BiGCAT: A Graph-Based Representation Learning Model with LLM Embeddings for Named Entity Recognition

Md. Akram Hossain[1], Abdul Aziz[1], Muhammad Anwarul Azim[1], Abu Nowshed Chy[1], Md Zia Ullah[2], and Mohammad Khairul Islam[1]

[1]Department of Computer Science and Engineering,
University of Chittagong, Chattogram-4331, Bangladesh
[2]School of Computing, Edinburgh Napier University, Edinburgh, UK
{akram.hossain.cse.cu, aziz.abdul.cu}@gmail.com,
{azim, nowshed}@cu.ac.bd, M.Ullah@napier.ac.uk, mkislam@cu.ac.bd

Abstract

Named entity recognition from financial text is challenging because of word ambiguity, huge quantity of unknown corporation names, and word abbreviation compared to nonfinancial text. However, models often treat named entities in a linear sequence fashion, which might obscure the model's ability to capture complex hierarchical relationships among the entities. In this paper, we proposed a novel named entity recognition model BiGCAT, which integrates large language model (LLM) embeddings with graph-based representation where the contextual information captured by the language model and graph representation learning can complement each other. The method builds a spanning graph with nodes representing word spans and edges weighted by LLM embeddings, optimized using a combination of graph neural networks, specifically a graph-convolutional network (GCN) and a graph-attention network (GAT). This approach effectively captures the hierarchical dependencies among the spans. Our proposed model outperformed the state-of-the-art by 10% and 18% on the two publicly available datasets FiNER-ORD and FIN, respectively, in terms of weighted F1 score. The code is available at: https://github.com/Akram1871/BiGCAT-RANLP-2025.

1 Introduction

Background. Named entity recognition (NER) is a classical information extraction problem in natural language processing. Accurately identifying named entities can benefit information access tasks such as question answering, recommendation systems, and ranking systems. With the rapid growth of E-financing activities that generate vast amounts of unstructured financial data, financial NER (FinNER) research becomes salient to understand the dynamics of the financial landscape. FinNER poses some unique challenges compared to traditional NER problems, including (1) financial texts feature complex structures with nested, long-range dependencies such as multi-word entities spanning different sentence segments, (2) large volume of organizational entities which are often used in short form and become arduous for extraction, and (3) variable length of spans and ambivalent semantic meaning of financial data (Swaileh et al., 2020; Feng et al., 2020; Sumithra and Sridhar, 2021). Although NER is an active area of research, FinNER is a less explored domain.

Problem Definition. Some studies have shown that graph–based representations can capture long-distance unstructured information via graph neural network (GNN), while LSTMs or LLMs effectively capture sequential contextual information (Cetoli et al., 2017; Xu et al., 2021; Zaratiana et al., 2022). Although integrating these features has the potential to enhance NER models, the mechanism of their interaction remains unclear, and the performance gains have been modest. Moreover, the effectiveness of this integration in the context of financial documents remains an open question. In this paper, we address this question and focus on exploring the impact of Graph neural networks on the FinNER task. We propose an integrated graph neural network model, BiGCAT, combining the benefits of two popular GNN models including graph-convolution network and graph-attention network.

Our contributions are as follows; (i) We introduce a novel model **BiGCAT** for recognizing named entities from financial documents fusing the PLMs with **B**iLSTM-**GC**N and **GAT** where we design span-based graph input representation rather than tokens. (ii) We evaluate our proposed method on two benchmark datasets (i.e., FiNER-ORD and FIN) and report the state-of-the-art performance. We compare our proposed BiGCAT with

Proceedings of Recent Advances in Natural Language Processing,pages 433–440
Varna, Sep 8–10, 2025
https://doi.org/10.26615/978-954-452-098-4-052

several baselines to demonstrate its effectiveness. (iii) To ensure reproducibility and facilitate further research, we make our code, experimental details, and a publicly available pretrained model [1] on Huggingface accessible to the community.

We organize the remaining content of this paper as follows: Section 2 includes a comprehensive background study. Section 3 introduces our proposed BiGCAT method for the FinNER task. Section 4 explains detailed experiments and evaluation along with the performance comparison with related approaches. We also provide an insightful discussion in Section 4.3. We conclude our work and draw future directions in Section 5.

2 Related Work

Feature-based methods either exploit hand-crafted lexical resources and fit them into a rule-based classifier (Farmakiotou et al., 2000; Collobert et al., 2011; Ju et al., 2018) or encode the input text using Neural network architectures like BiLSTM and then pass the embeddings to the conditional random field (CRF) (Sutton et al., 2004) to capture the dependencies between the adjacent labels (e.g., whether "New" and "York" should be labeled together as a single entity like "New York") using the sequence transition probabilities of the labels (Zhao et al., 2018). However, these models heavily depend on feature engineering that hampers the robustness and fails to model complex contexts like the financial domain.

These methods utilized transformers in two ways. In one approach, open domain pre-trained language models (PLMs) such as BERT, RoBERTa, and GPT4. are employed on FinNER task (Li et al., 2023). Another method introduces the BiLSTM layer after the Transformer to create contextual representations of words in the input sequence (Wang et al., 2022). The other focused on the fine-tuned PLMs on financial documents like FinBERT (Huang et al., 2023) to get better performance on FinNER tasks. Transformer-based models are superior to the sequence-based methods but still suffer in short text and domain-specific complex context problems. For this reason, their contextual representations need to pass an effective deep neural network (DNN) to compare the semantic relations for extracting entities (Gupta et al., 2023).

Graph neural network (GNN) based methods have gained attention in financial NER tasks. Guo et al. (2020) proposed a knowledge graph embedded approach by integrating the GNN and Transformer that performs well for the financial NER task. Zaratiana et al. (2022) proposed a GNN-based method to reduce overlapping spans during training that shows promising performance. This work motivates us to incorporate GNN methods after the Transformer embeddings for the financial named entity extraction task.

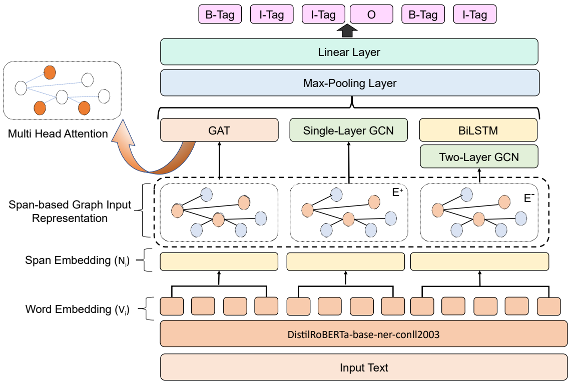


Figure 1: An overview of our proposed BiGCAT framework. The input text is tokenized and represented using the pre-trained transformer. We generate the representation of span from its token representation, construct a span-based homogeneous interval graph (**E**), and split it into two variants **E**$^+$ and **E**$^-$ based on their value of edge.

3 Proposed BiGCAT Framework

The overview of our proposed BiGCAT framework is depicted in Figure 1. It consists of three stages, including contextual modeling for input representation, GNN layers, and integration of multiple GNN layers to predict the label of tokens.

3.1 Contextual Modeling

We modeled our contextual input representation into two phases. In the first phase, we exploit a pre-trained DistilRoBERTa-base [2] (Sanh et al., 2019) Transformer model for word-based contextual representation to learn low-level features from the text. For example, given an input sentence x, we tokenized it into n words $\{t_1, t_2, t_3, \cdots, t_n\}$ and pass it to the transformer layer and exploit sequential embedding vector of each token $\{h^1, h^2, h^3, \cdots, h^n\}$.

[1] https://huggingface.co/mrx71/BiGCAT

[2] https://huggingface.co/philschmid/distilroberta-base-ner-conll2003

Prior studies found that the first subtoken representation of the transformer model works better than the other for token classification tasks; therefore, we use the first subtoken representation (Kenton and Toutanova, 2019). In the second phase, following the sliding window mechanism we build the spans over the tokens and represent a span using the contextual representation of its token from the first phase. Inspired by Zaratiana et al. (2022), we employ this span representation to reduce the overlapping span problem. It also facilitates us in designing graph neural network models to exploit long-range nested dependency of entities, which is crucial for FinNER tasks.

We use the AllenNLP library (Gardner et al., 2017) to establish end-to-end represention between the spans and the embeddings. We build k length of spans over the tokens, such as $s_{i,j} = Concat\{h^i, h^j, e_k\}$. Where $s_{i,j}$ denotes the span representation of index i and j. h_i and h_j denote the endpoints word representation of those indices, and e^k the corresponding embedding vector of span width of k.

3.1.1 Graph Structure

We construct an undirected weighted graph, $G = (S, E)$, with $S = \{s_1, s_2, \cdots, s_m\}$ as the set of spans and $E \in S \times S$ as the set of edges. The neighbor set of span s is denoted as $N(s) = \{u : (s, u) \in E\}$. The span features are represented by a feature matrix $X \in R^{|S| \times d}$, where the i-th row $x_i \in R^d$ is the feature vector of span s_i and d is the dimension of the feature. The weight of an edge between two spans s_i and s_j, $a_{i,j}$ is defined as follows:

$$a_{i,j} = \begin{cases} 1, & if\ s_i = s_j \\ 0, & if\ |s_i \cap s_j| \\ -1 & otherwise \end{cases}$$

To reduce the span overlap, we use contextual representation where the weight of edge 1, 0, and -1 denote self-connected span, non-overlapping span, and overlapping span, respectively. It avoids the recalculation of candidates and helps to learn better semantics among the tokens, especially between the neighbor tokens.

3.2 Graph Neural Network Framework

Our graph neural network (GNN) model combines two popular GNN architectures, graph convolution network (GCN) and graph attention network (GAT) (see Figure 1). We feed the input graph to a GAT, a single-layer GCN (SLGCN), and a two-layer GCN network (TLGCN). Moreover, we also plug a Bi-LSTM layer on top of the TLGCN. We concatenate all layer's outputs and pass them to the Max-pooling, Dropout, and Linear layers for predicting the label of the token.

3.2.1 Graph-Convolution Network (GCN)

We utilize vanilla GCN (Kipf and Welling, 2016) as one of the backbone components of our proposed BiGCAT method to extract global word-word relationships from tokens. Then split the span-based graph into two signed graphs, based on the positive edge E^+ and negative edge E^-, then pass them to SLGCN and TLGCN, respectively. We consider two intuitions for employing TLGCN on E^- graph, i) following 3.1.1 negative weight edges represent the dissimilar spans; GCN weight propagation mechanism considers positive edges as a similar neighbour and often ignore negative edges which are considered as dissimilar neighbours due to its bias nature, but in our case, we only pass E^- graph and it contains salient node features which can help GCN aggregation mechanics to exploit better semantic features, ii) the TLGCN outperforms in several graph classification tasks compared to the SLGCN (Malekzadeh et al., 2021; Hanh et al., 2021). The edges of the positive signed graph are self-connected; hence, we utilize SLGCN from the intuition that it can learn local neighbourhood features better than TLGCN. For the negative edge E^- span, we compute the propagation matrix of GCN as follows:

$$\begin{aligned} H^{(l+1)} &= \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}) \\ \tilde{A} &= A + I_M \end{aligned} \tag{1}$$

where A is the adjacency matrix of constructed graph, G defined in Section 3.2.1 and I_M denotes an identity matrix of M spans. Here, D is the degree matrix, $\tilde{D}_i = \sum_j A_{ij}$, the trainable weighted matrix of each layer denoted as $W^{(l)}$. Nevertheless, $\sigma(.)$ denote the activation function and $H^{(l)}$ denotes the activation matrix of l^{th} layer where $H^{(0)}$ initialize by the input span representation.

3.2.2 Graph-Attention Network (GAT)

We utilize the GAT network (Veličković et al., 2017) but exploit the multi-head attention mechanism (Vaswani et al., 2017) since it can learn better semantic contextual representation. For the self-attention, we compute our queries, keys, and values packed into the matrix **Q**, **K**, and **V**, respectively, where we extract those matrices from span representation using a two-layer feedforward network. We compute the attention for every single span s_i as follows:

$$f_{att}(Q, K, V) = (\frac{QK^T}{\sqrt{d_{model}}} \odot A)V \quad (2)$$

$$H' = Concat(S_1, S_2, ...S_r)W^0 \quad (3)$$

$$S_i = f_{att}(QW_i^Q, KW_i^K, VW_i^V) \quad (4)$$

where $\odot$ denotes the element-wise dot operation between the matrices. In the above equations 3 and 4, we present a representation of our estimated multi-head attention over all spans in a sentence. Here, H' denoted the final calculated attention of a sentence. The projections of trainable weight matrices are $W_i^Q \epsilon R^{d_{model}\ \times\ d_q}$, $W_i^K \epsilon R^{d_{model}\ \times\ d_k}$, $W_i^V \epsilon R^{d_{model}\ \times\ d_v}$, and $W^O \epsilon R^{rd_v\ \times\ d_{model}}$ where $r = m$ denotes the number of spans in the sentence.

3.2.3 BiLSTM

Bidirectional long short-term memory (BiLSTM) (Brueckner and Schulter, 2014), an extended version of recurrent neural network. Individual GCN modules are limited to learning sequential dependency information from the text because it does not consider word order (Malekzadeh et al., 2021). We employ a BiLSTM module over a two-layer GCN to overcome this shortfall and extract the long-term inter-relational dependencies among the tokens.

3.3 Integration of GNN Layers

To integrate the GAT, single-layer GCN (SLGCN), and two-layer GCN (TLGCN) with BiLSTM layers, we utilize the early fusion technique (Ebersbach et al., 2017). We fuse them using $Concat(R^{d_{ew} \times d_p}, S_1^{d_{ew} \times d_p}, S_2^{d_{ew} \times d_p})$ where R, S_1, and S_2 denote the outputs feature matrix from GAT, SLGCN, and TLGCN layer, respectively. d_{ew} denotes the maximum embedding width, and d_p is the projection embedding dimension in our task.

4 Experiments

In this section, we report our experimental evaluation by first describing the experimental settings and then presenting and discussing the results.

4.1 Experimental Setup

Dataset. We use two benchmarks datasets: FiNER-ORD (Shah et al., 2023) and FIN (Alvarado et al., 2015). FiNER-ORD dataset contains a total of 201 articles that are collected from financial news. However, it has a total of 1,16,721 tokens where the percentages of the entities of location (LOC), organization (ORG) and person (PER) are 1. 25%, 2. 44% and 1. 10%, respectively. On the other hand, the FIN dataset contains an additional entity, miscellaneous (MISC). Alvarado et al. (2015) utilized eight publicly available loan agreements to build this corpus where they randomly split them in 5: 3 and used these five agreements for the train split, namely FIN5 and the rest three for the test split, namely FIN3. FIN5 and FIN3 contain a total of 1014, 336,306,686, 4 and 150, 79, 114, 236, 7 articles, LOC, ORG, PER, and MISC tag, respectively. We considered various standard evaluation measures including F1-score (Chinchor and Sundheim, 1993), Precision, and Recall (Powers, 2011) to validate our proposed system performance where the primary evaluation metric is the Weighted average of F1-scores (W-F1) for the FiNER-ORD and FIN dataset, following Shah et al. (2023).

We used pre-trained DistilRoBERTa-base language model from the HuggingFace library (Wolf et al., 2019) and used 768-dimensional embedding features. We packed our word's input representation into spans utilizing the *max_span_length* is 8, and span embedding width is 128 (section 4.3.2). Later, for the GNN layer, features were down-projected from (768*2 + 128) to 256. To calculate the overall loss of the model, we utilize a cross-entropy loss algorithm (De Boer et al., 2005). We set the learning rate 2.3e-5, batch size 16, and Adam optimizer (Kingma and Ba, 2014) to learn the weights and train our proposed BiGCAT model. However, we set the epoch numbers for FiNER-ORD and FIN datasets is 30. We searched the best hyperparameters using Weight&Bias (Biewald, 2020) sweep configuration and reported the average results from three seeds: 5, 42, and 1871.

4.2 Experimental Results

We compare our proposed BiGCAT model with the following baselines: *SpacyNER* (Honnibal and Montani, 2017), *GLiNER* (Zaratiana et al., 2023), *Llama-3.1* (Llama Team, AI @ Meta, 2024), *GPT-4o* (OpenAI, 2023), *BERT* (Devlin et al., 2019), *FinBERT* (Araci, 2019), and *RoBERTa* (Liu et al.,

Table 1: Comparative performance of our proposed BiGCAT with related baselines on the FiNER-ORD test split. The models with * symbol are retrieved from (Shah et al., 2023). The best result is highlighted in bold, and the second best is underlined.

Model	PER	LOC	ORG	Weighted Average
BiGCAT	**.9429**	**.8495**	**.7912**	**.8400**
Performance of the fine-tuned PLMs baselines				
RoBERTa-large*	.9263	.7717	.6769	.7648
BERT-large-cased*	.8954	.7289	.6272	.7216
BERT-base-cased*	.8811	.6820	.6013	.6931
FinBERT-base-cased*	.7456	.6836	.6002	.6589
RoBERTa-base*	.9050	.7154	.6304	.7220
Performance of the zero-shot baselines				
SpacyNER	.7824	.1714	.6134	.5499
GLiNER	.6884	.6208	.5841	.6159
Llama-3.1-70B-Turbo*	.6706	.5919	.4981	.5661
Llama-3.1-405B-Turbo*	.7413	.6587	.5751	.6389
GPT-4o*	.8004	.6651	.6036	.6692

Table 2: Comparative performance of our proposed BiGCAT with related baselines on the FIN test split. The best result is highlighted in bold, and the second best is underlined.

Model	PER	LOC	ORG	MISC	Weighted Average
BiGCAT	**.9489**	.3011	**.5217**	.0000	**.6989**
Performance of the fine-tuned PLMs baselines					
BERT-base-cased	.7028	.0241	.0342	.0000	.3929
DistilRoBERTa-base	.8258	.4056	.2500	.0000	.5899
Performance of the zero-shot baselines					
SpacyNER	.6104	.3310	.1477	.0000	.4290
GLiNER	.4864	**.6709**	.3142	.0000	.4670

2019). In Table 1, we presented the comperative study our BiGCAT method of FiNER-ORD in terms of weighted F1 score (W-F1) following Shah et al. (Shah et al., 2023). Our proposed BiGCAT model outperforms the baseline methods across each entity, improving the W-F1 score by approximately 10% compared to the second-best performing system, the fine-tuned RoBERTa-large. Fine-tuned PLMs show dominance in performance compared to zero-shot models due to domain-specific training. As discussed in Section 1, FinNER contains several syntactical and semantical unique challenges, which require more financial knowledge for the models. We fine-tuned the hybrid GNN layers over a financial corpus, which helped the model learn financial contextual cues more effectively.

In Table 2, we compared the performance of our BiGCAT model with several other models, including SpacyNER and GLiNER as zero-shot baselines, and BERT and DistilRoBERTa as fine-tuned PLMs baseline models. The experimental results show that our method consistently outperforms all Transformer models across entities except for LOC. However, all models failed to identify the MISC entity due to its very low density in both the training and test splits of the FIN (Alvarado et al., 2015) dataset. The test dataset contains only 7 MISC entities, and the validation dataset does not contain a single MISC entity, which led the models to underfit on this entity. Our BiGCAT model improved over the state-of-the-art baselines as follows: BERT: 78%, DistilRoBERTa: 18%, SpacyNER: 63%, GLiNER: 50% in terms of W-F1. This consistent performance justifies the inclusion of the GNN method for the financial named entity recognition task.

Although our evaluation was limited to the FinNER task, our model may also perform well in general NER tasks, as it outperformed the SOTA GLiNER model, which had previously outperformed 20 NER benchmarks.

4.3 Empirical Analysis

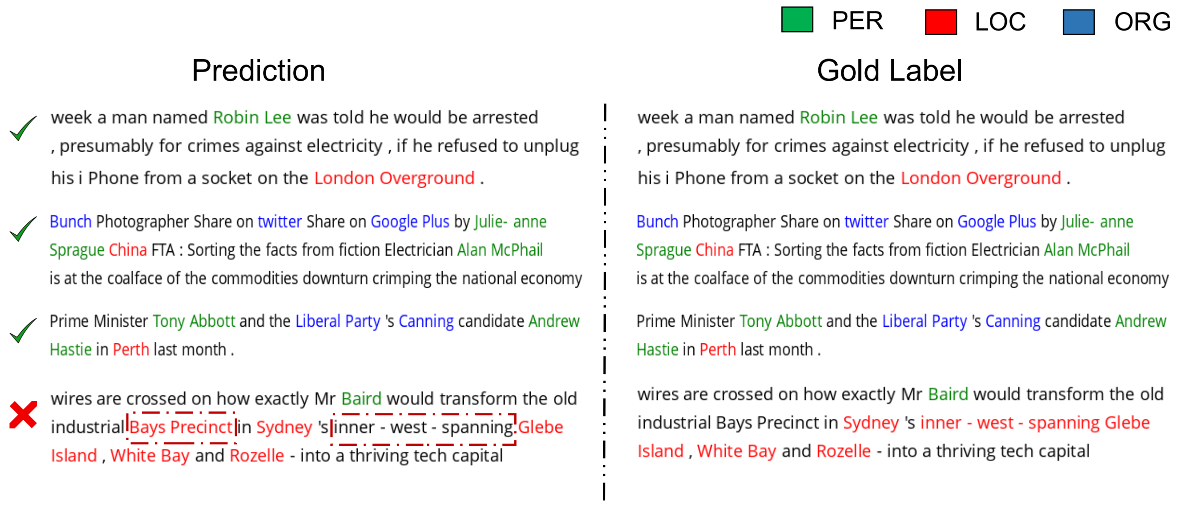
In this section, we perform qualitative and quantitative analyses, including the counting of overlapping spans, error analysis, and an ablation study to evaluate the robustness of the proposed BiGCAT model for named entity recognition.

4.3.1 Ablation Study

We conducted an ablation study and discussed the individual component's performance of our proposed BiGCAT method, as illustrated in Table 3. The comparative performance of our system against other settings validates the effectiveness of the integration approach of our model. However, the *DistilRoBERTa+*$GCN_{+BiLSTM}$ performance compared to *DistilRoBERTa+GAT* in Table 3 shows that it achieves lower precision but higher recall, indicating that the $GCN_{+BiLSTM}$ module often makes false positive errors than the GAT module. Contrarily, *DistilRoBERTa+GAT* model improves precision but reduces recall, making the model more cautious and leading to missed correct entities. This validates our fusion strategy of the GNN modules, and BiGCAT proves to be the most effective balanced approach for this task.

4.3.2 Overlapping Spans

The number of overlapping spans is one of the key parameters that indicate how well a GNN model is on the NER task because overlapping spans may conflict in prediction and provide different labels for a single token. We analyze the relationships be-



week a man named Robin Lee was told he would be arrested , presumably for crimes against electricity , if he refused to unplug his i Phone from a socket on the London Overground .

Bunch Photographer Share on twitter Share on Google Plus by Julie- anne Sprague China FTA : Sorting the facts from fiction Electrician Alan McPhail is at the coalface of the commodities downturn crimping the national economy

Prime Minister Tony Abbott and the Liberal Party 's Canning candidate Andrew Hastie in Perth last month .

wires are crossed on how exactly Mr Baird would transform the old industrial Bays Precinct in Sydney 's inner - west - spanning Glebe Island , White Bay and Rozelle - into a thriving tech capital

week a man named Robin Lee was told he would be arrested , presumably for crimes against electricity , if he refused to unplug his i Phone from a socket on the London Overground .

Bunch Photographer Share on twitter Share on Google Plus by Julie- anne Sprague China FTA : Sorting the facts from fiction Electrician Alan McPhail is at the coalface of the commodities downturn crimping the national economy

Prime Minister Tony Abbott and the Liberal Party 's Canning candidate Andrew Hastie in Perth last month .

wires are crossed on how exactly Mr Baird would transform the old industrial Bays Precinct in Sydney 's inner - west - spanning Glebe Island , White Bay and Rozelle - into a thriving tech capital

Figure 2: Some successful and unsuccessful examples of our proposed BiGCAT from the FiNER-ORD test set. Here, the green, red, and blue colored tokens represent the person, location, and organization entities, respectively.

Table 3: Ablation study of our proposed BiGCAT model in its different variants: results on the validation split of Finer-ORD.

Model	W-F1	Prec.	Rec.
BiGCAT	**.9220**	**.9182**	**.9220**
$DistilRoBERTa + GAT$	.8916	.8972	.8883
$DistilRoBERTa + GCN_{+BiLSTM}$	.8857	.8776	.8949

tween the number of overlapping spans, maximum span length (max_span), and weighted F1 scores of the top three BiGCAT runs at max_span values of 4, 8, and 16. Our proposed BiGCAT model has the lowest number of overlapping spans when max_span = 4, which is close to 0. However, its w-F1 score on the FiNER-ORD test set is 0.7972, indicating that the model may be overfitted. For max_span = 8, we observe that the number of overlapping spans is below the average line, so we used this for training the rest of our model settings. The results are shown in Table 1, indicating how well our model fits this task.

4.3.3 Qualitative Results

In Figure 2, we illustrated an insightful qualitative analysis of our BiGCAT model using the FiNER-ORD test dataset. The first three examples were the successful ones where our method identified all the entities correctly, especially when token's length are relatively long, which is crucial for FinNER task. The last example of Figure 2 show that our model sometimes misidentified long multi-word entities because of LLM's sub-tokenization and the imbalanced training dataset contains almost 92.5% of "O (other)" tokens.

5 Conclusion

In this paper, we propose BiGCAT for named entity recognition, which integrates graph-based representation learning for the first time specifically in the financial domain. First, we construct a span graph from the input text, weighting the graph with embeddings from LLMs to capture sequential context among the words of the span. Next, we develop a multi-layered graph neural network (GNN) framework that utilizes graph-convolutional networks and graph-attention networks to exploit global semantic dependencies among spans. Finally, we pass the GNN outputs through a max-pooling layer and a dense layer to obtain token-label predictions.

Results of experiments conducted on the FiNER-ORD and FIN datasets demonstrate that the proposed model achieves state-of-the-art performance in the primary evaluation metric and in the F1 score for each individual entity, compared to baseline models. This indicates that span-based representations are more suitable for the NER task than token-based representations. Additionally, we provide a detailed qualitative and quantitative analysis of the model, including error analysis, the rate of overlapping spans, and an ablation study of different settings. In our future work, we plan to investigate the use of heterogeneous graph representations and graph embeddings to develop more robust and lightweight models for named entity recognition.

References

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *ArXiv*, abs/1908.10063.

Lukas Biewald. 2020. Experiment tracking with weights and biases. Software available from wandb.com.

Raymond Brueckner and Björn Schulter. 2014. Social signal classification using deep blstm recurrent neural networks. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4823–4827. IEEE.

Alberto Cetoli, Stefano Bragaglia, Andrew D O'Harney, and Marc Sloan. 2017. Graph convolutional networks for named entity recognition. *arXiv preprint arXiv:1709.10053*.

Nancy Chinchor and Beth M Sundheim. 1993. Muc-5 evaluation metrics. In *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537.

Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. 2005. A tutorial on the cross-entropy method. *Annals of operations research*, 134:19–67.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL:HLT)*, pages 4171–4186.

Mike Ebersbach, Robert Herms, and Maximilian Eibl. 2017. Fusion methods for icd10 code classification of death certificates in multilingual corpora. In *CLEF (Working Notes)*, page 36.

Dimitra Farmakiotou, Vangelis Karkaletsis, John Koutsias, George Sigletos, Constantine D. Spyropoulos, and Panagiotis Stamatopoulos. 2000. Rule-based named entity recognition for greek financial texts.

Fuli Feng, Cheng Luo, Xiangnan He, Y. Liu, and Tat-Seng Chua. 2020. Finir 2020: The first workshop on information retrieval in finance. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Kaihao Guo, Tianpei Jiang, and Haipeng Zhang. 2020. Knowledge graph enhanced event extraction in financial documents. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 1322–1329. IEEE.

Shashank Gupta, Mohamed Reda Bouadjenek, and Antonio Robles-Kelly. 2023. A mask-based logic rules dissemination method for sentiment classifiers. In *European Conference on Information Retrieval*, pages 394–408. Springer.

Tran Thi Hong Hanh, Antoine Doucet, Nicolas Sidere, Jose G Moreno, and Senja Pollak. 2021. Named entity recognition architecture combining contextual and global features. In *Towards Open and Trustworthy Digital Societies: 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1–3, 2021, Proceedings*, pages 264–276. Springer.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Allen H Huang, Hui Wang, and Yi Yang. 2023. Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*, 40(2):806–841.

Meizhi Ju, Makoto Miwa, and Sophia Ananiadou. 2018. A neural layered model for nested named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1446–1459.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Xianzhi Li, Xiao-Dan Zhu, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2023. Are chatgpt and gpt-4 general-purpose solvers for financial text analytics? an examination on several typical tasks. *ArXiv*, abs/2305.05862.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019.

RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv e-prints*, pages arXiv–1907.

Llama Team, AI @ Meta. 2024. The llama 3 herd of models. *ArXiv*, abs/2407.21783.

Masoud Malekzadeh, Parisa Hajibabaee, Maryam Heidari, Samira Zad, Ozlem Uzuner, and James H Jones. 2021. Review of graph neural network in text classification. In *2021 IEEE 12th annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*, pages 0084–0091. IEEE.

OpenAI. 2023. Gpt-4 technical report.

David M. W. Powers. 2011. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *ArXiv*, abs/2010.16061.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Agam Shah, Ruchit Vithani, Abhinav Gullapalli, and Sudheer Chava. 2023. Finer: Financial named entity recognition dataset and weak-supervision model. *arXiv preprint arXiv:2302.11157*.

M Kani Sumithra and Rajeswari Sridhar. 2021. Information retrieval in financial documents. In *Evolving Technologies for Computing, Communication and Smart World: Proceedings of ETCCS 2020*, pages 265–274. Springer.

Charles Sutton, Khashayar Rohanimanesh, and Andrew McCallum. 2004. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *Proceedings of the twenty-first international conference on Machine learning*, page 99.

Wassim Swaileh, Thierry Paquet, Sébastien Adam, and Andres Rojas Camacho. 2020. A named entity extraction system for historical financial data. In *Document Analysis Systems: 14th IAPR International Workshop, DAS 2020, Wuhan, China, July 26–29, 2020, Proceedings 14*, pages 324–340. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Xiaoguo Wang, Yanning Sun, Chao-Yu Chen, and Jianwen Cui. 2022. A relation extraction model based on bert model in the financial regulation field. *2022 2nd International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI)*, pages 496–501.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Lu Xu, Zhanming Jie, Wei Lu, and Lidong Bing. 2021. Better Feature Integration for Named Entity Recognition. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3457–3469, Online. Association for Computational Linguistics.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2022. Gnner: Reducing overlapping in span-based ner using graph neural networks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 97–103.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2023. Gliner: Generalist model for named entity recognition using bidirectional transformer. *ArXiv*, abs/2311.08526.

Huasha Zhao, Yi Yang, Qiong Zhang, and Luo Si. 2018. Improve neural entity recognition via multi-task data selection and constrained decoding. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 346–351.