r)}$ represents the write weight of node pair (a, b) on relation type r,α and β represent the regulation coefficient between node fusion term and relation alignment term, and R represents the total number of candidate relation categories. Equation (14) is used to complete weight normalization among multiple candidate relations and determine the final retained structural relations.

Finally, the system updates the graph adjacency structure according to the writing weights and threshold rules, and the update method is as follows:

$$A_{ab}^{(r)*} = \begin{cases} 1, & w_{ab}^{(r)} \geq \tau_r \\ 0, & w_{ab}^{(r)} < \tau_r \end{cases} \tag{15}$$

where $A_{ab}^{(r)*}$ denotes the updated adjacency state of the relation, and τ_r denotes the write threshold corresponding to the relation type r. When the judgment result of Equation (15) is 1, the system writes node a, relation r and node b into the main structure of the graph. When the result is 0, it is retained as a candidate relation and no graph entry is performed. Equation (15) is used to ensure that the graph writing result has a clear filtering boundary in structure.

After the above steps, the nodes and triples generated in the previous paragraphs no longer stay at the local text level, but are organized as a unified, queryable, and extensible biological knowledge graph. Since node fusion, relationship alignment and structure writing are completed continuously in the same computing link, duplicate nodes, conflict relationships and weak evidence edges in the graph can be effectively controlled, and the final graph structure is more suitable for subsequent retrieval, statistical analysis and graph learning tasks.

4 Experiment and results

This section is used to validate the overall effectiveness of the proposed deep learning method

in the biological knowledge graph construction task. The experimental data are collected from PubMed abstracts, public biological terminology databases and structured relational databases, and the evaluation scope covers five aspects: entity recognition, entity standardization, relation extraction, triple generation and graph fusion construction. The original corpus contains a total of 62400 abstracts, and 182000 entity mentions, 396000 candidate relations and 128000 triplet candidates are obtained after cleaning. Entity types cover five categories: genes, proteins, diseases, drugs, and biological processes. The training set, validation set and test set were divided by 8:1:1, and all models were trained and tested under the same data segmentation conditions to ensure that the results were comparable.

The experimental platform uses Python 3.10, PyTorch 2.1 and CUDA 12.1, and the main card is NVIDIA A100 80GB. The dimension of the entity encoding layer is set to 256, the dimension of the relation representation layer is set to 384, the batch size is set to 32, the optimizer is AdamW, and the initial learning rate is set to 2e-5. The upper limit of candidate nodes in the standardized mapping stage is set to 20, and the graph writing threshold is set to 0.71 according to the results of the validation set. The above setting ensures that entity representation, relation discrimination and structure writing can be done consecutively in the same computing link.

The entity recognition and normalized mapping stage first compares the overall performance differences of different models. The overall results of different models on entity recognition and standardization tasks are shown in Table III. Table 3 shows that the proposed method is superior to BiLSTM-CRF, BioBERT and PubMedBERT in four indicators: recognition accuracy, recognition F1, standardized accuracy and alias alignment accuracy, where recognition F1 reaches 92.8%, standardized accuracy reaches 91.3%, and alias alignment accuracy reaches 89.7%. This result indicates that entity semantic encoding and node mapping are not lifted in isolation, but form a continuous support in a unified link.

Table 3: Comparison of results of different models on entity recognition versus standardization tasks

Model	Recognition Accuracy / %	Recognition F1 / %	Normalization Accuracy / %	Alias Alignment Accuracy / %
BiLSTM-CRF	89.7	89.1	85.8	83.6
BioBERT	91.8	91.2	88.9	86.7
PubMedBERT	92.4	92.0	89.8	88.4
Proposed Method	93.2	92.8	91.3	89.7

In order to further observe the impact of standardized mapping on the graph writing stage, this paper continues to calculate the node compression ratio of different entity categories before and after standardization. The compression effect of nodes for different entity types is shown in Fig. 4. Fig. 4 reflects that the node compression rate of disease class is 29.4%, gene class is 26.8%, drug class is 25.7%, protein class is 24.9%, and biological process class is 23.6%. It can be seen that disease class and gene class entities are more likely to form centralized nodes after standardization, while drug class and protein class still retain a certain degree of name dispersion, which is related to the complexity of term sources and the number of aliases difference.

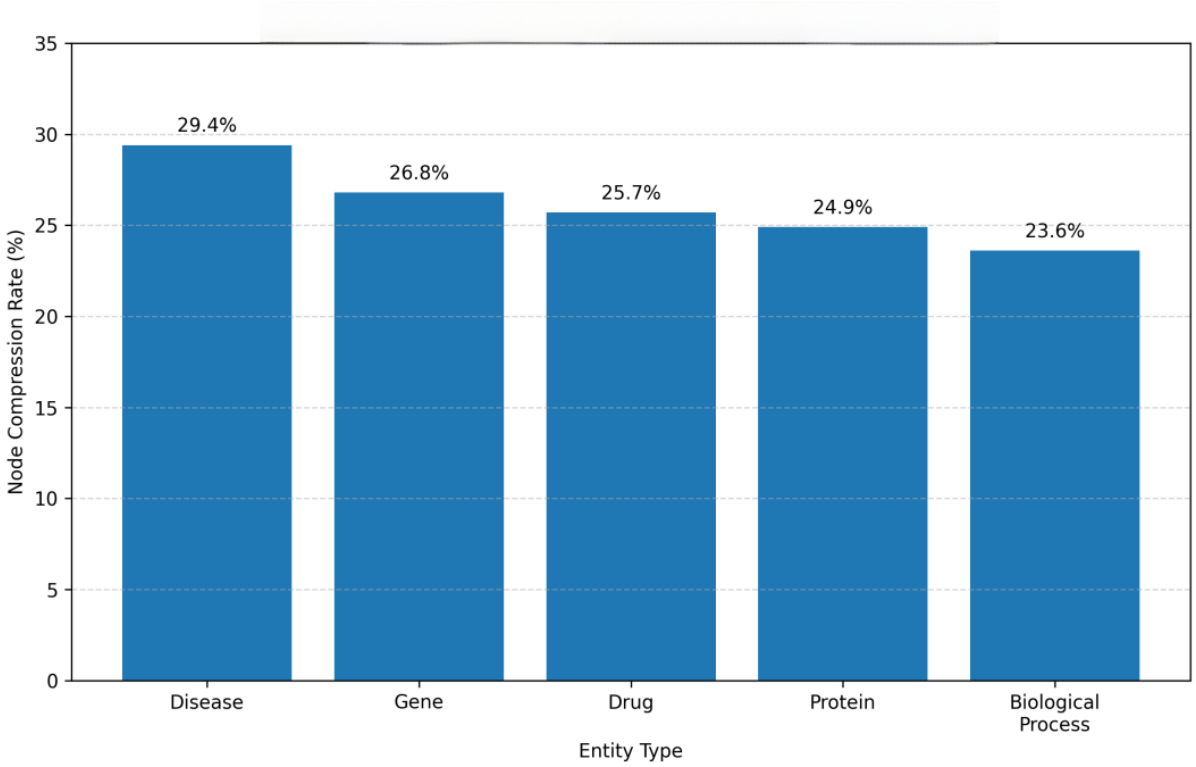

Figure 4: Comparison plots of node compression rates for different entity types

In the stage of relation extraction and triple generation, this paper compares the overall performance of different models. The proposed method achieves 90.6% in relation classification accuracy, 89.8% in macro-average F1 value, and 87.9% in triplet efficiency. As a comparison, the results of BioBERT were 88.9%, 87.2% and 84.6%, SciFive-RE were 89.4%, 87.9% and 85.1%, and BiLSTM-Attention were 85.7%, 83.8% and 80.4%. From the overall results, after deep representation learning and structured relation modeling work together, both relation label determination and triple selection show higher stability.

In order to analyze the specific contribution of each module to the overall performance, this paper further conducts ablation experiments. The performance changes after ablation of different modules are shown in Table IV. Table 4 shows that after removing the entity semantic coding enhancement, the recognition F1 decreases from 92.8% to 91.4%, and the standardization accuracy, relationship classification accuracy and triple effective rate also decrease synchronously, indicating that the quality of front-end semantic representation will continue to affect the subsequent links. After removing the standardized mapping module, the standardized accuracy dropped to 86.1%, and the accuracy of relation classification and the effective rate of triples decreased, which indicated that the node unification results would directly affect the quality of relation modeling. After removing the relation matrix scoring module, the accuracy of relation classification decreases to 89.1%, and the effective rate of triples also decreases, which indicates that the stability of relation discrimination boundary depends on the module. After removing the graph fusion building block, the effective rate of triples decreases to 85.8%, indicating that the subsequent structure fusion has a direct support effect on the quality of relation writing. Overall, there is a stable link dependence between the front and back modules, and the weakening of any single link will affect the final output quality.

Table 4: Performance changes after ablation for different modules

Model Configuration	Recognition F1 / %	Normalization Accuracy / %	Relation Classification Accuracy / %	Triple Validity / %
Full Model	92.8	91.3	90.6	87.9
Without Entity Semantic Encoding Enhancement	91.4	89.8	88.9	85.7
Without Normalization Mapping Module	92.1	86.1	88.8	85.5
Without Relation Matrix Scoring Module	92.3	90.7	89.1	86.1
Without Graph Fusion Construction Module	92.5	90.9	89.4	85.8

In order to observe the convergence differences of different models during training, this paper further statistics the variation trend of macro-average F1 on the validation set with training rounds. The macro-average F1 convergence trends of different models are shown in Fig. 5. The macro-average F1 of our method is 81.6% in the fifth round, increases to 86.9% in the 10th round, reaches 88.7% in the 15th round, and stabilizes at 89.6% in the 20th round. As a control, BioBERT is 87.0% in round 20, SciFive-RE is 87.4%, and BiLSTM-Attention is 83.5%. This result indicates that the proposed method not only has higher endpoint performance, but also maintains a more stable upward trend in the middle and late training stages, and the coupling between relation representation and triple screening is tighter.

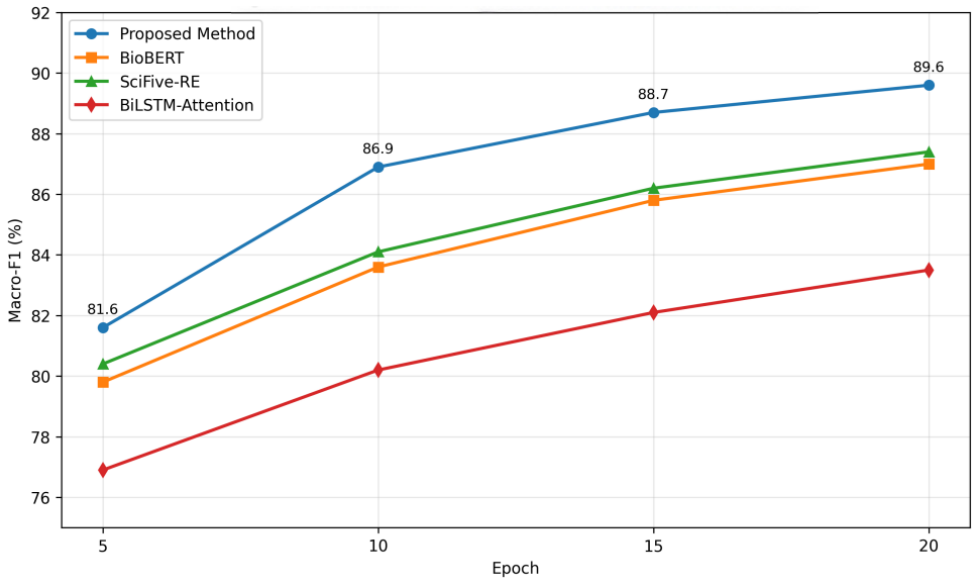

Figure 5: Plot of macro average F1 change with training rounds for different models

In the knowledge graph fusion construction stage, this paper further compares the structural quality and usability performance of different construction methods. The overall comparison results of different graph construction methods are shown in Table 5. Table 5 shows that the node repetition rate of rule-driven construction is 9.8%, the conflict relationship rate is 7.4%, and the effective writing success rate is 79.6%. BioBERT+ rules were written in 7.2%, 5.9% and 83.1%, respectively. Relation extraction + static fusion are

5.6%, 4.7% and 85.9%, respectively; In the proposed method, the node repetition rate is reduced to 4.1%, the conflict rate is reduced to 3.2%, the effective write success rate is improved to 88.4%, and the one-hop query response time is compressed to 84 ms. This result shows that the graph fusion building block not only improves the structural quality, but also enhances the computational efficiency of the graph in the subsequent retrieval scene.

Table 5: Structural quality versus usability comparison of different graph construction methods

Method	Node Redundancy Rate / %	Conflict Relation Rate / %	Valid Write Success Rate / %	Single-Hop Query Response Time / ms
Rule-Driven Construction	9.8	7.4	79.6	126
BioBERT + Rule-Based Writing	7.2	5.9	83.1	108
Relation Extraction + Static Fusion	5.6	4.7	85.9	93
Proposed Method	4.1	3.2	88.4	84

In order to analyze the influence of the change of the atlas writing threshold on the structure quality, this paper continues to compare the results under the four groups of threshold conditions of 0.63, 0.67, 0.71 and 0.75. Fig. 6 shows the graph quality variation under different writing thresholds. The results show that when the threshold is 0.63, the writing success rate reaches 91.2%, but the invalid edge rate increases to 7.9%. When the threshold is 0.67, the invalid edge rate drops to 6.1%, and the write success rate is 89.7%. When the threshold is 0.71, the conflict relationship rate is controlled at 3.2%, and the writing success rate is maintained at 88.4%, which is the most balanced overall. When the threshold is increased to 0.75, the conflict edges continue to decrease, but the valid relations are over-cut, and the write success rate drops to 83.1%. This trend indicates that too low a threshold relaxes the conditions for wrong edges to enter the graph, while too high a threshold weakens effective relation writing, so the fusion stage needs to maintain a balance between structural quality and integrity.

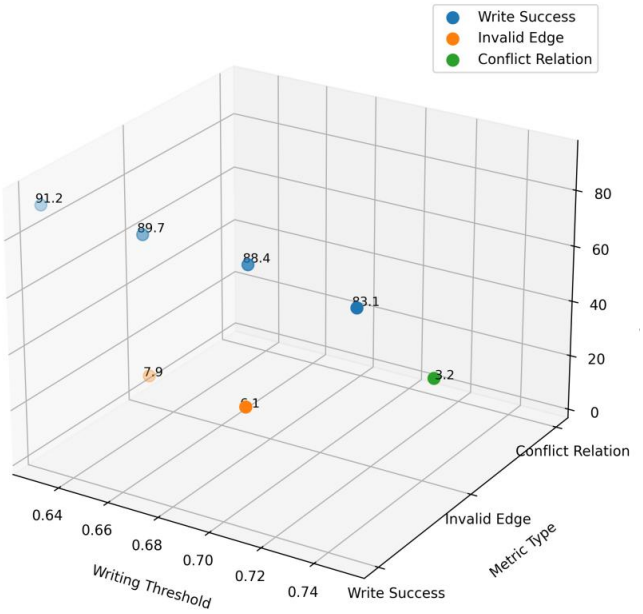

Figure 6: Graph of atlas quality change for different write thresholds

In addition to the overall metrics, this paper also verifies the consistency and usability of the atlas results output by the system. By querying the graph database, it is found that the gene-disease, drug-disease and protein-biological process relationships are the most concentrated in the final generated graph, accounting for 28.7%, 24.9% and 18.6% of all effective edges, respectively. 500 nodes were randomly selected to perform neighborhood retrieval, the average response time was 84 ms, and the node attribute integrity rate reached 96.1%. The average response time for multi-hop path queries is 137 ms. This shows that the fused graph structure not only maintains high semantic consistency, but also supports subsequent retrieval, statistical analysis and graph learning tasks.

Taking the above results together, it can be seen that the proposed method maintains stable performance in the four stages of entity recognition, standardized mapping, relation extraction and graph fusion construction. The experimental results do not only improve on a single indicator, but form a consistent improvement in three levels: node quality, relationship quality and graph usability. This shows that the continuous construction link established around deep learning can better support the complete transformation of biological knowledge graph from text to structure, and also shows that this method has good application value in engineering implementation and subsequent computing tasks.

5 Conclusions and future work

Focusing on the key computational links in the construction of biological knowledge graphs, this paper constructs a deep learning method consisting of entity recognition, standardized mapping, relation extraction, triple generation and graph fusion construction. Experimental results show that the proposed method achieves stable performance in entity recognition, relation classification and graph writing. The accuracy of entity recognition reaches 93.2%, the standardization accuracy reaches 91.3%, the accuracy of relation classification reaches 90.6%, and the effective writing success rate reaches 88.4%. These results show that the continuous computing link constructed in this paper can better support the transformation of biological knowledge from text representation to graph structure representation.

There is still room for further improvement of the proposed method. The utilization of current models for long-distance cross-sentence dependencies is still mainly based on local evidence aggregation, and the global relationship modeling ability in multi-paragraph document scenarios still has room for improvement. Although entity normalized mapping can reduce the node duplication rate, the adaptive alignment ability still needs to be enhanced when facing the terms with large differences in semantic granularity across databases. In the graph fusion stage, node merging and relationship alignment have been realized, but the distinction between weak evidence edges and high similar noise edges still relies on threshold control, which makes the accuracy of structure screening in complex scenes still have room to be improved.

Future research can be carried out along three directions. One direction is to introduce stronger document-level representation mechanisms and cross-sentence reasoning strategies to enhance relation recognition in complex documents. The other direction is to combine the dynamic graph update mechanism and the incremental node fusion method to improve the continuous absorption ability of the graph to new documents, new terms and new relations. Another direction is to design more detailed evaluation metrics for downstream tasks such as graph learning, knowledge reasoning, and question answering retrieval, so that the results of graph construction can be more comprehensively verified at three levels: structural quality, computational performance, and application effect.

References

[1] Sousa D, Couto F M. Biomedical relation extraction with knowledge graph-based recommendations[J]. IEEE Journal of Biomedical and Health Informatics, 2022, 26(8): 4207-4217.

[2] Alshahrani M, Almansour A, Alkhaldi A, et al. Combining biomedical knowledge graphs and text to improve predictions for drug-target interactions and drug-indications[J]. PeerJ, 2022, 10: e13061.

[3] Ren Z H, You Z H, Yu C Q, et al. A biomedical knowledge graph-based method for drug–drug interactions prediction through combining local and global features with deep neural networks[J]. Briefings in bioinformatics, 2022, 23(5): bbac363.

[4] Yang J, Li Z, Wu W K K, et al. Deep learning identifies explainable reasoning paths of mechanism of action for drug repurposing from multilayer biological network[J]. Briefings in Bioinformatics, 2022, 23(6): bbac469.

[5] Gao Z, Ding P, Xu R. KG-Predict: A knowledge graph computational framework for drug repurposing[J]. Journal of biomedical informatics, 2022, 132: 104133.

[6] Erdengasileng A, Han Q, Zhao T, et al. Pre-trained models, data augmentation, and ensemble learning for biomedical information extraction and document classification[J]. Database, 2022, 2022: baac066.

[7] Bang D, Lim S, Lee S, et al. Biomedical knowledge graph learning for drug repurposing by extending guilt-by-association to multiple layers[J]. Nature Communications, 2023, 14(1): 3570.

[8] Chandak P, Huang K, Zitnik M. Building a knowledge graph to enable precision medicine[J]. Scientific data, 2023, 10(1): 67.

[9] Morris J H, Soman K, Akbas R E, et al. The scalable precision medicine open knowledge engine (SPOKE): a massive knowledge graph of biomedical information[J]. Bioinformatics, 2023, 39(2): btad080.

[10] Murali L, Gopakumar G, Viswanathan D M, et al. Towards electronic health record-based medical knowledge graph construction, completion, and applications: A literature study[J]. Journal of biomedical informatics, 2023, 143: 104403.

[11] Lyu K, Tian Y, Shang Y, et al. Causal knowledge graph construction and evaluation for clinical decision support of diabetic nephropathy[J]. Journal of Biomedical Informatics, 2023, 139: 104298.

[12] Lin Y, Lu K, Yu S, et al. Multimodal learning on graphs for disease relation extraction[J]. Journal of biomedical informatics, 2023, 143: 104415.

[13] Wang Z, Wei Z. PT-KGNN: A framework for pre-training biomedical knowledge graphs with graph neural networks[J]. Computers in Biology and Medicine, 2024, 178: 108768.

[14] He J, Li F, Li J, et al. Prompt tuning in biomedical relation extraction[J]. Journal of

healthcare informatics research, 2024, 8(2): 206-224.

[15] Li Z, Wei Q, Huang L C, et al. Ensemble pretrained language models to extract biomedical knowledge from literature[J]. Journal of the American Medical Informatics Association, 2024, 31(9): 1904-1911.

[16] Yuan J, Zhang F, Qiu Y, et al. Document-level biomedical relation extraction via hierarchical tree graph and relation segmentation module[J]. Bioinformatics, 2024, 40(7): btae418.

[17] Zhang Y, Yang Z, Yang Y, et al. Location-enhanced syntactic knowledge for biomedical relation extraction[J]. Journal of Biomedical Informatics, 2024, 156: 104676.

[18] Jia Y, Wang H, Yuan Z, et al. Biomedical relation extraction method based on ensemble learning and attention mechanism[J]. BMC bioinformatics, 2024, 25(1): 333.

[19] Schäfer H, Idrissi-Yaghir A, Arzideh K, et al. Biokgrapher: initial evaluation of automated knowledge graph construction from biomedical literature[J]. Computational and Structural Biotechnology Journal, 2024, 24: 639-660.

[20] Ivanisenko T V, Demenkov P S, Ivanisenko V A. An accurate and efficient approach to knowledge extraction from scientific publications using structured ontology models, graph neural networks, and large language models[J]. International Journal of Molecular Sciences, 2024, 25(21): 11811.