Financial Named Entity Recognition: How Far Can LLM Go?

Yi-Te Lu[1] **and Yintong Huo**[2]

[1] National Taiwan University, Taiwan
[2] Singapore Management University, Singapore

Correspondence: b08901016@ntu.edu.tw

Abstract

The surge of large language models (LLMs) has revolutionized the extraction and analysis of crucial information from a growing volume of financial statements, announcements, and business news. Recognition for named entities to construct structured data poses a significant challenge in analyzing financial documents and is a foundational task for intelligent financial analytics. However, how effective are these generic LLMs and their performance under various prompts are yet need a better understanding. To fill in the blank, we present a systematic evaluation of state-of-the-art LLMs and prompting methods in the financial Named Entity Recognition (NER) problem. Specifically, our experimental results highlight their strengths and limitations, identify five representative failure types, and provide insights into their potential and challenges for domain-specific tasks.

1 Introduction

As an increasing amount of information is contained within documents and text available online, utilizing a series of natural language processing (NLP) techniques to automate the process of extracting meaningful information from unstructured text has become a critical task, especially in the financial domain (Ashtiani and Raahemi, 2023). Among all, named entity recognition (NER) serves as a foundational first step in identifying key entities, such as persons, organizations, and locations, enabling the construction of knowledge graphs and other applications.

With the surge of large language models (LLMs), LLMs have demonstrated transformative capabilities in generative tasks, leveraging reinforcement learning from human feedback (RLHF) (Christiano et al., 2017). LLMs achieve remarkable performance across a wide range of NLP tasks with minimal adaptation (Qin et al., 2024). However, their ability to perform domain-specific tasks, such as NER in the financial domain, remains less explored. For instance, in the sentence *"Johnson Brothers rethink plan for St. Paul waterfront Shepard Road Development."*, a generic NER model might incorrectly classify the company *"Johnson Brothers"*as a person. This understanding is critical, as it could influence numerous applications in finance.

In this paper, we aim to evaluate the capabilities of state-of-the-art LLMs in performing NER tasks within the financial domain, their response to various prompt types, and their limitations in this context. To achieve this, we conduct a systematic analysis and present experimental results, comparing the effectiveness of leading LLMs with recent fine-tuned approaches. Specifically, we evaluate three advanced LLMs with different parameter sizes, GPT-4o (OpenAI, 2024), LLaMA-3.1 (Dubey et al., 2024), and Gemini-1.5 (Google, 2024)—under three distinct prompting techniques: direct prompting, in-context learning, and chain-of-thought (CoT) prompting. We perform our study by investigating the following two research questions (RQs):

- **RQ1**: How do different LLMs perform in NER tasks under various prompts?
- **RQ2**: What types of mistakes do LLMs commonly make?

To sum up, the main contributions of this paper are as follows:

- To the best of our knowledge, this is the first study to comprehensively compare state-of-the-art generically trained LLMs on NER tasks in the financial domain.
- We analyze LLM performance across three distinct prompting techniques, identify their limitations, categorize five representative types of failures and underlying causes, and elicit two future directions based on our findings.

Proceedings of the Joint Workshop of the 9th FinNLP, the 6th FNP, and the 1st LLMFinLegal, pages 164–168
January 19–20, 2025.

2 Related Work

2.1 Large Language Models in Finance

LLMs have recently been applied to finance, particularly in automatic information retrieval and financial analysis (Li et al., 2023b). Li et al., 2023a empirically explore ChatGPT and GPT-4's capabilities in analyzing financial texts and compare them to state-of-the-art fine-tuned models. However, existing research mainly focuses on fine-tuned finance LLMs or individual generic LLMs, lacking comparisons of their performance under various prompt designs. This paper addresses this gap by providing a comprehensive evaluation of state-of-the-art LLMs under various prompting styles in the context of financial NER tasks.

3 Study Setup

To understand current LLMs' capabilities in handling financial NER problems, we choose three state-of-the-art LLMs, each with three popular prompting strategies. We further select two representative transformer-based models and fine-tune them on financial data for comparison.

3.1 Financial NER Datasets

In this study, we use the FiNER-ORD dataset (Shah et al., 2023) as our benchmark. While the CRA NER dataset (Alvarado et al., 2015), based on financial agreements from the SEC, is widely used for research (Li et al., 2023a) and includes four entity types (person/PER, location/LOC, organization/ORG, and miscellaneous/MISC), it suffers from a skewed distribution of entity types and limited source of data.

FiNER-ORD resolves this imbalance and removes the ambiguous miscellaneous category, consisting of a manually annotated dataset of 201 financial news articles. This provides a more robust and high-quality benchmark for financial NER tasks and has been adopted in recent research (Xie et al., 2024). As reported by Shah et al., 2023, the entity ratio in FiNER-ORD for ORG, LOC, and PER is 2.29:1.17:1, compared to the heavily skewed ratio of 0.31:0.22:1 in the CRA dataset.

3.2 Models

We evaluate three state-of-the-art LLMs and their lightweight versions on the FiNER-ORD task: GPT-4o, GPT-4o-mini (OpenAI, 2024), LLaMA-3.1-70B-Instruct, LLaMA-3.1-8B-Instruct, Gemini-1.5-flash, and Gemini-1.5-flash-8B (Google, 2024). The model versions are 20240806 for GPT-4o, 20240718 for GPT-4o-mini, 20240723 for LLaMA-3.1, and the latest stable release for Gemini-1.5-flash models as of November. LLaMA-3.1 models are accessed through the DeepInfra API (DeepInfra, 2024). All models use default configurations as per their respective API documentation (OpenAI, 2024; Google, 2024; DeepInfra, 2024).

Direct Prompt

[Instruction]:
You are an expert in Named Entity Recognition in finance. Please identify the Person, Organization, and Location Entity from the given text.
[Text]:
The target text.
[Answer Format]:
Please answer in the following format, separate each with a comma, and answer 'none' if there is no specific entity in that type:
[Person]: list of entities
[Organization]: list of entities
[Location]: list of entities

Figure 1: Direct prompt for the NER task.

Additionally, we evaluate transformer-based models for comparison: BERT (Devlin, 2018) and RoBERTa (Liu, 2019). These models are initialized with pre-trained versions available in the Hugging Face Transformers library (Wolf et al., 2020), using a batch size of 16, a learning rate of 1e-05, and 50 epochs. Fine-tuning is performed on an Nvidia Tesla A100 GPU via Google Colab (Google, 2024).

3.3 Prompt Design

We design three types of prompt methods: direct prompt, in-context learning (Dong et al., 2022), and chain-of-thought (Wei et al., 2022). As shown in Figure 1, the direct prompt first gives instructions for the NER task, followed by the given text and the answer format. Next, we conduct few-shot learning (five shots) experiments through in-context learning and CoT prompts. The shots are chosen randomly and the same five shots are used in every experiment. For the in-context learning prompt, we simply add the five examples after the NER task instruction of the direct prompt. For the chain-of-thought prompt, we use the instruction "let's think step by step" to design intermediate steps for identifying each named entity in the text, as shown in Figure 2.

Chain-of-Thought Prompt

[Instruction]
[Example]:
Here are five examples, let's think step by step:
1. *[text]*
   The named person in the text are ....
   The named organization in the text are ....
   The named locations in the text are ....
   Therefore, the answer should be ....
2. ...
[Text]
[Answer Format]
Let's think step by step.

Figure 2: The chain-of-thought prompt for experiments.

3.4 Evaluation Metrics

After obtaining answers from the generated text, we label the identified entities through word matching. The evaluation metrics include the *entity-level F1 score* and the *weighted F1 score*. The formula for *entity-level F1 score* is described below, where TP, FP, and FN represent the counts of True Positives, False Positives, and False Negatives, respectively.

$$Precision = \frac{TP}{(TP + FP)}, \; Recall = \frac{TP}{(TP + FN)} \quad (1)$$

$$F1\_Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (2)$$

The *weighted F1 score* is defined as follows:

$$w_i = \frac{No.\_of\_entities\_in\_class_i}{Total\_number\_of\_entities} \quad (3)$$

$$Weighted\_F1 = \sum_{i=1}^{N} (w_i * F1\_Score_i) \quad (4)$$

4 Experiments

In this work, we conduct experiments to answer the following two research questions.

4.1 RQ1: How do different LLMs perform in FiNER-ORD tasks under different prompts?

We present the performance results of three leading LLMs under three distinct prompts in Table 1. The results are measured using the F1 scores for three entity types and the weighted F1 score (shown in the *Weighted* column). The LLMs are grouped into two sections based on their size, with **bold values** highlighting the best performance. From these results, we can draw the following observations.

(1) Fine-tuned language models consistently outperform generic LLMs, the performance gap can be narrowed through prompt design, few-shot learning, and model size. Table 1 demonstrates that fine-tuned language models surpass generic LLMs in zero-shot direct prompting. However, the performance of generic LLMs improves significantly with diverse zero-shot prompting styles, surpassing the prompt designs proposed by Shah et al., 2023. Additionally, few-shot learning and larger LLMs demonstrate notable advantages over their smaller counterparts.

(2) Chain-of-Thought prompting has limited effect on LLMs performance and can sometimes reduce effectiveness. While few-shot learning generally enhances generic LLMs' performance, Table 1 shows that the difference between prompting styles is marginal. CoT prompting only improves the performance of the GPT-4o-mini model, whereas it significantly degrades the performance of the LLaMA 3.1 series. Notably, LLaMA 3.1 frequently suffers from "implied entities" errors, where it tends to overanalyze and tag words that merely imply a named entity. This failure type is further discussed in subsequent sections.

Table 1: Performance of different fine-tuned language models and LLMs under different prompts on FiNER-ORD task.

Model	PER	LOC	ORG	Weighted
Fine-Tuned Language Models				
BERT	**0.9664**	0.8674	0.8313	0.8744
RoBERTa	0.9663	**0.8748**	**0.8379**	**0.8792**
LLMs				
GPT-mini	0.8296	0.7669	0.6824	0.7396
LLaMA-8B	**0.8799**	**0.7973**	**0.7299**	**0.7839**
Gemini-8B	0.8536	0.7773	0.6732	0.7434
GPT	0.9023	0.8009	**0.7312**	**0.7910**
LLaMA-70B	**0.9042**	0.7958	0.7073	0.7781
Gemini	0.8802	**0.8228**	0.7238	0.7868
Few-Shot Learning (5-shot) In-Context Learning				
GPT-mini	0.9265	**0.8061**	0.6841	0.7743
LLaMA-8B	0.8681	0.7681	0.7132	0.7655
Gemini-8B	**0.9308**	0.7991	**0.7468**	**0.8059**
GPT	0.9372	**0.8381**	0.7541	0.8203
LLaMA-70B	0.9415	0.7947	0.7948	0.8321
Gemini	**0.9418**	0.8106	**0.7966**	**0.8368**
Chain-of-Thought (CoT)				
GPT-mini	0.9221	**0.8072**	0.7389	0.8015
LLaMA-8B	0.8467	0.7505	0.7005	0.7494
Gemini-8B	**0.9343**	0.7900	**0.7408**	**0.8016**
GPT	0.9361	**0.8295**	0.7466	0.8142
LLaMA-70B	0.9122	0.7996	0.7514	0.8036
Gemini	**0.9378**	0.8171	**0.7958**	**0.8369**

Table 2: Failure types, distributions, and examples. Entities and their wrong recognitions are highlighted with blue and red, respectively.

Failure Type	Ratio	Example text and mislabeled entities
Contextual misunderstanding	31.3%	*Johnson Brothers rethink plan for St. Paul waterfront Shepard Road Development.* The company *"Johnson Brothers"* is mislabeled as a person.
Pronouns and generic terms	26.3%	*Nokia was holding exclusive talks with the German car makers.* Non-entity *"German car makers"* is mislabeled as an organization entity.
Citizenship	10.3%	*One suffered by a reported 66% of the British population.* Non-entity *"British"* is mislabeled as a location entity as it relates to the UK.
Implied entities	10.7%	*People use Google Maps or another navigation service to get to their destination .* Non-entity *"Google Maps"* is mislabeled as an organization as it refers to Google.
Entity omission	21.4%	*Will General Motors (NYSE : GM) be next ?* Abbreviation entity *"NYSE"* is not recognized.
Boundary errors		*Johnson Brothers rethink plan for St. Paul waterfront Shepard Road Development.* Only *"St. Paul"* is labeled instead of complete location, *"St. Paul waterfront Shepard Road"*

(3) The Gemini series outperforms the GPT-4o and LLaMA 3.1 series in the FiNER-ORD task after few-shot learning. The Gemini series outperforms the GPT-4o and LLaMA 3.1 series in the FiNER-ORD task after few-shot learning. Experimental results indicate a consistent performance ranking, with the Gemini series achieving the optimal performance, followed closely by the GPT-4o series. The LLaMA 3.1 series exhibits the lowest performance among the three.

4.2 RQ2: What types of mistakes do LLMs commonly make?

We manually annotate the failure types, summarize the limitations of LLMs, and analyze the underlying causes based on their responses, as shown in Table 2. The most common failure cases include:

(1) Contextual misunderstanding of proper noun. LLMs often fail to classify entities that rely on context correctly, such as domain-specific terms or ambiguous entities. For example, person names that overlap with location names, and organizational entities containing person or location names may be incorrectly categorized.

(2) Pronouns and generic terms. Terms such as pronouns (*"he"* or *"a woman"*), and generic phrases (*"universities"* or *"automakers"*) are sometimes misclassified as specific entities.

(3) Citizenship Terms. Words related to citizenship, such as *"Chinese"* or *"British"*, are often misclassified as locations despite referring to national identities.

(4) Implied entities. LLMs frequently misinterpret terms that imply specific entities. For example, product names like *"iPhone"* or *"Google Maps"* are often mislabeled as organizational entities due to their association with companies.

(5) Entity omission and boundary errors. LLMs struggle to recognize certain entities, such as abbreviations or long entities (e.g., long addresses). They may either omit these entities entirely or incorrectly segment them.

5 Discussion

The findings of our study highlight several potential directions for improving the performance of LLMs on financial NER tasks:

Tuning LLMs for the Financial Domain. A significant proportion of the observed failure cases involve domain-specific proper nouns. Fine-tuning LLMs with financial data could enhance their ability to accurately recognize such entities.

Implementing self-correction strategies. Our analysis in RQ2 identifies common mistakes made by LLMs in the FiNER-ORD task. Developing self-verification prompting strategies could allow LLMs to recognize and address these errors, thereby reducing recurrent failures.

6 Conclusion

This study presents the first systematic evaluations of generic LLMs in the FiNER-ORD task under different prompt designs, compared to state-of-the-art fine-tuned transformer-based models. Through comprehensive experiments with LLMs and their related lightweight versions, we demonstrate the capabilities and limitations of generic LLMs in handling domain-specific tasks. Our findings categorize five representative types of failures, along with their underlying causes. We release artifcats

for future research [1].

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.

Matin N Ashtiani and Bijan Raahemi. 2023. News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Systems with Applications*, 217:119509.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

DeepInfra. 2024. Deep Infra model cards. https://deepinfra.com/models. Accessed: 2024-11-10.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Google. 2024. Gemini API. https://ai.google.dev/gemini-api. Accessed: 2024-11-20.

Google. 2024. Google Colaboratory. https://colab.research.google.com/. Accessed: 2024-11-15.

Xianzhi Li, Samuel Chan, Xiaodan Zhu, Yulong Pei, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2023a. Are ChatGPT and GPT-4 general-purpose solvers for financial text analytics? A study on several typical tasks. *arXiv preprint arXiv:2305.05862*.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023b. Large language models in finance: A survey. In *Proceedings of the fourth ACM international conference on AI in finance*, pages 374–382.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

OpenAI. 2024. GPT-4o. https://chat.openai.com. Accessed: 2024-10-24.

OpenAI. 2024. Vision Guide. https://platform.openai.com/docs/guides/vision. Accessed: 2024-10-24.

Libo Qin, Qiguang Chen, Xiachong Feng, Yang Wu, Yongheng Zhang, Yinghui Li, Min Li, Wanxiang Che, and Philip S Yu. 2024. Large language models meet nlp: A survey. *arXiv preprint arXiv:2405.12819*.

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*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2024. Pixiu: A comprehensive benchmark, instruction dataset and large language model for finance. *Advances in Neural Information Processing Systems*, 36.

[1] https://github.com/Alex-Lyu0419/Financial-Named-Entity-Recognition-How-Far-Can-LLM-Go