IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 November 2024, accepted 11 December 2024, date of publication 23 December 2024, date of current version 25 July 2025.

Digital Object Identifier 10.1109/ACCESS.2024.3520971

RESEARCH ARTICLE

Few-Shot Named Entity Recognition Based on the Collaborative Graph Attention Network

HAORAN NIU AND ZHAOMAN ZHONG
Jiangsu Ocean University, Lianyungang, Jiangsu 222005, China
Corresponding author: Zhaoman Zhong (zmzhong@jou.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 72174079.

ABSTRACT Few-shot Named Entity Recognition (NER) aims to extract entity information from limited annotated samples, addressing the scarcity of data in specialized domains. However, existing few-shot NER methods relying on data augmentation struggle to adequately augment semantic features, limiting their learning and representation capabilities. To overcome this, we introduce a novel few-shot NER encoder based on a Collaborative Graph Attention network (ColGAT). This encoder utilizes a collaborative graph-based data augmentation mechanism to thoroughly extract latent semantic features of entities within sentences, enabling precise entity recognition. Furthermore, to facilitate information interaction between support and query sets, we develop an entity classifier with Match Processing (MP), where adaptive weights allow the support set to flexibly adapt to different query instances, enhancing entity classification performance. Our model achieved an average F1 result of 65.87% across six datasets, surpassing the second-ranked model by 2.19% and achieving state-of-the-art performance, demonstrating significant improvements over previous methods.

INDEX TERMS Few-shot named entity recognition, collaborative graph attention network, match processing, data augmentation.

I. INTRODUCTION

In the era of rapid advancements in information technology and the internet, people operate within an environment characterized by a deluge of data, both in volume and complexity. Efficiently, accurately, and comprehensively processing and utilizing this data has emerged as a critical challenge. Knowledge graphs, which organize entities, concepts, and relations of the real world into a large-scale, cross-domain network, offer a powerful tool for comprehending and leveraging vast amounts of data. Essentially, knowledge graphs represent a specialized visual manifestation of knowledge bases, storing entities and relations in the form of triples (head entity, relation, tail entity). This representation not only enhances the ability to organize knowledge but also improves human perception of the stored information.

Named Entity Recognition (NER) aims to identify entities and their types from diverse data sources. NER plays a pivotal role in knowledge acquisition and syntactic analysis, underpinning the practical application of natural language processing techniques. The present study focuses on extracting span information of entities from natural language text, aiming to pinpoint entity locations and classify them accurately. FIGURE 1 shows an example of span and type recognition of entities in a sentence.

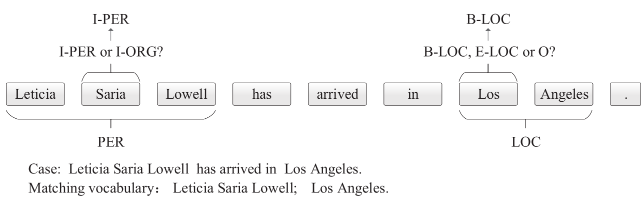


FIGURE 1. An example of the named entity recognition.

Addressing the challenge of sparse training samples in NER, researchers have explored the integration of few-shot learning methods. This approach enables the model to

The associate editor coordinating the review of this manuscript and approving it for publication was Maria Chiara Caschera.

recognize entity information with fewer training samples, as compared to traditional deep learning methods. However, the scarcity of training data often leads to overfitting and limited generalization capabilities in few-shot NER models. To mitigate this issue, data augmentation [1] techniques have been widely applied, effectively expanding the training corpus and enhancing model generalization through introducing variations such as scaling, flipping, noise addition, and synonym replacement. Nevertheless, traditional text structure transformation methods may constrain the model's learning capacity, focusing excessively on grammatical structures while overlooking the influence of semantics and context.

Therefore, this research endeavors to devise a data augmentation method tailored for few-shot NER, capable of transforming syntactic structures and reinforcing the connections between entities and their contextual environment. Specifically, we propose a few-shot NER model grounded on a Collaborative Graph Attention Network (ColGAT), which recognizes entity information from sparse training data. Our approach incorporates an Enhanced Semantic Boundary method, augmenting the training data by constructing three collaborative graphs for each sentence. This strategy provides syntactic structure information and harnesses semantic cues at phrase boundaries to deeply analyze sentence features, yielding more precise entity representations. Subsequently, the ColGAT sequentially inputs the three collaborative graph layers into the graph attention network, excavating graph structural features and enhancing parameter efficiency. Finally, we establish an entity classifier based on match processing, facilitating information interaction between the support set and query set through adaptive weights in the match processing unit.

II. RELATED WORKS

Few-shot Named Entity Recognition (Few-shot NER) methods can be broadly categorized into three main types based on their focus and approach: word semantics-based methods, label semantics-based methods, and prompt-based methods. Each of these approaches leverages different sources of information to address the challenges posed by limited training data in few-shot scenarios. Word semantics-based methods [2], [3], [4] focus on the semantic information inherent in individual words. These methods assume that each word possesses an intrinsic semantic representation, which is mapped to specific entity categories. To address nested named entity structures, Ming et al. [5] proposed span-based and hierarchical approaches to model the structure of nested entities, and leverage the core semantics of words, which are particularly effective for words with clear semantic characteristics. However, when the entity categories are numerous and their distribution is complex, traditional few-shot methods struggle to distinguish subtle differences between categories. To overcome this limitation, Zha et al. [6] introduced a contrastive learning-enhanced two-stage prototypical network (CEPTNER) to improve category distinction. By combining contrastive learning and prototypical learning, their method enhances semantic alignment between samples and categories. This approach reduces reliance on large-scale data by judging based on word and contextual semantics. Similarly, Yang and Katiyar [7] proposed a few-shot NER model employed a sentence encoder to extract contextual semantics and a decoder that combines nearest-neighbor meta-learning with structured methods to capture semantic associations between entities. However, a significant limitation of these methods lies in their susceptibility to linguistic polysemy. Many words exhibit different meanings depending on the context, which remains a challenge for word-semantics-based approaches.

Label semantics-based methods [8], [9], [10] focus on understanding the semantic meaning of entity labels. These methods typically involve two steps [11]: first, identifying potential entity spans in the text, and then determining the entity type based on the semantic information embedded in the labels. Zhou et al. [12] introduced a Semantics-induced Optimal Transport (SOT) mechanism that leverages semantic embeddings of both samples and labels to construct an optimal transport plan. By modeling the semantic associations between word semantics in the samples and label semantics, this approach enhances classification and recognition accuracy. Feng et al. [13] proposed a taxonomy-guided prototypical approach that utilizes hierarchical relationships among labels to model few-shot NER tasks. By constructing a prototype representation for each label and incorporating its hierarchical contextual information, their method improves the model's ability to identify entity types. However, accurately extracting and understanding label semantics often require sophisticated techniques and extensive domain knowledge. In cases where label semantics are inaccurately extracted or ambiguous, the model's performance can be significantly degraded.

Prompt-based methods [14], [15], [16] enhance few-shot NER by introducing external prompt information, such as domain-specific knowledge, templates, rules, or other forms of prior knowledge. These prompts serve to guide the model in understanding task-specific requirements, leveraging external information to compensate for the lack of training data. Ye et al. [17] proposed a Decomposed Two-Stage Prompt Learning framework, which involves designing task-specific prompts for each subtask. These prompts explicitly convey task requirements to the model, enabling it to effectively decompose and optimize the task. By integrating domain knowledge and prior information through prompt learning, this approach improves the model's ability to identify entities in specialized domains. Chen et al. [18] introduced a few-shot NER model based on a self-descriptive mechanism, which utilizes descriptive instances and a general concept set to represent entity types. This mechanism transfers knowledge from external resources to address the issue of sparse features in few-shot learning. Similarly, Chen et al. [19] incorporated biomedical domain knowledge to generate task-relevant instances, alleviating data scarcity in few-shot scenarios. They employed prompt-based learning by

TABLE 1. A 2-way 1-shot example of the NER.

Entity type	[Organization] {Location}
Support instances	(1) "[The Lakers] were founded in 1947, joined the NBLleague and won the championship." (2) "Kobe D Bryant was born in {Pennsylvania} on August 23, 1978."
Query instances	"Yao Ming was drafted by the Houston Rockets as a champion in the 2002 NBA draft."
Desired output	"Yao Ming was drafted by the {Houston} [Rockets] as a champion in the 2002 NBA draft."

constructing contrastive pairs, enhancing the model's ability to distinguish between positive and negative samples. The flexibility of prompts allows them to be adapted to different tasks and domains, making them highly scalable and suitable for complex application scenarios. However, in handling complex and diverse text structures and linguistic expressions, prompts may need frequent updates and optimization to maintain their effectiveness.

Our ColGAT-MP is a few-shot named entity recognition model that incorporates Label semantics. Its innovation lies in the introduction of collaborative graphs and dynamic matching mechanisms, making it particularly well-suited for handling complex semantic feature modeling tasks in few-shot learning.

III. PROBLEM DEFINITION

In the few-shot NER task, the N-way K-shot sample training and testing approach is adopted. Specifically, during each training iteration, a support set $\mathcal{S} = \{(X^{(i)}, Y^{(i)})\}^{N\times K}$ is provided, which comprises N entity classes, with K sentence instances belonging to each class. Additionally, a query set $\mathcal{Q} = \{(X^{(i)}, Y^{(i)})\}^{N\times Q}$ is presented, containing the same N classes but with Q different instances in each class. The goal of this N-way K-shot approach is to empower the Few-Shot NER model to rapidly learn and recognize entities from limited training data, subsequently applying the acquired knowledge to efficiently predict entity types within the query set. TABLE 1 illustrates an example of a 2-way 1-shot NER scenario, where only one training sample is available for each class.

The Few-Shot NER is a sequence labeling task that aims to assign type labels to each token in a given sequence. The label types are typically represented using distinct symbols or tags [20], for example, 'P' for person, 'L' for location, 'G' for organization, and 'O' for non-entity words (indicating that a token does not belong to any of the predefined entity types).

Given the input $S = (X, Y)$, where $X = \{x_1, \cdots, x_L\}$ represents the sentence sequence consisting of L tokens, and $Y = \{y_1, \cdots, y_L\}$ denotes the corresponding sequence of labels for each token in X. The objective of the Few-Shot NER model is to accurately identify the spans (i.e., the start and end positions) and types of entities within the sentence. This involves not only recognizing individual tokens that belong to an entity but also correctly classifying the entire entity span into one of the predefined entity types.

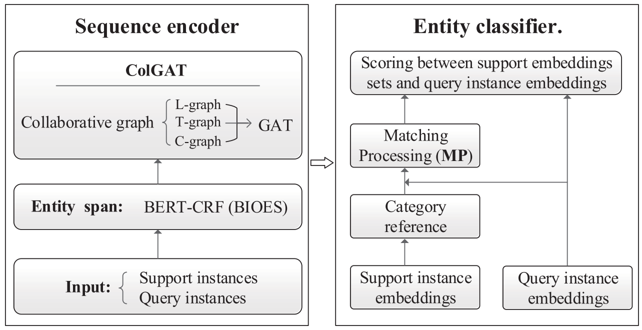


FIGURE 2. Overall framework of few-shot NER model based on ColGAT-MP.

IV. THE FEW-SHOT NAMED ENTITY RECOGNITION MODEL BASED ON COLLABORATIVE GRAPH ATTENTION NETWORK-MATCH PROCESSING

A. OVERALL FRAMEWORK

The few-shot NER model comprises two key stages: a ColGAT-based sentence encoder and an MP (Match Processing)-based entity classifier. In the encoding stage, the BERT-CRF algorithm is employed to initially obtain word embeddings and perform entity boundary segmentation. These outputs are then utilized to construct a collaborative graph, whose layers are sequentially fed into a Graph Attention Network (GAT). The GAT processes the graph to generate collaborative graph node embedding representations. During the entity classification and prediction stage, a Match Processing Unit is designed to facilitate information exchange between the query set and the support set. This enables the support prototypes to adaptively align with the query instances. Scoring operations are subsequently conducted, leveraging the proximity between the support references and the query instances, to predict the entity types for the query instances. The overall architecture of the few-shot NER model is illustrated in FIGURE 2.

The model framework primarily encompasses word-level feature representation and metric learning methodologies. During the encoding of word embeddings, to mitigate the interference caused by redundant and irrelevant feature information on model parameter training, the few-shot NER model emphasizes the importance of capturing global semantic information for precise discrimination of entity boundaries. The ColGAT encoder mitigates the impact of noise information on the encoded embeddings by enhancing the representation of useful features. In the meta-learning phase, traditional metric-learning-based few-shot learning models tend to overlook the criticality of target domain matching, solely focusing on the source training domain. This oversight can hinder the full exploitation of feature information within the source domain, especially in light of the notable domain bias present in few-shot NER datasets.

TABLE 2. An example of sentence labeling.

Token	Leticia	Saria	Lowell	has	arrived	in	Los	Angeles
Label	B	I	E	O	O	O	B	E

Consequently, the MP-based entity classifier is designed to incorporate target domain information, enabling it to handle sample data from diverse domains with greater stability. By integrating the encoded information of samples into the similarity calculation process within the entity classifier, this few-shot NER model framework effectively leverages the representations to achieve desirable experimental outcomes.

B. BIOES LABELING OF SENTENCES BASED ON BERT-CRF

The BERT-CRF (Bidirectional Encoder Representations from Transformers - Conditional Random Field) framework can effectively reduce noise in annotated data through joint training of BERT and CRF, thereby improving the model's robustness. An example of a sentence sequence labeled using the BERT-CRF model is shown in TABLE 2.

The BERT-CRF model outputs a set of BIOES subscripts to indicate entity span boundaries: B (Begin) denotes the start word of an entity; I (Inside) denotes internal words belonging to an entity; O (Outside) denotes words not belonging to any entity; E (End) denotes the last word of an entity that is not followed by another word belonging to the same entity; and S (Single) denotes a word that is an independent entity.

The goal of the entity classification task is to assign a specific entity type to each span, which requires first obtaining the span representation of the entities. After BERT performs contextual encoding on an input sentence with L tokens, the corresponding entity token spans representation $x_{[i,j]}$, i.e., from the start token $x_{[i]}$ to the end token $x_{[j]}$, are obtained. The span representation of $s_{[i,j]}$ is calculated by averaging the embeddings of all the tokens within the $X_{[i,j]}$:

$$S_{[i,j]} = \frac{1}{j-i+1}\sum_{k=i}^{j} X_k \tag{1}$$

It is important to note that the BERT-CRF model can indeed perform classification on labeled entities. However, the BERT-CRF model has high data requirements for the entity classification task and necessitates sufficient annotated data for optimal performance. In the few-shot entity classification task, data imbalance is an issue, where the number of instances for different entity types can vary greatly, potentially leading to poor performance for underrepresented types, especially when the number of samples is significantly lower. Additionally, the BERT-CRF model is limited by the input sentence length, which can be problematic for tasks involving long documents or paragraphs, where important entities may be missed due to truncation. To address these issues, the syntactic transformation-based data augmentation method and the adaptive type match processing approach in a meta-learning mode are adopted. The syntactic transformation-based data augmentation method artificially generates additional labeled data by transforming existing sentences in syntactically meaningful ways, thereby increasing the diversity and quantity of training examples. The adaptive type match processing approach, on the other hand, dynamically adjusts the model's focus during training to better handle the imbalance in entity type representations, effectively leveraging the limited training data to enhance the model's few-shot learning capability for entity classification.

C. COLLABORATIVE GRAPH CONSTRUCTION

After extracting entity span embeddings utilizing the BERT-CRF model, we devised a data augmentation method tailored specifically for NER tasks. This approach aims to mitigate the model's over-reliance on the predefined label set, which can hinder its generalization ability. Drawing inspiration from Sui et al. [21], who introduced a graph-based representation of sentences centered on Chinese characters, we propose a collaborative graph strategy for enhancing the interplay between words and phrases. This strategy addresses the challenge of inadequate contextual feature information for entities, particularly in complex scenarios.While the BERT-CRF model inherently captures contextual information surrounding individual words, it struggles to extract nuanced semantic features for entity embeddings in intricate contexts, including nested, ambiguous, or cross-domain entities. Given the scarcity of training data in few-shot settings, the model often fails to accurately identify the type of these entities. The collaborative graph strategy that augments the training data and enhances the model's sensitivity to contextual semantics. This strategy decomposes each instance into three distinct graph layers: the Containment Graph (C-graph), which captures hierarchical relationships among entities; the Transformation Graph (T-graph), which transforms and enriches contextual features for entity representations; and the Lattice Graph (L-graph), inspired by lattice structures, which models intricate character-word interactions, thereby compensating for the lack of contextual features. Collectively, these graph layers work in harmony to fully exploit the semantic information of entities, ultimately improving the performance of few-shot NER systems by augmenting the training data and enhancing the model's contextual understanding.

The C-graph represents each entity as a node and associates it with the words it comprises, thereby modeling the intrinsic semantic correspondence between the entity and its constituent words. This approach leverages the contextual information within the entity boundaries to capture semantic relations among the words, as words within an entity often exhibit some form of connection or semantic relatedness. By explicitly modeling this correspondence, the model can more effectively utilize the internal semantic information, enhancing its ability to recognize nested entity types. For instance, in a person's name, the surname and given name inherently share a semantic relationship that the C-graph captures, contributing to improved recognition accuracy.

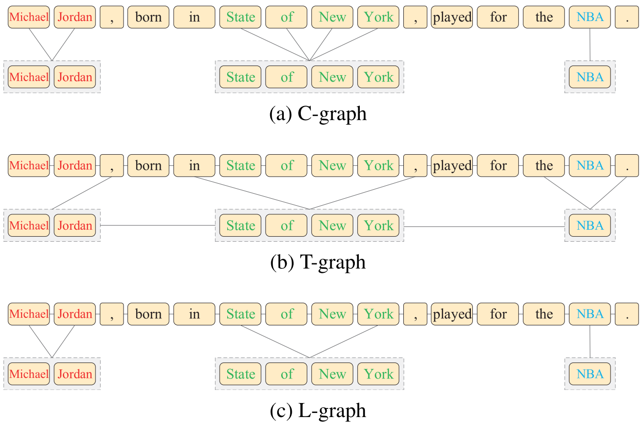


FIGURE 3. **Example of the construction of the collaborative graph. (a) C-graph. (b) T-graph. (c) L-graph.**

An illustrative example of a sentence transformed into a C-graph is presented in Figure 3(a).

The T-graph focuses on the contextual words surrounding each entity, within a specified range. Neighboring words preceding and following the entity occurrence in the text are extracted as nodes, and edges are established between entity nodes based on their contextual co-occurrence relations. The choice of window size for defining the contextual boundary is crucial and can be adjusted according to the task's requirements. In this work, to minimize noise, a window size of 1 is adopted. This T-graph construction enables the model to leverage contextual information more effectively, improving its performance in recognizing entity features. For example, when two person names appear in the same event description, the T-graph's edges can capture their contextual semantic relation, enhancing the accuracy of similar entity type recognition. An example of a sentence transformed into a T-graph is depicted in Figure 3(b).

The L-graph adopts the principles of the Lattice LSTM structure to represent the sentence as a graph. It utilizes entities and their positions in the text as connecting points, establishing edges between entity nodes based on their relative positions. This approach captures positional relations between entities, which are crucial for understanding their contextual significance. When entities overlap in the text, edges are created between them, facilitating the model's comprehension of their spatial arrangement. For instance, adjacent entities in a sentence are likely to be related, and the L-graph's edges enable the model to capture this positional relation, enhancing the recognition accuracy of nearby entities. An illustrative example of a sentence transformed into an L-graph is shown in Figure 3(c).

The constructed C-graph, T-graph, and L-graph are collectively referred to as the Collaborative Graph. Since the node sets of the three graphs are the same while the edge sets are completely different, the adjacency matrices of the three graphs have the same shape but different contents. The 0 / 1 elements in the adjacency matrix represent whether there is an association between the nodes - if phrase i and word j are connected by an edge, then $A_{[i][j]} = 1$, otherwise $A_{[i][j]} = 0$.

Each designed graph layer focuses on different aspects of information, such as hierarchical relationships, contextual transformation, and character-word interactions. Through the collaborative operation of these layers, the model captures and comprehends the semantic information of entities in a more comprehensive manner. The collaborative graph design effectively augments the training data, particularly in few-shot learning scenarios, by providing richer contextual information and features. By leveraging enhanced training data and the structural advantages of graphs, this approach compensates for data scarcity, thereby improving the model's generalization ability and its capacity to recognize complex entities.

D. SENTENCE ENCODING BASED ON COLLABORATIVE GRAPH ATTENTION NETWORK

The sentence encoding framework leveraging ColGAT is illustrated in FIGURE 4. Initially, the BERT-CRF model processes the input sentence, generating a series of BIOES tags to delineate the boundaries of entity spans, alongside word embeddings that capture semantic nuances. Subsequently, adhering to the principles of collaborative graph construction, the annotated sentence is transformed into three distinct collaborative graphs. These graphs facilitate the derivation of adjacency matrices (A_c, A_t, A_l), each encapsulating the structural intricacies of the sentence under varying word-level interactions. Building upon these matrices, we introduce a ColGAT model tailored to the collaborative matrix representation. Within this network architecture, the collaborative graph matrices and initial node embeddings are sequentially fed into the Graph Attention Network (GAT). The GAT subsequently employs an attention mechanism to quantify the degree of association between disparate nodes, thereby updating the node embeddings in a manner that integrates both the inter-node interactions and the contextual integrity of the sentence. This approach ensures that the model holistically considers the intricate relationships within the sentence while preserving its contextual richness.

In the realm of collaborative graph encoding, the conventional approach involves separately feeding three collaborative matrices into distinct GAT, subsequently aggregating neighbor features around each node to encode the input collaborative graph. However, upon scrutinizing the data samples, we discern that the majority of sentences translate into simplistic graph structures, which inherently harbor substantial redundancy in feature information. The adoption of three independent GAT, while feasible, poses the risk of introducing an unwieldy number of training parameters and fostering model redundancy. To address this issue, we propose a streamlined approach by designing a unified GAT architecture tailored specifically for collaborative graph matrices. This novel architecture, termed ColGAT, sequentially processes the three collaborative graphs within a single GAT instance. As depicted in the ColGAT section of

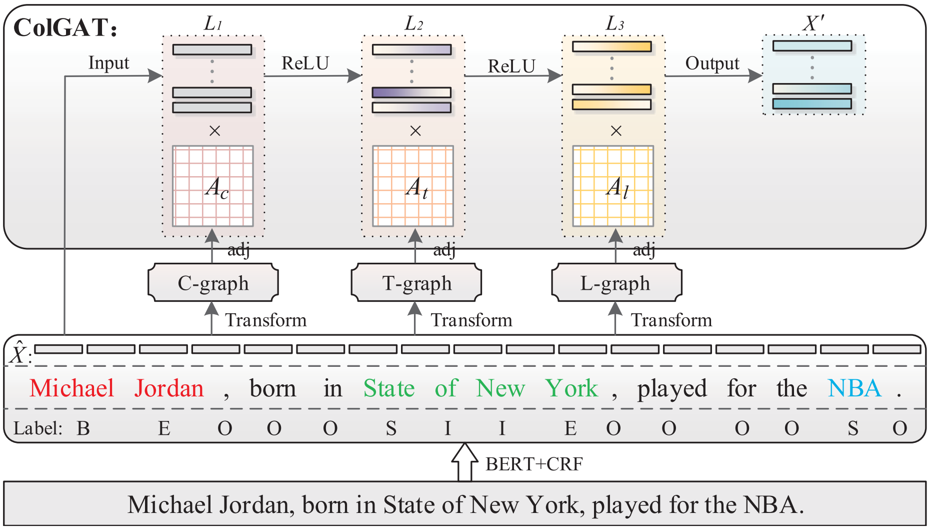


FIGURE 4. Framework of the sentence encoding based on ColGAT.

FIGURE 4, the initial node embeddings within a collaborative graph are denoted as $\hat{X}$. Each subsequent graph input serves as an aggregation layer within the GAT, thereby structuring the ColGAT Network as a three-layered framework. After each aggregation phase, the derived neighbor features are passed through the ReLU activation function, facilitating their transition to the subsequent layer. Ultimately, after traversing the three ColGAT layers, the refined sentence representation, denoted as X', is attained. This refined approach not only mitigates the issue of parameter proliferation but also enhances efficiency by minimizing the generation of redundant information.

In the encoding process of the collaborative graph layers using the GAT, the adjacency matrices A_c, A_t, A_l of the C-graph, T-graph, and L-graph layers are sequentially added to each layer of the network, and the node embeddings trained in the previous layer are used as the input to the next layer. Each iteration of training maintains the same dimension of the input and output embeddings. The collaborative graph node embeddings is denoted as $\hat{X} \in \mathbb{R}^{m \times d}$, where m is the number of nodes in the collaborative graph. Then, attention coefficients:

$$\alpha_{ij} = \text{LeakyReLU}\left(\phi\left(We_i, W\, e_j\right)\right) \tag{2}$$

where LeakyReLU is the activation function, and ϕ is the inner product operation $\mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$. The attention coefficient α_{ij} represents the influence of the neighbor node j on the encoded node i. The α_{ij} serves as a measure of the influence of neighbor node j on the encoded node i, and can be used to calculate the forward hidden state of node i. Furthermore, the attention coefficients are assigned to all the node embeddings, which are then input into the GAT to obtain the node embeddings for the next layer of the network. The computation process:

$$X^{l+1} = \text{ReLU}(WX^l A^l) \tag{3}$$

where l is the layer number of the network, the initial word and phrase embeddings input are $X^1 = \hat{A}\hat{X}$, and $\hat{A} \in \mathbb{R}^m$ is the collection of attention coefficients α_{ij}. At each layer, the attention coefficients A^l are sequentially selected from A_c, A_t, A_l, and the final output sentence embedding is X'.

ColGAT integrates graph neural networks with a multi-layer graph structure. For each graph layer, the model computes attention scores between each node and its neighboring nodes to determine the direction and intensity of information flow. This design effectively models the semantic relationships between tokens and entities while enabling information sharing across multiple graph layers. Traditional sentence structure representations often fail to fully capture the intricate relationships between tokens. By leveraging these graph layers, ColGAT not only enhances the semantic representation of individual entities but also captures complex interactions between entities and their surrounding context. This approach improves data utilization efficiency and strengthens the model's ability to understand and recognize entities. Furthermore, contextual information for each word is directly integrated into the feature representation of the graph nodes, allowing the vocabulary and its most relevant contextual words to form matching entities while disregarding unrelated words.

E. ENTITY CLASSIFIER BASED ON MATCH PROCESSING

In the few-shot NER scenarios, traditional machine learning methods often struggle with the perils of overfitting or underfitting due to the scarcity of training data. To circumvent these limitations, we introduce a novel entity classification prediction framework that leverages a match processing unit as its cornerstone. This unit innovatively learns a set of weights for the embeddings of each reference instance within the support set, quantifying their respective significance for the given task. These weights are dynamically calculated based on a harmonious interplay between the network's predictions and the ground truth labels, enabling the model to adaptively prioritize relevant instance features. Concretely, the method commences by comparing the embeddings of the query entity span, denoted as q_r, against the reference embeddings of entity spans in the support set, formulated as $S_r = \{s_{rk}\}_{k=1}^{K}$, $s_{rk} \in X'$, where s_{rk} represents the encoded phrase embeddings, and S_r comprises the collection of entity spans sharing the same entity type within the support set. This comparison mechanism, in essence, equips the model with the ability to discern the most pertinent instances from the limited support set, thereby fostering more informed and robust predictions in the few-shot NER paradigm.

Considering the semantic interaction between the support set and the query set, a metric function $\varphi(q_r, s_{rk})$ is defined to measure the semantic similarity between the q_r and s_{rk}, which can serve as the interaction result between the two embeddings. The embedded dot product method is used to implement the similarity interaction calculation between the query embeddings and the reference embeddings, represented as: $\varphi(q_r, s_{rk}) = (q_r \cdot s_{rk})$. Thus, instead of learning static embedding representations to predict queries, a adaptive weighted approach is adopted to obtain the reference representation $\mathcal{G}(S_r)$ that is matching the query set:

$$\mathcal{G}(S_r) = \sum_{s_{ri} \in S_r} \beta_k S_{rk} \tag{4}$$

$$\beta_k = \frac{\exp(\varphi(q_r, s_{rk}))}{\sum_{s_{rj} \in S_r} \exp(\varphi(q_r, s_{rk}))} \tag{5}$$

where β_k represents the adaptive weights, capturing the semantic association between the support embedding and the query embedding. This causes the features with similar meanings to the query set to be more referential, allowing the support set S_r to have an adaptive representation for different queries. To make predictions, a metric function is used to calculate the semantic similarity between the query embedding and the reference representation, represented as $\delta(q_r, S_r) = q_r \cdot \mathcal{G}(S_r)$, $\delta()$ is the cosine similarity measure. The larger the value of $\delta(q_r, S_r)$, the more consistent the query embedding is expected to be with the support set class type.

The match processing (MP) unit demonstrates excellent entity classification performance in few-shot NER scenarios due to its ability to adaptively capture differences in the characteristics of various query instances. This flexibility helps avoid the overfitting problem commonly encountered in few-shot learning. Unlike traditional methods with fixed embedding representations, the MP unit employs an adaptive weighting approach, dynamically adjusting the embeddings of reference instances to enhance the semantic representation of query entities. For example, if a query entity shares high semantic similarity with certain reference entities, those references are assigned higher weights, which directly influences the classification outcome of the query entity. This allows the model to effectively extract the most relevant information from a limited set of reference samples, improving

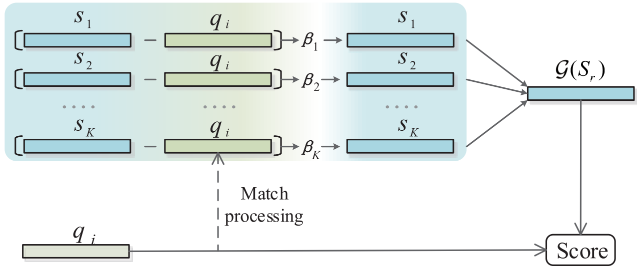


FIGURE 5. Schematic diagram of entity classifier based on match processing.

the accuracy of entity recognition. Additionally, the MP unit leverages an attention mechanism to encode inputs, enabling the model to comprehensively account for interactions between different inputs. This encoding approach effectively captures associative information between instances, resulting in more accurate predictions. The flexible structure of the classifier further enhances its performance, as the attention mechanism's parameters can be easily adjusted to accommodate various tasks and scenarios. This adaptability improves both the model's generalization ability and its classification accuracy in diverse few-shot learning contexts. The schematic diagram of entity classifier based on match processing is shown in FIGURE 5.

The model is trained on the training set. Based on the distance between the support set and the query set, the probability that belongs to an entity type is calculated, thereby representing the possibility of the instance belonging to each class type. The calculation of the entity type probability:

$$p(y_k; x_{[i,j]}) = \frac{\exp\{-\phi(q_r, S_{\text{train}})\}}{\sum_{y_i \in I} \exp\{-\phi(q_r, S_{\text{train}})\}} \tag{6}$$

where $y_i \in Y$ represents the ground truth entity type, $q_r \in Q_t rain$ is the query embedding representation encoded by the ColGAT. Finally, the cross-entropy loss function is used as the optimizer for this few-shot NER model, and the calculation of the model loss:

$$L(\gamma) = \sum_{x_{[i,j]} \in Q_{train}} - \log p(y_{[i,j]}, x_{[i,j]}) \tag{7}$$

where γ represents the training hyperparameters. During type prediction, given a new (S_{train}, Q_{train}, Y_{train}) in an iteration, the model is meta-trained on the provided data. The trained model is then used to calculate the prototypes of all the new classes y_k) in the training set. Subsequently, the model is used to infer the type for the entity span $x_{[i,j]}$ in the query set, and the type label with the highest probability is assigned as the predicted entity type for the query.

V. ALGORITHMS

In the few-shot NER model based on ColGAT-MP, the training sentence features is input to the ColGAT, which outputs the word embeddings. The embeddings are then input to the entity classification learning framework, where the

TABLE 3. Statistics of the few-shot NER datasets.

Dataset	CoNLL 2003	WNUT17	Res	Movie1	Movie2	Re3d	OntoNote 5
Domain	News	Social	Review	Review	Review	Defense	Mixed
#Classes	4	6	8	12	12	10	18
#Test	3453	1287	1521	1953	2443	200	5089

match processing unit is used to compute the probability of the query entities belonging to each type, thereby performing the entity classification. The few-shot NER model based on ColGAT-MP is described as:

Algorithm 1 The Few-Shot NER Model Based on ColGAT-MP

Input: support set and query set: N-way $(\mathcal{S}, \mathcal{Q})$;
Output: probability of query entity type: p;

for N-way **do**:
  for K&Q-shot $X \in (\mathcal{S}, \mathcal{Q})$ **do**:
    Initialize the support and query sentence embeddings $X = \{x_1, \cdots, x_L\}$ using BERT;
    Obtain the entity spans $S_r = \{s_{[i,j]}\}$ in BIOES format using CRF;
    Construct the collaborative graph: C-graph, T-graph, and L-graph;
    Calculate the self-attention coefficients among nodes in C-graph, T-graph, and L-graph using Equation (2);
    Output the encoded sentence embeddings X' using Equation (3);
  end for
  Calculate the support entity type reference weight β_k using Equation (5);
  Measure the semantic similarity between the query instance and the reference: $\delta(q_r, S_r) = q_r \cdot \mathcal{G}(S_r)$;
  Calculate and output the probability of the query entity belonging to the support type using Equation (6);
end for
Calculate the model loss L using Equation (7).

VI. EXPERIMENTS

A. DATASET AND SETTING

The OntoNotes 5 dataset [22] is used as the training data for the few-shot NER model, and sentence sequences with a length greater than 100 and without any mentioned entities are filtered out. Six other benchmark datasets are used as the target datasets for evaluating the model. The datasets providing experimental results are: CoNLL 2003 [23], WNUT17 [24], Re3d, the sub-datasets Res, and the MIT corpus [25], [26] including Movie1 and Movie2. The statistics of these 7 datasets are shown in TABLE 3.

The training environment for the model in this work is a 24G NVIDIA GeForce RTX 3090 GPU. The results of the baseline models are obtained from the relevant literature. For BERT-CRF entity boundary labeling, the learning rate is set to $5e$-5, and the embedding dimension d is 768. The optimization is performed using the Adam optimizer. During the model training process, the warmup steps coefficient is set to 100, and the weight decay rate is set to 0.01. The number of pretraining iterations on the source dataset is 10, and the number of iterations on the target datasets is 100. The Micro-F1 score is used as the evaluation metric for the model.

B. EXPERIMENTAL RESULTS AND ANALYSIS

1) EXPERIMENTAL RESULTS

To demonstrate the effectiveness of the ColGAT-MP model, comparative experiments were conducted against nine baseline models categorized into three groups: semantic-based models, label-based models, and prompt-based models. word semantic-based Models: RoBERTa [27] and its domain-specific fine-tuned variant RoBERTa-DS enhance contextual semantic understanding through optimized datasets and extended training time. The MRC-NER model [28] incorporates semantic information of entities using a question-answering mechanism, enabling more effective semantic feature extraction. SpanNER and its domain-specific variant SpanNER-DS [29] focus on modeling possible entity spans in the text, avoiding the limitations of token-by-token predictions. Label semantic-based models: Proto and its domain-specific fine-tuned version Proto-DS employ a Prototypical Network framework for classification, representing each class as a prototype in the feature space. LabelSemantic model [30] integrates semantic information of labels by embedding them into the feature space, allowing the model to better understand the relationship between labels and entities. The HMP model [31] represents each class with multiple prototypes to account for intra-class semantic diversity. Utilizing the Type-Guider Prototype approach, the TaGuProto model [13] incorporates entity type hierarchy information to guide prototype generation and classification. Prompt-Based Models: the SDNet model combines contextual information and self-descriptive representations of entities to enhance prediction accuracy.

TABLE 4 shows the comparison of the micro-F1 scores of the ColGAT-MP model and the other nine baseline models on 5-shot settings across the 6 datasets. The optimal result for each dataset is marked in bold.

The comparative experimental outcomes underscore the superiority of the ColGAT-MP model over various baseline methods across a majority of datasets, establishing it as a state-of-the-art approach in the NER. This comparative analysis not only validates the model's efficacy but also sheds light on its limitations. While the ColGAT-MP model exhibits a marginal decline in performance compared to SDNet and TaGuProto on the Re3d dataset, with deficits

TABLE 4. Statistics of the few-shot NER datasets.

	CoNLL	WNUT17	Res	Movie1	Movie2	Re3d
RoBERTa	53.5	25.7	48.7	51.3	44.1	32.2
RoBERTa-DS	61.4	34.2	49.1	53.1	46.8	33.0
Proto	58.4	29.5	44.1	38.0	31.5	28.7
Proto-DS	60.9	35.9	48.4	43.8	33.5	29.2
MRC-NER	28.5	40.0	43.1	/	58.7	/
SpanNER	71.1	25.8	49.1	/	65.4	/
SpanNER-DS	75.6	38.5	51.2	/	67.8	/
LabelSemantic	78.0	35.9	55.4	53.3	73.0	47.6
SDNet	71.4	44.1	60.7	61.3	67.8	65.4
HMP	/	34.2	/	/	/	60.5
TaGuProto	79.4	39.1	61.2	**63.6**	72.3	**66.5**
ColGAT-MP	**80.8**	**50.1**	**61.9**	63.4	**75.4**	63.6

of approximately 1.8% and 2.9% respectively, and a negligible 0.2% decrement than TaGuProto on Movie1, a closer examination of the Re3d dataset reveals a crucial factor. Specifically, the significantly smaller test set size coupled with the sparsity of features within this set poses challenges for the model to effectively address data imbalance during sentence encoding, ultimately hindering the acquisition of sufficient discriminative features. However, on average, the ColGAT-MP model surpasses the second-best performer, TaGuProto, by a notable margin of 1.96%. Notably, on the WNUT17 dataset, the ColGAT-MP model achieves a remarkable 6.00% performance boost over SDNet, the second-ranked model, highlighting its exceptional adaptability and effectiveness. This enhanced performance can be attributed primarily to the graph-structured sentence encoding strategy employed by ColGAT, which facilitates the layer-wise encoding of three collaborative graphs. This approach not only enhances the model's stability and efficiency but also introduces a novel dimension to the data representation. By transforming one-dimensional sentence sequences into two-dimensional non-Euclidean graphs, the proposed collaborative graph construction method effectively augments the dataset, leveraging the increased dimensionality to further bolster the model's capabilities. Crucially, the ColGAT-MP model demonstrates that harnessing the deep structural information inherent in sentences, from a syntactic perspective, can significantly mitigate the challenges posed by sparse training samples. This finding underscores the potential of structural analysis in advancing NER models, particularly in low-resource scenarios.

2) ANALYSIS OF COLLABORATIVE GRAPH

The collaborative graph is a method of data augmentation by transforming the text into different forms. It converts the sentence into three distinct graph structures according to the rules, thereby uncovering deeper semantic features of the sentences. To rigorously evaluate the contribution of each collaborative graph, we conduct an ablation study where the graph attention network layer is modified by systematically eliminating one of the collaborative graph layers at a time, with the results summarized in TABLE 5.

Our comparative analysis of the experimental outcomes reveals that the absence of any of the collaborative graph layers adversely affects the model's overall performance, albeit to varying degrees. This underscores the efficacy of the collaborative graph transformation strategy as a data augmentation technique, effectively broadening the representational capacity of the input sentences. Notably, the model's peak performance with the full complement of collaborative graph layers underscores the paramount importance of capturing the multifaceted structural and semantic intricacies inherent in the diverse graph views. Specifically, upon the exclusion of the C-graph layer, the model experiences a pronounced performance drop across most datasets, with the decline being particularly acute on the CoNLL 2003 benchmark. Conversely, the absence of either the L-graph or the T-graph layer elicits a relatively more muted impact on the model's performance. This observation underscores the pivotal role of the C-graph within the ColGAT-MP framework, empowering the model to acquire more holistic and discriminative representations crucial for few-shot NER tasks. By incorporating information integration from both the global and local perspectives, the model is able to adeptly handle the complex interplay of multiple factors within sentences, fostering enhanced performance.

3) ANALYSIS OF THE COLLABORATIVE GRAPH ATTENTION NETWORK STRUCTURE

The ColGAT framework incorporates a collaborative graph strategy, meticulously transforming the original sentences into three distinct graph representations to enhance the depth of captured sentence semantics. These three graph layers are then fed into the GAT via two principal approaches:

Parallel Graph Structure Training (Method_1): In this method, each of the three graphs is simultaneously processed by a dedicated, separate GAT instance. This approach enables parallel processing but potentially requires a higher parameter overhead.

Serial Graph Structure Training (Method_2): Alternatively, the graphs are sequentially fed into a single GAT, with each graph serving as an additional layer of context. This serial input scheme allows for the

TABLE 5. Statistics of the few-shot NER datasets.

	CoNLL	WNUT17	Res	Movie1	Movie2	Re3d
ColGAT (without L-graph)	80.1	50.2	54.8	61.8	75.4	67.2
ColGAT (without C-graph)	76.8	48.1	51.2	60.1	74.1	63.9
ColGAT (without T-graph)	79.1	50.1	55.3	61.9	74.2	65.9
ColGAT	80.8	50.1	55.9	62.1	75.4	67.6

TABLE 6. Statistics of the few-shot NER datasets.

	CoNLL	WNUT17	Res	Movie1	Movie2	Re3d
Method_1	80.6	48.5	62.2	63.0	74.8	63.6
Method_2	80.8	50.1	61.9	62.1	75.4	63.6

integration of multi-faceted information within a single GAT, promoting parameter efficiency.

As depicted in TABLE 6, a comparative analysis of these two input strategies reveals nuanced differences in performance across various metrics evaluated for the few-shot NER task. Notably, Method_2, wherein the collaborative graphs are sequentially introduced to the GAT, effectively enables the word embeddings to grasp long-distance features embedded within the different graph structures. When juxtaposed with Method_1, which employs separate GATs for each graph, Method_2 exhibits a clear advantage in terms of parameter efficiency. This efficiency stems from the reduced complexity of a single, three-layer GAT compared to three individual, two-layer GATs. This analysis underscores the ability of the sequential input method to not only effectively capture multifaceted sentence representations but also achieve this in a more parameter-economical manner. Consequently, the flexibility in choosing the graph structure input method offers the ColGAT-MP model versatility in balancing performance gains against computational costs, contingent upon the specific requirements of the few-shot NER task at hand.

4) ANALYSIS OF THE MATCH PROCESSING

To delve into the pivotal role of the match processing unit in enhancing experimental outcomes, we conducted a ablation study comparing the performance of the ColGAT-MP model against a variant of ColGAT devoid of this unit. As depicted in FIGURE 6, the notable disparity in performance between the ablation variant and the ColGAT-MP model across six diverse datasets underscores the efficacy of the match processing unit in bolstering the semantic interplay between the support and query sets.

Furthermore, to provide a visual illustration of this impact, FIGURE 7 showcases heatmaps for ten randomly sampled entities and their corresponding types within the WNUT17 dataset. These heatmaps vividly exhibit the more pronounced effect of incorporating the match processing unit, manifesting in a clearer distinction between entity types and strengthened contextual associations. This ablation analysis underscores the criticality of the match processing unit in propelling the overall model towards achieving state-of-the-art performance in few-shot NER tasks.

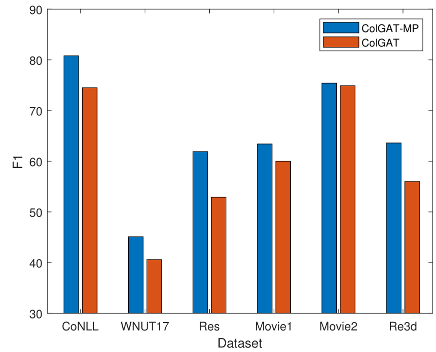


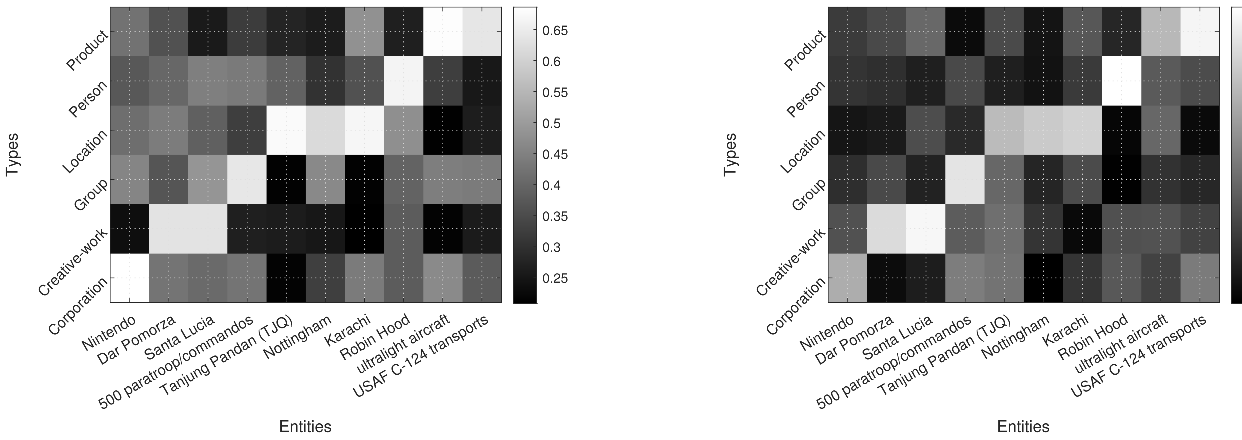
FIGURE 6. Comparison of evaluation of the ColGAT-MP and the ablation variant-ColGAT.

5) ANALYSIS OF THE MODEL RESULTS UNDER THE K-SHOT

In the meta-learning process, K signifies the quantity of samples within the Nth class set, a crucial parameter in training few-shot NER models leveraging metric-learning approaches. By augmenting the number of training samples, the model's bias can be mitigated, subsequently enhancing its accuracy and generalization capabilities. Nevertheless, in the context of small-scale datasets, the scarcity of samples frequently poses a challenge, predisposing the model to overfitting. Consequently, conducting low-shot experiments serves as a valuable means to assess the model's generalization potential, offering a more realistic portrayal of its performance in practical applications. To this end, we have devised a series of comparative experiments specifically tailored to evaluate the generalization ability of the ColGAT-MP model under various N-way k-shot settings.

The performance of the model across six datasets under varying N-way k-shot settings is presented in FIGURE 8. As the number of categories (N) increases, the trend of the performance curves tends to flatten, indicating a limited gain in indicator values. This phenomenon underscores the challenge faced by the model in processing a larger number of categories, where the complexity of the feature space to be learned escalates, potentially compromising the model's ability to discern between individual categories.

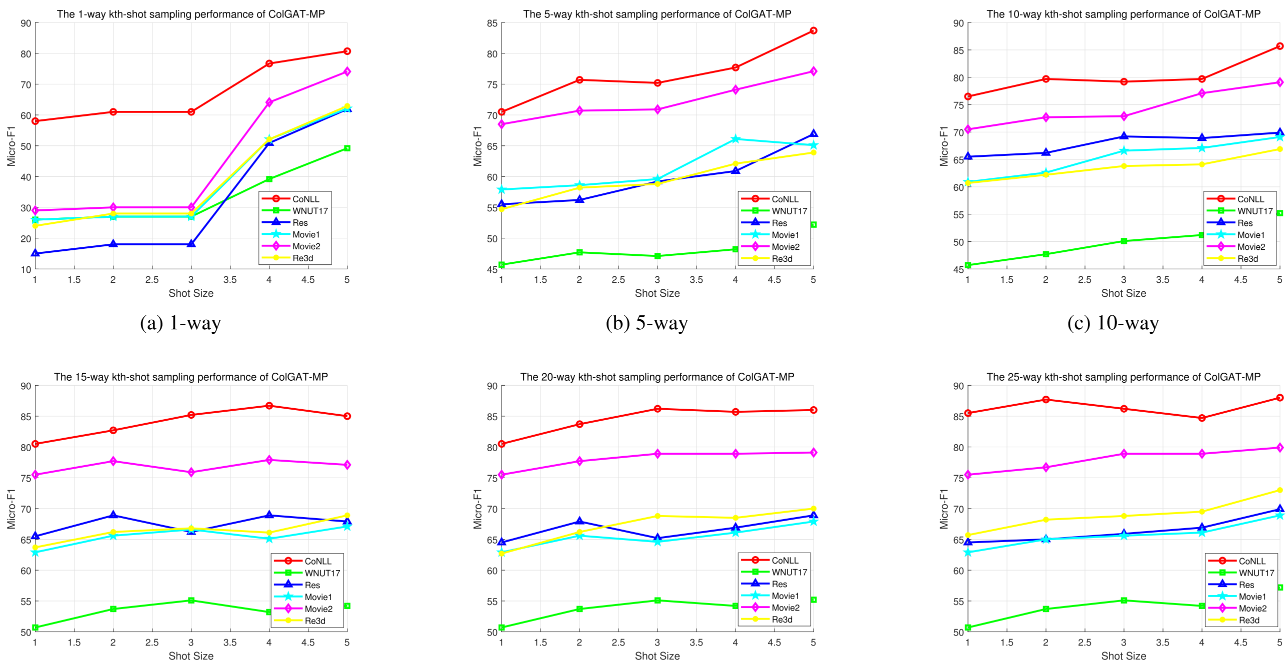
Further analysis of FIGURE 8(a) reveals a notable observation: on the CoNLL 2003 dataset, the ColGAT-MP model exhibits exceptional performance, with a relatively flat curve trend from 1-shot to 5-shot scenarios. In contrast, on the remaining datasets, the model's performance curve exhibits a pronounced upward trajectory as the shot value increases. By correlating this behavior with the characteristics of the



(a) ColGAT-MP

(b) ColGAT

FIGURE 7. Heatmaps of entities and types. (a) ColGAT-MP. (b) ColGAT.



(a) 1-way

(b) 5-way

(c) 10-way

(d) 15-way

(e) 20-way

(f) 25-way

FIGURE 8. The *k*th-shot sampling performance of ColGAT-MP. (a) 1-way. (b) 5-way. (c) 10-way. (d) 15-way. (e) 20-way. (f) 25-way.

dataset feature distributions and data scales, we conclude that the ColGAT-MP model performs particularly well in scenarios where the dataset comprises fewer categories. This underscores the robust applicability and generalization prowess of the overall framework in extreme data conditions, where resources are limited.

VII. CONCLUSION

In this paper, we have carried out in-depth explorations on few-shot named entity recognition techniques and thus constructed a ColGAT-MP model with a more efficient and accurate few-shot named entity recognition method. To obtain more knowledge to deal with the increasingly severe shortage of labeled data, we proposed a collaborative graph strategy to enhance the semantic representation of the data. To further conduct few-shot classification tasks under the condition of limited available training data, we designed an adaptive matching processing unit to strengthen the interaction between the support set and the query set. We have carried out comprehensive measurement experiments on real-world datasets, and the results show that our method can significantly improve the state-of-the-art methods.

REFERENCES

[1] S. S. S. Das, A. Katiyar, R. Passonneau, and R. Zhang, "CONTaiNER: Few-shot named entity recognition via contrastive learning," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguisticsin*, 2021, pp. 6338–6353.

[2] H. Wang, L. Cheng, W. Zhang, D. W. Soh, and L. Bing, "Order-agnostic data augmentation for few-shot named entity recognition," in *Proc. 62nd Annu. Meeting Assoc. Comput. Linguistics*, 2024, pp. 7792–7807.

[3] Y. Ge, M. A. Al-Garadi, and A. Sarker, "Data augmentation with nearest neighbor classifier for few-shot named entity recognition," *MedInfo*, vol. 310, pp. 690–694, Jan. 2024.

[4] A. Davody, D. I. Adelani, T. Kleinbauer, and D. Klakow, "TOKEN is a MASK: Few-shot named entity recognition with pre-trained language models," 2022, *arXiv:2206.07841*.

[5] H. Ming, J. Yang, F. Gui, L. Jiang, and N. An, "Few-shot nested named entity recognition," *Knowl.-Based Syst.*, vol. 293, Jun. 2024, Art. no. 111688.

[6] E. Zha, D. Zeng, M. Lin, and Y. Shen, "CEPTNER: Contrastive learning enhanced prototypical network for two-stage few-shot named entity recognition," *Knowl.-Based Syst.*, vol. 295, Jul. 2024, Art. no. 111730.

[7] Y. Yang and A. Katiyar, "Simple and effective few-shot named entity recognition with structured nearest neighbor learning," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 6365–6375.

[8] S. Zhang, B. Cao, T. Zhang, Y. Liu, and J. Fan, "Task-adaptive label dependency transfer for few-shot named entity recognition," in *Proc. Assoc. Comput. Linguistics*, Jan. 2023, pp. 3280–3293.

[9] J. Golde, F. Hamborg, and A. Akbik., "Large-scale label interpretation learning for few-shot named entity recognition," in *Proc. Assoc. Comput. Linguistics*, vol. 1, 2024, pp. 2915–2930.

[10] R. Kumar, S. Goyal, A. Verma, and V. Isahagian, "ProtoNER: Few shot incremental learning for named entity recognition using prototypical networks," 2023, *arXiv:2310.02372*.

[11] W. Xu, Y. Chen, and J. Ouyang, "A streamlined span-based factorization method for few shot named entity recognition," in *Proc. LREC/COLING*, 2024, pp. 1673–1683.

[12] D. Zhou, S. Li, Q. Chen, and H. Yao, "Improving few-shot named entity recognition via semantics induced optimal transport," *Neurocomputing*, vol. 597, Sep. 2024, Art. no. 127938.

[13] J. Feng, G. Xu, Q. Wang, Y. Yang, and L. Huang, "Note the hierarchy: Taxonomy-guided prototype for few-shot named entity recognition," *Inf. Process. Manage.*, vol. 61, no. 1, Jan. 2024, Art. no. 103557.

[14] J. Zhang, X. Liu, X. Lai, Y. Gao, S. Wang, Y. Hu, and Y. Lin, "2INER: Instructive and in-context learning on few-shot named entity recognition," in *Proc. EMNLP*, Jan. 2023, pp. 3940–3951.

[15] S. Nandi and N. Agrawal, "Improving few-shot cross-domain named entity recognition by instruction tuning a word-embedding based retrieval augmented large language model," in *Proc. EMNLP (Industry Track)*, 2024, pp. 686–696.

[16] R. Qi, J. Lou, and Y. Mao, "Label prompt guiding for two-stage few-shot named entity recognition," in *Proc. ICIC*, Jan. 2024, pp. 334–346.

[17] F. Ye, L. Huang, S. Liang, and K. Chi, "Decomposed two-stage prompt learning for few-shot named entity recognition," *Information*, vol. 14, no. 5, p. 262, Apr. 2023.

[18] J. Chen, Q. Liu, H. Lin, X. Han, and L. Sun, "Few-shot named entity recognition with self-describing networks," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2022, pp. 5711–5722.

[19] P. Chen, J. Wang, H. Lin, D. Zhao, and Z. Yang, "Few-shot biomedical named entity recognition via knowledge-guided instance generation and prompt contrastive learning," *Bioinformatics*, vol. 39, no. 8, Aug. 2023, Art. no. btad496.

[20] X. Wu, T. Zhang, S. Yuan, and Y. Yan, "One improved model of named entity recognition by combining BERT and BiLSTM-CNN for domain of Chinese railway construction," in *Proc. 7th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2022, pp. 728–732.

[21] D. Sui, Y. Chen, K. Liu, J. Zhao, and S. Liu, "Leverage lexical knowledge for Chinese named entity recognition via collaborative graph network," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3830–3840.

[22] R. Weischedel, M. Palmer, and M. Marcus, "Ontonotes release 5.0 ldc2013t19," in *Proc. Linguistic Data Consortium*, Philadelphia, PA, USA, 2013, p. 13.

[23] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in *Proc. 7th Conf. Natural Lang. Learn. HLT-NAACL*, vol. 4, 2003, pp. 142–147.

[24] L. Derczynski, E. Nichols, M. van Erp, and N. Limsopatham, "Results of the WNUT2017 shared task on novel and emerging entity recognition," in *Proc. 3rd Workshop Noisy User Generated Text*, 2017, pp. 140–147.

[25] J. Liu, P. Pasupat, S. Cyphers, and J. Glass, "Asgard: A portable architecture for multilingual dialogue systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8386–8390.

[26] J. Liu, P. Pasupat, Y. Wang, S. Cyphers, and J. Glass, "Query understanding enhanced by hierarchical parsing structures," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2013, pp. 72–77.

[27] J. Huang, C. Li, K. Subudhi, D. Jose, S. Balakrishnan, W. Chen, B. Peng, J. Gao, and J. Han, "Few-shot named entity recognition: A comprehensive study," 2020, *arXiv:2012.14978*.

[28] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu, and J. Li, "A unified MRC framework for named entity recognition," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 5849–5859.

[29] P. Wang, R. Xu, T. Liu, Q. Zhou, Y. Cao, B. Chang, and Z. Sui, "An enhanced span-based decomposition method for few-shot sequence labeling," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2021, pp. 5012–5024.

[30] T. Ma, H. Jiang, Q. Wu, T. Zhao, and C.-Y. Lin, "Decomposed meta-learning for few-shot named entity recognition," in *Proc. Assoc. Comput. Linguistics (ACL)*, 2022, pp. 1584–1596.

[31] Z. Liao, J. Fei, W. Zeng, and X. Zhao, "Few-shot named entity recognition with hybrid multi-prototype learning," *World Wide Web*, vol. 26, no. 5, pp. 2521–2544, Sep. 2023.

HAORAN NIU received the Ph.D. degree in software engineering from Yanshan University, in 2023. He is currently working as a Lecturer with Jiangsu Ocean University. His research interests include knowledge graph, natural language processing, and semantic web.

ZHAOMAN ZHONG received the Ph.D. degree in computer science from Shanghai University, in 2011. He is currently working as a Professor with Jiangsu Ocean University. His research interests include information retrieval and artificial intelligence.