TermGPT: Multi-Level Contrastive Fine-Tuning for Terminology Adaptation in Legal and Financial Domains

Yidan Sun[1], Mengying Zhu[1]*, Feiyue Chen[1], Yangyang Wu[1], Xiaolei Dan[2], Mengyuan Yang[1], Xiaolin Zheng[1], Shenglin Ben[1]

[1]Zhejiang University, China
[2] Innovation Department, National FinTech Risk Monitoring Center, China
{yidansun, mengyingzhu, fychen02, zjuwuyy, yangmy412, xlzheng, benshenglin}@zju.edu.cn
danxiaolei@nfrm.org.cn

Abstract

Large language models (LLMs) have demonstrated impressive performance in text generation tasks; however, their embedding spaces often suffer from the isotropy problem, resulting in poor discrimination of domain-specific terminology, particularly in legal and financial contexts. This weakness in terminology-level representation can severely hinder downstream tasks such as legal judgment prediction or financial risk analysis, where subtle semantic distinctions are critical. To address this problem, we propose `TermGPT`, a multi-level contrastive fine-tuning framework designed for terminology adaptation. We first construct a sentence graph to capture semantic and structural relations, and generate semantically consistent yet discriminative positive and negative samples based on contextual and topological cues. We then devise a multi-level contrastive learning approach at both the sentence and token levels, enhancing global contextual understanding and fine-grained terminology discrimination. To support robust evaluation, we construct the first financial terminology dataset derived from official regulatory documents. Experiments show that `TermGPT` outperforms existing baselines in term discrimination tasks within the finance and legal domains.

Code — https://github.com/Thoams0211/TermGPT

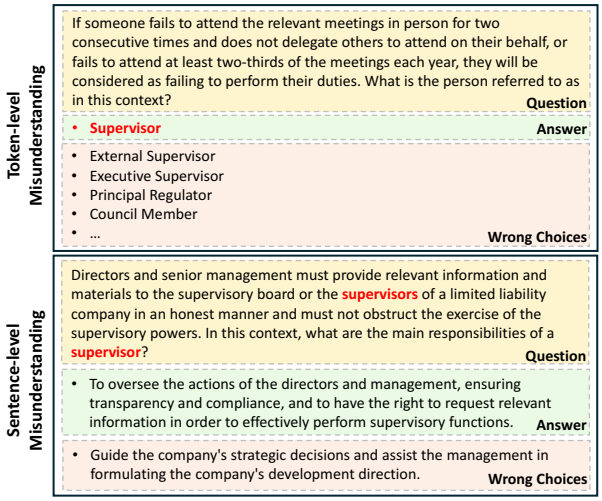

Figure 1: Motivation Example: Terminology Misunderstanding in a Loan Application.

Introduction

Large language models (LLMs) have made significant progress in text generative tasks. However, the embedding spaces learned from LLMs often suffer from the isotropy problem, where the token embeddings are distributed too uniformly in high-dimensional space, resulting in insufficient semantic discriminability (Mickus, Grönroos, and Attieh 2024; Mu, Bhat, and Viswanath 2017; Tsukagoshi and Sasano 2025). This problem hampers the LLMs' ability to distinguish subtle yet crucial semantic differences between domain-specific terminology, thereby limiting accurate understanding of terminology.

This limitation is particularly problematic in high-stakes domains, such as finance and law, where precise interpretation of terminology is critical for downstream tasks, including loan application and compliance advisory services (Chen et al. 2023; Liu et al. 2023). In these scenarios, even slight confusion between related terms may lead to incorrect reasoning and serious real-world consequences. Taking loan application as an example, as shown in Figure 1, an LLM deployed in a banking system misinterprets term "*supervisor*" as "*executive supervisor*" during a loan review. This misunderstanding causes the LLM to overlook a key omission of disclosure to the supervisory board. As a result, loan is mistakenly approved, exposing institution to financial loss, regulatory penalties, and reputational risk. To mitigate such risks, there is growing need for fine-tuning methods that enhance the ability of LLMs to distinguish domain-specific terminology with high semantic precision. In response to this need, we formally define a new task, **terminology-aware fine-tuning task**, which focuses on improving the representation and discrimination of specialized terms in domain-specific contexts.

Recently, several studies (Kim, Lee, and Hwang 2024; Rudman and Eickhoff 2023) have focused on improving the quality of token embeddings, demonstrating potential for completing terminology-aware fine-tuning task. First, the supervised fine-tuning (SFT) method (Kim et al. 2024) aligns LLMs' output with labeled instruction–response pairs, helping the

*Corresponding author

model distinguish between semantically correct and incorrect responses, and improving separability of token embeddings. Second, sentence-level contrastive learning methods (Gunel et al. 2020; Seanie Lee and Ju 2020) fine-tune LLMs by constructing positive and negative pairs at the sentence level, enhancing model's ability to encode global contextual semantics. Thirdly, token-level contrastive learning methods (Zhang et al. 2022; Jiang et al. 2022) apply contrastive objectives at token level, often incorporating frequency-based weighting or regularization to enhance term-level discrimination. While these methods offer valuable components for terminology-aware fine-tuning, they are not directly applicable in practice.

This is primarily due to three challenges that arise in the context of terminology-aware fine-tuning. First, *supervision signals for terminology are ambiguous* **(CH1)**. Terminology is highly context-dependent, with the same terminology potentially carrying different meanings across legal, financial, or regulatory contexts. This ambiguity makes it difficult to provide consistent supervision during fine-tuning. Secondly, *terminology is extremely sparse in domain-specific corpora* **(CH2)**. Terms constitute only 0.08% of the JecQA dataset (Zhong et al. 2020). This extreme sparsity leads to weak signal strength, causing term representations to be overwhelmed by general contextual patterns. Thirdly, *terminology-aware fine-tuning involves critical trade-off in leveraging contextual information* **(CH3)**. Over-reliance on context may obscure term-specific semantics, while ignoring context may isolate terms from their functional usage. Without a proper balance between global context and local term meaning, model struggles to learn accurate and discriminative term representations.

To address the above challenges, we propose `TermGPT`, a multi-level contrastive learning framework for terminology-aware fine-tuning. Specifically, we construct a sentence graph that jointly captures semantic and structural relationships among sentences. Based on this graph, we introduce a graph-driven data augmentation method that generates more accurate and diverse positive and negative pairs (*for addressing* **CH1**). We then design a multi-level contrastive learning mechanism. On the one hand, we propose a token-level contrastive learning mechanism that constructs positive and negative samples around key tokens, thereby preventing important terms from being diluted during training (*for addressing* **CH2**). On the other hand, we propose a sentence-level contrastive learning mechanism, where context-enriched sentence embeddings are optimized to capture the global semantics of terms, thereby alleviating the limitations of isolated token-level modeling (*for addressing* **CH3**).

In summary, our main contributions are as follows:

- *New task:* We define terminology-aware fine-tuning as a novel task that aims to enhance LLMs' fine-grained understanding of domain-specific terminology. To the best of our knowledge, this is the first work to formalize terminology-aware fine-tuning as a standalone objective, providing clear benefits for terminology-sensitive downstream tasks.
- *Novel framework:* We propose `TermGPT`, a multi-level contrastive fine-tuning framework that jointly models global context and local term semantics. It enhances representation of sparse and imbalanced terminology while effectively integrating contextual information to improve semantic discrimination.
- *Specialized dataset:* We construct a new dataset for financial terminology-aware fine-tuning, consisting of 3,647 domain-specific terms extracted from 425 authoritative financial regulatory documents.
- *Extensive experiments:* Experiments on both the newly collected financial dataset and the legal-domain JecQA benchmark show that `TermGPT` outperforms existing baselines, achieving an average improvement of 6.14% in terminology Question-Answering (QA) tasks and 2.60% in terminology Question-Choice Answering (QCA) tasks.

Related Work

Improving the quality of token embeddings has great potential for terminology-aware fine-tuning, as it significantly enhances LLM's ability to understand and distinguish terms. Therefore, we introduce methods aimed at enhancing the effectiveness of token embeddings.

Traditional Embedding Methods. Early embedding methods, such as Sentence-Bert (Reimers and Gurevych 2019), GTR (Ni et al. 2021), and bge-M3 (Chen et al. 2024), are based on BERT (Devlin et al. 2019) and T5 (Raffel et al. 2020) and fine-tuned on large-scale supervised datasets for tasks like retrieval. As LLMs evolved, generative embedding methods like GenEOL (Thirukovalluru and Dhingra 2024), MetaEOL (Lei et al. 2024), and Echo embedding (Springer et al. 2024) emerged, improving sentence embeddings through diverse transformations and contextualized generation. However, they still struggle with subtle semantic variations, limiting embedding precision and discriminability.

Contrastive Learning-based Embedding Methods. Contrastive learning is widely used in text embedding to align similar texts and separate dissimilar ones. Existing studies can be divided into sentence-level and token-level mechanisms. On the one hand, **sentence-level contrastive learning** builds positive/negative pairs and optimizes embeddings with methods like InfoNCE (Oord, Li, and Vinyals 2018) and SimCSE (Gao, Yao, and Chen 2021). Approaches such as GritLM (Muennighoff et al. 2024), LLM2Vec (BehnamGhader et al. 2024), AutoRegEMbed (Deng et al. 2025) and NV-embed (Lee et al. 2024) enhance sentence embedding by introducing bidirectional or latent attention to capture global semantics. Nevertheless, sentence-level contrastive learning often dilutes token-level semantics, reducing effective for terminology-sensitive tasks. On the other hand, **token-level contrastive learning** enhances terminology distinction by optimizing individual tokens. CT (Jiang et al. 2022) introduced contrastive token learning to improve terminology discrimination. TCL and FCL (Zhang et al. 2022) increased the weight of low-frequency tokens to improve representation learning. SimCTG (Su et al. 2022) penalized token similarity to learn more distinguishable embeddings, boosting the recognition and generation of fine-grained semantic differences. However, token-level contrastive learning focuses on isolated token optimization, limiting the capture of global semantics and interdependencies, and hindering sentence embedding.

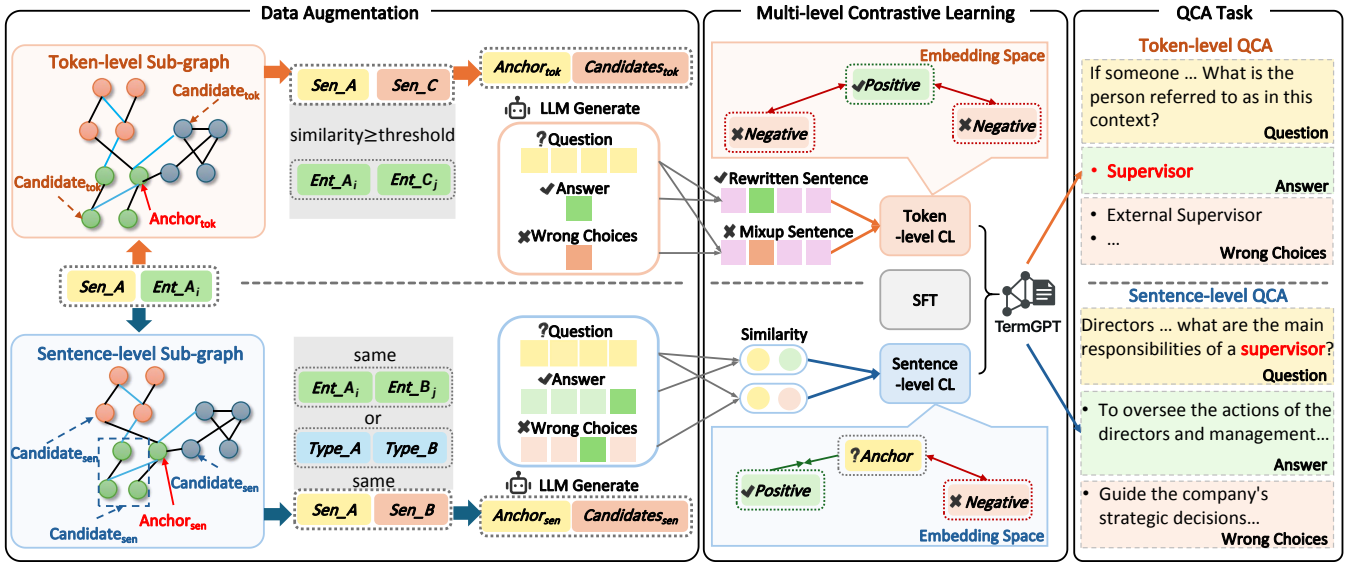

Figure 2: Overall framework of TermGPT. We first construct a sentence graph with sentences as nodes and different semantic and structural relationships as edges, where edges representing semantic ambiguity are black and lexical ambiguity edges are blue. Each node is used as an anchor sample, and its candidate samples are used for data augmentation to generate QCA pairs. Finally, contrastive learning is applied at different levels to distinguish the differences in terminology embeddings based on the QCA categories.

Preliminary

In this section, we will introduce basic concepts of terminology-aware fine-tuning, the QCA task, and sentence graph construction.

Terminology-aware Fine-tuning. Terminology-aware fine-tuning reduces the probability of terminology misuse by minimizing the cross-entropy loss. Given a question q, the correct answer a, and candidate incorrect answers $c_1, c_2, \ldots, c_k$, the model computes the probability $P_\theta(a|q)$ and is trained to maximize it by minimizing the loss $-\sum_{i=1}^{n} \log P_\theta(a_i|q_i)$. This enables LLMs to more accurately distinguish between semantically similar but distinct terms, significantly reducing the risk of misuse in domain-specific applications.

The QCA Task. In the QCA task, the objective is to select the correct answer from multiple candidate choices $a_i, c_{i_1}, c_{i_2}, c_{i_3}$, given the question q_i. Specifically, the input sample $Q_{sen} = (q_i, a_i, c_{i1}, c_{i2}, c_{i3})$ contains the question and its candidate answers, and the LLM is required to understand the context of the question q_i and correctly select a_i from the candidates.

Sentence Graph. In the sentence graph $G = (V, E)$, nodes represent sentences and edges represent semantic and structural relationships between sentences. There are various semantic relationships between sentences, including sentences that contain the same entities, which are often prone to confusion at the sentence-level semantics. Sentences containing similar entities are more likely to cause confusion at lexical meaning level. Additionally, some sentences may not contain exactly the same entities but may belong to the same semantic category, such as legal clauses in the same field. These sentences of the same type often exhibit high similarity in terminology of theme or structure, which can still lead to misjudgment at sentence meaning level. Therefore, in the sentence graph G, different types of confusion are represented by different edges. Specifically, for a sentence s, we perform entity extraction by combining LLM with a schema, obtaining an entity set $C = (c_1, c_2, \ldots, c_k)$. If two sentences s_i, s_j contain the same entities or have the same type, they are connected with an edge denoted as $\text{edge}_{\text{sen}}(s_i, s_j)$. While, if two sentences (s_i, s_j) contain similar entities (c_i, c_j), and the embeddings of c_i and c_j exceed a certain threshold θ, an edge $\text{edge}_{\text{token}}(s_i, s_j)$ is connected between the two sentences.

Methodology

In this section, we introduce the TermGPT framework, which consists of a data augmentation strategy and a multi-level contrastive learning mechanism. We begin by describing the data augmentation method based on a sentence graph. Then, we present the multi-level contrastive learning mechanism from both sentence-level and token-level perspectives. Finally, we summarize the overall optimization pipeline.

Data Augmentation

In the terminology-aware fine-tuning dataset, terms typically occupy only a small portion of the dataset, which makes the embedding of terminology susceptible to dilution in sentence-level modeling, thereby affecting the LLM's ability to perceive fine-grained differences between terms. To effectively enhance the representational ability of terminology, we propose a sentence graph-based data augmentation method. By constructing a graph that captures semantic relationships between sentences, we generate more diverse and

informative positive and negative sample pairs, thus improving the model's discriminative ability. The edges in the sentence graph can reflect various semantic relations, helping the model capture the diversity and subtle differences between terms. This method can effectively enrich the dataset, thereby enhancing the semantic differentiation ability of terminology and improving the model's reasoning performance.

Specifically, in the sentence graph, we first traverse each node, with the current selected node being the anchor node s_{anchor}. For each node s_j connected to it by an $edge_{\text{sen}}$, if the cosine similarity between its sentence-level embedding vector $\mathbf{e}_{\text{sen}}(s_j)$ and the anchor node's sentence-level embedding vector $\mathbf{e}_{\text{sen}}(s_{\text{anchor}})$ is greater than a threshold θ_{sen}, then s_j is added to the candidate set $S_{\text{sen}}(s_{\text{anchor}})$ of s_{anchor}. For each node s_i connected to s_{anchor} by an $edge_{\text{tok}}$, let the similar entities be c_{anchor} and c_i. Then c_i is directly added to the candidate set $C_{\text{tok}}(s_{\text{anchor}}, c_{\text{anchor}})$ of s_{anchor}.

For any candidate set $S_{\text{sen}}(s_{\text{anchor}})$, we have the LLM generate a question q with s_{anchor} as anchor, and s_{anchor} as the correct option a, $s_j \in S_{\text{sen}}(s_{\text{anchor}})$ as the incorrect option. Additionally, two hard negative samples s_{j_1} and s_{j_2} are generated. This set of QCA samples $(q, a, s_j, s_{j_1}, s_{j_2})$ is added to the dataset Q_{sen}. Similarly, for any candidate set $C_{\text{tok}}(s_{\text{anchor}}, c_{\text{anchor}}) = \{c_1, c_2, \ldots, c_k\}$, we have the LLM generate a question q based on s_{anchor} for c_{anchor}, with c_{anchor} as the correct option a and $\{c_1, c_2, \ldots, c_k\}$ as incorrect options. This set of QCA samples $(q, a, c_1, \ldots, c_k)$ is then added to the dataset Q_{tok}.

Multi-level Contrastive Learning

Terminology-aware fine-tuning involves a trade-off between contextual understanding and terminology-specific semantics. To balance this trade-off, we propose a multi-level contrastive learning mechanism that integrates both global context and local semantic features for learning accurate terminology embeddings.

Sentence-level contrastive learning. We first utilize a LLM to generate embeddings for the input sample $x_i \in Q_{\text{sen}} = (q_i, a_i, c_{i1}, c_{i2}, c_{i3})$, where q_i is the question, a_i is the correct answer, and c_{i1}, c_{i2}, c_{i3} are the three candidate incorrect answers. We then activate a bidirectional attention mechanism to ensure the full integration of context information. Next, we optimize the contrastive learning objective using the InfoNCE loss function by computing the similarity between the embeddings of the question and the correct answer, as well as the similarity between the question and the three candidate incorrect answers. The loss function L_{sen} is defined as:

$$L_{\text{sen}} = \sum_{x_i \sim Q_{\text{sen}}} -\log \frac{\exp(\mathbf{e}_{q_i} \cdot \mathbf{e}_{a_i}/\tau)}{\sum_{j=1}^{4} \exp(\mathbf{e}_{q_i} \cdot \mathbf{e}_{c_{ij}}/\tau)}, \quad (1)$$

where τ is the temperature parameter. The goal is to maximize the similarity between the question and the correct answer, while minimizing the similarity between the question and the incorrect options, thus enhancing the LLM's ability to discriminate the semantics of the terminology.

Token-level contrastive learning. Given the input sample $x_i \in Q_{\text{tok}} = (q_i, a_i, c_{i1}, \ldots, c_{ik})$, we use the LLM to convert (q_i, a_i) into a declarative sample t_i, where a_i appears as a subsequence $\{a_i\} \subset t_i$ within the sentence t_i, and the set of negative samples is denoted as $C_{\text{neg}} = (c_{i1}, \ldots, c_{ik})$. We treat the correct answer a_i as a positive sample z^+, and each incorrect answer c_{ij} as a negative sample z^-, and mix the positive and negative sample pair (z^+, z^-) with the sentence t to form a mixed sequence $\tilde{t} = \text{Mix}(t, z^+, z^-)$. During the mixing process, the function replaces z^+ in the sentence t with z^-. We define a sequence φ with the same length as $\tilde{t}$, where 1 indicates tokens from z^+ or t, and 0 indicates tokens from z^-. Based on the mixed sequence $\tilde{t}$ and φ, we design the following loss function l_{mix}:

$$l_{\text{mix}}(t, z^+, z^-, q) = -\sum_{j=1}^{|\tilde{t}|} \Big[\varphi_{i,j} \log p_\theta(\tilde{t}_{i,j} | \tilde{t}_{i,<j}, q) + (1 - \varphi_{i,j}) \log \big(1 - p_\theta(\tilde{t}_{i,j} | \tilde{t}_{i,<j}, q)\big) \Big]. \quad (2)$$

For the entire set Q_{tok}, the token-level contrastive learning loss function L_{tok} is expressed as:

$$L_{\text{tok}} = \sum_{x \sim Q_{\text{tok}}} \sum_{i=1,\ldots,k} l_{\text{mix}}(t, z^+, z_i^-, q). \quad (3)$$

By integrating sentence-level and token-level contrastive learning, we construct a robust framework that enhances both fine-grained terminology differentiation and contextual coherence, thereby improving overall model performance.

Algorithm 1: Optimization Pipeline of `TermGPT`

Input: $Q_{\text{sen}}, Q_{\text{tok}}$
Output: θ
// Step 1: Supervised Fine-Tuning (SFT)
for $x \in Q_{\text{sen}} \cup Q_{\text{tok}}$ **do**
  Compute L_{SFT}.
  Update θ to minimize L_{SFT}.
end for
// Step 2: Sentence-level Contrastive Learning
for $(q_i, a_i, c_{i_1}, c_{i_2}, c_{i_3}) \in Q_{\text{sen}}$ **do**
  Let $\mathbf{e}_{q_i}, \mathbf{e}_{a_i}, \mathbf{e}_{c_{i1}}, \mathbf{e}_{c_{i2}}, \mathbf{e}_{c_{i3}}$ be the embeddings.
  Compute L_{sen}.
  Update θ to minimize L_{sen}.
end for
// Step 3: Token-level Contrastive Learning
for $(q_i, a_i, c_{i_1}, \ldots, c_{i_k}) \in Q_{\text{tok}}$ **do**
  Convert (q_i, a_i) to a declarative sentence t_i using the LLM.
  Let $\tilde{t} = \text{Mix}(t, z^+, z^-)$ be the mixed sequence.
  Let $\mathbf{e}_t$ be the token embeddings of $\tilde{t}$ and φ be the binary sequence indicating whether a token is from z^+ or z^-.
  Compute $l_{\text{mix}}(t, z^+, z^-, q)$.
  Compute $L_{\text{tok}} = \sum_{x \sim Q_{\text{tok}}} \sum_{z_i^- \sim C_{\text{Neg}}}^{i=1,\ldots,k} l_{\text{mix}}(t, z^+, z_i^-, q)$.
  Update θ to minimize L_{tok}.
end for

Optimization Pipeline

In the overall optimization pipeline, we first perform a unified SFT on the sentence set Q_{sen} and token set Q_{tok}. During this phase, the LLM learns the mapping from the original input q_i to the target output y_i by fine-tuning on downstream tasks. The loss function for SFT, denoted as L_{SFT}, is defined as:

$$L_{\text{SFT}} = -\sum_{x \in Q_{\text{sen}} \cup Q_{\text{tok}}} \log p_\theta(y_q | x_q), \tag{4}$$

where x_q represents the input sample, y_q is the corresponding target output. After completing the SFT phase, sentence-level contrastive learning is first applied to Q_{sen}, maximizing the similarity between the question and the correct answer while minimizing the similarity between the question and the incorrect options. Subsequently, `TermGPT` performs token-level contrastive learning on Q_{tok}, further refining the fine-grained distinction of terminology through mixed sequences t and φ. The pseudocode is as Algorithm 1.

Experiment

In this section, we present extensive experiments to answer the following questions:

RQ1: How does `TermGPT` perform compared to the baselines on the terminology QCA tasks?

RQ2: How does `TermGPT` perform compared to the baselines on the terminology QA tasks?

RQ3: How does the multi-level contrastive learning mechanism affect `TermGPT`'s ability to understand domain-specific terminology?

RQ4: How well does `TermGPT` generalize to terminology understanding tasks across different domains?

Experimental Settings

Dataset. We conduct experiments on two datasets covering financial regulations and legal QA to evaluate terminology understanding capabilities of `TermGPT` in high-stakes domains. The statistics of datasets are shown in Table 1.

(1) Financial Regulations Dataset. We construct a dataset from 425 regulatory rules extracted from officially published Chinese financial supervision documents, covering domains such as anti-money laundering, interbank lending, and loan management. Sentence graphs are built for all samples, and to prevent data leakage, we augment the nodes in the train set and test set separately.

(2) JecQA Dataset. To further verify `TermGPT`'s generalization, we adopt JecQA (Zhong et al. 2020), a large-scale legal QA dataset composed of knowledge-driven multiple-choice questions collected from the National Judicial Examination and related sources, which contains 26365 questions. For alignment with our regulatory dataset, we sample 1038 original questions from JecQA, using the same 7:3 split and data augmentation pipeline.

We construct two test formats for both datasets: QCA-format, which assesses fine-grained semantic distinction, and QA-format, which evaluates answer generation and rule comprehension. All test sets are derived from QCA samples via LLM-based transformation. Details of data collection and processing are described in the Appendix A.

Dataset		Original		Augmented	
		Train	Test	Train	Test
JecQA	Economic law	99	42	802	560
	Civil law	268	114	5586	4887
	Civil Procedure Law	211	90	7120	6277
	Law of commerce	150	64	1112	657
	Total	728	310	14620	12381
Financial Regulations	Loan management	99	29	11718	5551
	Plan fund management	61	26	12798	4043
	Risk management	65	27	7742	3589
	Company management	83	35	13974	6645
	Total	308	117	46232	19828

Table 1: Statistics of the datasets.

Comparison Methods. We compare `TermGPT` with three categories of baselines. The first includes pre-trained domain LLMs, such as Lawyer-LLaMA (Huang et al. 2023) and Xuanyuan (Zhang and Yang 2023), trained on legal and financial data to enhance domain-specific capabilities without task-specific supervision. The second consists of contrastive learning-based methods, including AutoRegEmbed (Deng et al. 2025), PromptEOL (Jiang et al. 2023), GritLM (Muennighoff et al. 2024), NV-embed (Lee et al. 2024) and Flag-Embedding (Li et al. 2024), which optimize sentence embeddings via contrastive learning. The third includes commercial LLMs, such as Qwen-2.5-plus (Yang et al. 2025) and Deepseek-v3 (Liu et al. 2024), evaluated via prompting to assess general language and domain knowledge.

Metrics. Following previous studies (Raffel et al. 2020; Lewis et al. 2019), we adopt a set of generation metrics to evaluate the quality of the responses generated in the QA task, including BLEU-1, BLEU-4 (Papineni et al. 2002), ROUGE-1, ROUGE-L (Lin 2004), METEOR (Banerjee and Lavie 2005), and BERTScore (Zhang et al. 2019). These metrics collectively capture lexical overlap, fluency, and semantic similarity between the generated answers and the ground truth. For the QCA task, we report standard classification metrics: Accuracy, Precision, Recall, and F1, respectively.

Implemental Details. Model configuration and training setup are provided in the Appendix B.

Overall Comparison (RQ1 & RQ2)

To address RQ1 and RQ2, we compare `TermGPT` with all baseline methods on terminology QCA (in Table 2) and QA (in Table 4) tasks. We highlight three main findings below.

Result 1: Performance on Terminology QCA Tasks. Contrastive learning-based methods slightly trail commercial LLMs but still outperform Lawyer-LLaMA by 63.00% and Xuanyuan by 17.69% on average. This highlights the efficiency of contrastive objectives in improving terminology-level discrimination, especially in low-parameter settings. Among contrastive learning-based methods, `TermGPT` outperforms the best baseline by 2.60% on average. This is primarily because its multi-level supervision, which integrates sentence-level and token-level contrastive signals to better distinguish subtle semantic differences between terms.

Dataset	JecQA				Financial Regulations			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Pre-trained Domain LLMs								
Lawyer-LlaMA(13B)	0.480	0.496	0.480	0.457	0.620	0.621	0.620	0.620
Xuanyuan(13B)	0.808	0.812	0.808	0.807	0.701	0.707	0.701	0.701
Contrastive Learning-based Methods								
AutoRegEmbed-Qwen3(8B)	0.778	0.779	0.778	0.778	0.883	0.884	0.883	0.883
PromptEOL-Qwen3(8B)	0.831	0.832	0.831	0.831	0.887	0.888	0.887	0.887
GritLM-Qwen3(8B)	0.834	0.835	0.834	0.834	0.876	0.877	0.876	0.876
NV-Embed-Qwen3(8B)	0.816	0.817	0.816	0.816	0.886	0.887	0.886	0.886
Flag-Embedding-Qwen3(8B)	0.792	0.793	0.792	0.792	0.879	0.879	0.879	0.879
Commercial LLMs								
Qwen-2.5-plus (72B)	0.894	0.895	0.894	0.894	0.935	0.936	0.935	0.936
Deepseek-v3 (671B)	0.884	0.885	0.884	0.884	0.938	0.939	0.938	0.938
Ours								
TermGPT-LlaMA (8B)	0.681	0.682	0.681	0.681	0.856	0.857	0.857	0.857
TermGPT-Qwen3 (8B)	**0.858**	**0.858**	**0.858**	**0.858**	**0.908**	**0.909**	**0.908**	**0.908**
Improvement [1]	+2.87%	+2.75%	+2.87%	+2.87%	+2.36%	+2.36%	+2.36%	+2.36%
p-value [2]	0.008	0.008	0.008	0.008	0.001	0.001	0.001	0.001

[1] Improvement over the best-performing baselines, which exclude commercial LLMs due to their substantially larger parameter scales.
[2] The improvement is significant based on a paired t-test at the significance level of 0.05 (p-value with paired t-test).

Table 2: Comparison of different models on terminology QCA task in terms of Accuracy, Precision, Recall and F1.

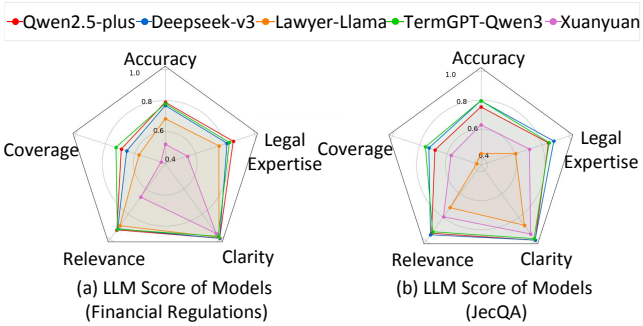

Figure 3: Comparison of different models on various datasets in terms of LLM Score.

Dataset		Accuracy	Precision	Recall	F1
TermGPT (w/o SFT)	JecQA	0.796	0.797	0.796	0.796
	Financial	0.818	0.819	0.818	0.818
TermGPT (w/o CL)	JecQA	0.834	0.835	0.834	0.834
	Financial	0.864	0.865	0.864	0.864
TermGPT (w/o Token-level CL)	JecQA	0.844	0.844	0.844	0.844
	Financial	0.899	0.900	0.899	0.899
TermGPT (w/o Sentence-level CL)	JecQA	0.849	0.850	0.849	0.849
	Financial	0.894	0.895	0.894	0.894
TermGPT	JecQA	0.858	0.858	0.858	0.858
	Financial	0.908	0.909	0.908	0.908

Table 3: Impact of CL and SFT on TermGPT performance.

Result 2: Performance on Terminology QA Tasks. We evaluate the generated results with LLM in Figure 3, which shows that TermGPT achieves comparable performance to commercial LLMs and significantly outperforms the domain-pretrained Lawyer-LLaMA and Xuanyuan. This suggests that although our model is smaller (8B), targeted supervision enables it to bridge the performance gap. Lawyer-LLaMA and Xuanyuan, while domain-pretrained, lacks task adaptation, limiting their effectiveness. Besides, the results in Table 4 shows that TermGPT outperforms the best baseline by 6.14% on average, benefiting from SFT, which improves task alignment, and contrastive learning, which enhances terminology-specific generation accuracy.

Result 3: Backbone Selection. We experiment with two backbone LLMs and observe that Qwen3 significantly outperforms LLaMA. In particular, TermGPT-Qwen3 yields an average performance gain of 15.98% on the QCA task and 43.52% on the QA task compared to TermGPT-LLaMA. Thus we adopt Qwen3 as the default backbone in our experiments.

Ablation Experiment (RQ3)

To investigate RQ3, we conduct a series of ablation experiments to examine the individual contributions of multi-level contrastive learning and SFT to the performance of TermGPT. We report the results in Table 3.

Result 4: Effectiveness of Multi-level Contrastive Learning. We first remove the token-level contrastive learning (TermGPT w/o Token-level CL) and sentence-level contrastive learning (TermGPT w/o Sentence-level CL), respectively. The performance drops by 1.31% and 1.28% on average compared to the full multi-level version, indicating that aligning both global semantics and fine-grained terminology is critical for precise terminology understanding. Next, we compare the full model against two further variants: one without contrastive learning (TermGPT w/o CL) and one without SFT (TermGPT w/o SFT). The results show that removing contrastive learning leads to an average drop of 3.81%, while removing SFT results in a drop of 8.55%. This sug-

Dataset	JecQA						Financial Regulations					
	BLEU-1	BLEU-4	BERTscore	ROUGE-1	ROUGE-L	Meteor	BLEU-1	BLEU-4	BERTscore	ROUGE-1	ROUGE-L	Meteor
Pre-trained Domain LLMs												
Lawyer-Llama (13B)	0.203	0.105	0.678	0.313	0.232	0.192	0.216	0.096	0.680	0.310	0.221	0.176
Xuanyuan (13B)	0.260	0.119	0.713	0.375	0.311	0.221	0.238	0.106	0.683	0.334	0.254	0.179
Contrastive Learning-based Methods												
AutoRegEmbed-Qwen3(8B)	0.261	0.121	0.714	0.375	0.313	0.217	0.223	0.081	0.697	0.318	0.254	0.185
PromptEOL-Qwen3(8B)	0.253	0.119	0.712	0.371	0.309	0.219	0.222	0.082	0.697	0.316	0.253	0.188
GritLM-Qwen3(8B)	0.260	0.122	0.714	0.374	0.314	0.217	0.225	0.084	0.698	0.316	0.256	0.186
NV-Embed-Qwen3(8B)	0.261	0.121	0.713	0.375	0.314	0.220	0.224	0.080	0.695	0.317	0.253	0.187
Flag-Embedding-Qwen3(8B)	0.256	0.120	0.714	0.372	0.309	0.217	0.229	0.085	0.699	0.324	0.260	0.192
Commercial LLMs												
Qwen-2.5-plus (72B)	0.160	0.069	0.697	0.296	0.218	0.237	0.255	0.111	0.710	0.361	0.283	0.213
Deepseek-v3 (671B)	0.261	0.109	0.717	0.368	0.299	0.227	0.242	0.084	0.713	0.333	0.262	0.221
Ours												
TermGPT-LlaMA (8B)	0.152	0.074	0.624	0.251	0.182	0.132	0.212	0.091	0.656	0.297	0.217	0.152
TermGPT-Qwen3 (8B)	**0.266**	**0.125**	**0.716**	**0.376**	**0.316**	**0.222**	**0.274**	**0.126**	**0.708**	**0.371**	**0.302**	**0.201**
Improvement [1]	+1.92%	+2.46%	+0.28%	+0.27%	+0.64%	+0.91%	+15.13%	+18.87%	+1.28%	+11.08%	+16.15%	+4.69%
p-value [2]	0.013	0.015	0.005	0.027	0.044	0.037	0.024	0.022	0.010	0.023	0.002	0.001

[1] Improvement over the best-performing baselines, which exclude commercial LLMs due to their substantially larger parameter scales.
[2] The improvement is significant based on a paired t-test at the significance level of 0.05 (p-value with paired t-test).

Table 4: Comparison of different models on terminology QA task in terms of BLEU, BERTscore, ROUGE and Meteor.

gests that SFT helps the model better adapt to task-specific contexts, enhancing its ability to accurately understand and generate terminology, whereas contrastive learning improves the model's capacity to distinguish between semantically similar terms. Together, these findings demonstrate that both SFT and multi-level contrastive learning are essential for maximizing terminology understanding in TermGPT.

Performance Comparison in Different Domains (RQ4)

We assess TermGPT's performance across different sub-domains in both JecQA and financial regulations datasets.

Result 5: Stable Terminology Understanding in Different Domains As shown in Figure 4, the model maintains strong and balanced performance in both QCA and QA tasks in various domains. In the QCA task, Economic Law and Risk Management show slightly lower scores, likely due to their limited training data. In the QA task, Civil Law lags slightly. This may be attributed to the broader scope and greater ambiguity of Civil Law questions, which pose a challenge for accurate modeling. These results highlight TermGPT's robustness and reliability under diverse sub-domain conditions.

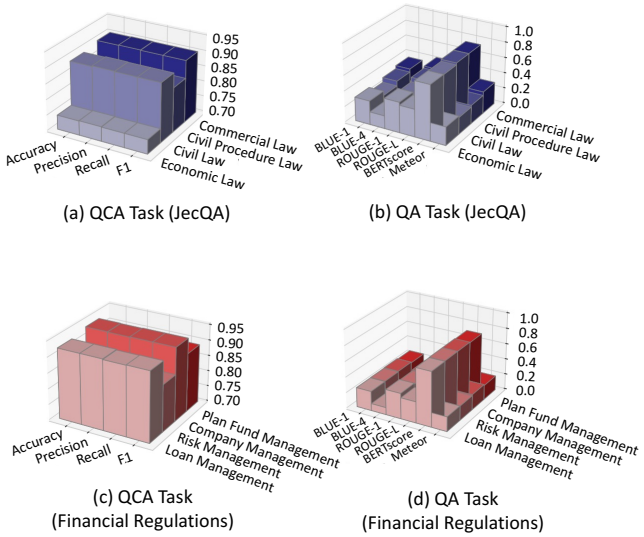

Figure 4: Performance of different domains on QCA and QA tasks.

Conclusions and Future Work

In this paper, we propose TermGPT, a fine-tuning framework that integrates multi-level contrastive learning, and define the novel task of terminology-aware fine-tuning. By constructing sentence graphs to capture semantic relationships, we generate high-quality sample pairs to help LLMs better distinguish subtle terminology differences. Our TermGPT framework balances the optimization of global and local semantics, thereby improving sensitivity to terminological distinctions. Extensive experiments demonstrate that TermGPT effectively performs terminology-aware fine-tuning in various tasks and domain-specific applications.

An important direction for future work is enhancing the robustness of terminology-aware LLMs against adversarial inputs. In high-stakes domains, adversarial inputs may distort LLMs' interpretation of key terminology. Future work can explore adversarial training or robust optimization to improve reliability under such conditions.

Acknowledgements

This work was supported by the National Key R&D Program of China (2024YFC3307702) and the cooperation project of ZJU-ZTCB Financial Technology Joint Research Center.

References

Banerjee, S.; and Lavie, A. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. of ACL Workshop*, 65–72.

BehnamGhader, P.; Adlakha, V.; Mosbach, M.; Bahdanau, D.; Chapados, N.; and Reddy, S. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; and Liu, Z. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Chen, W.; Wang, Q.; Long, Z.; Zhang, X.; Lu, Z.; Li, B.; Wang, S.; Xu, J.; Bai, X.; Huang, X.; et al. 2023. Disc-finllm: A chinese financial large language model based on multiple experts fine-tuning. *arXiv preprint arXiv:2310.15205*.

Deng, J.; Jiang, Z.; Pang, L.; Chen, L.; Xu, K.; Wei, Z.; Shen, H.; and Cheng, X. 2025. Following the autoregressive nature of llm embeddings via compression and alignment. *arXiv preprint arXiv:2502.11401*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*, 4171–4186.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Gunel, B.; Du, J.; Conneau, A.; and Stoyanov, V. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Huang, Q.; Tao, M.; Zhang, C.; An, Z.; Jiang, C.; Chen, Z.; Wu, Z.; and Feng, Y. 2023. Lawyer llama technical report. *arXiv preprint arXiv:2305.15062*.

Jiang, S.; Zhang, R.; Vakulenko, S.; and de Rijke, M. 2022. A simple contrastive learning objective for alleviating neural text degeneration. *arXiv preprint arXiv:2205.02517*.

Jiang, T.; Huang, S.; Luan, Z.; Wang, D.; and Zhuang, F. 2023. Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645*.

Kim, J.; Lee, D.; and Hwang, S.-w. 2024. Hil: Hybrid isotropy learning for zero-shot performance in dense retrieval. In *Proc. of NAACL*, 7885–7896.

Kim, S.; Sung, M.; Lee, J.; Lim, H.; and Perez, J. F. G. 2024. Efficient terminology integration for llm-based translation in specialized domains. *arXiv preprint arXiv:2410.15690*.

Lee, C.; Roy, R.; Xu, M.; Raiman, J.; Shoeybi, M.; Catanzaro, B.; and Ping, W. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

Lei, Y.; Wu, D.; Zhou, T.; Shen, T.; Cao, Y.; Tao, C.; and Yates, A. 2024. Meta-task prompting elicits embeddings from large language models. *arXiv preprint arXiv:2402.18458*.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2019. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Li, C.; Qin, M.; Xiao, S.; Chen, J.; Luo, K.; Shao, Y.; Lian, D.; and Liu, Z. 2024. Making text embedders few-shot learners. *arXiv preprint arXiv:2409.15700*.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Proc. of ACL Workshop*, 74–81.

Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Liu, X.-Y.; Wang, G.; Yang, H.; and Zha, D. 2023. Fingpt: Democratizing internet-scale data for financial large language models. *arXiv preprint arXiv:2307.10485*.

Mickus, T.; Grönroos, S.-A.; and Attieh, J. 2024. Isotropy, clusters, and classifiers. *arXiv preprint arXiv:2402.03191*.

Mu, J.; Bhat, S.; and Viswanath, P. 2017. All-but-the-top: Simple and effective postprocessing for word representations. *arXiv preprint arXiv:1702.01417*.

Muennighoff, N.; Hongjin, S.; Wang, L.; Yang, N.; Wei, F.; Yu, T.; Singh, A.; and Kiela, D. 2024. Generative representational instruction tuning. In *Proc. of ICLR*.

Ni, J.; Qu, C.; Lu, J.; Dai, Z.; Abrego, G. H.; Ma, J.; Zhao, V. Y.; Luan, Y.; Hall, K. B.; Chang, M.-W.; et al. 2021. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, 311–318.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Rudman, W.; and Eickhoff, C. 2023. Stable anisotropic regularization. *arXiv preprint arXiv:2305.19358*.

Seanie Lee, D. B. H.; and Ju, S. 2020. Contrastive learning with adversarial perturbations for conditional text generation. *arXiv preprint arXiv:2012.07280*.

Springer, J. M.; Kotha, S.; Fried, D.; Neubig, G.; and Raghunathan, A. 2024. Repetition improves language model embeddings. *arXiv preprint arXiv:2402.15449*.

Su, Y.; Lan, T.; Wang, Y.; Yogatama, D.; Kong, L.; and Collier, N. 2022. A contrastive framework for neural text generation. *Advances in Neural Information Processing Systems*, 35: 21548–21561.

Thirukovalluru, R.; and Dhingra, B. 2024. Geneol: Harnessing the generative power of llms for training-free sentence embeddings. *arXiv preprint arXiv:2410.14635*.

Tsukagoshi, H.; and Sasano, R. 2025. Redundancy, Isotropy, and Intrinsic Dimensionality of Prompt-based Text Embeddings. *arXiv preprint arXiv:2506.01435*.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; Lin, H.; Yang, J.; Tu, J.; Zhang, J.; Yang, J.; Yang, J.; Zhou, J.; Lin, J.; Dang, K.; Lu, K.; Bao, K.; Yang, K.; Yu, L.; Li, M.; Xue, M.; Zhang, P.; Zhu, Q.; Men, R.; Lin, R.; Li, T.; Tang, T.; Xia, T.; Ren, X.; Ren, X.; Fan, Y.; Su, Y.; Zhang, Y.; Wan, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; and Qiu, Z. 2025. Qwen2.5 Technical Report. arXiv:2412.15115.

Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K. Q.; and Artzi, Y. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zhang, T.; Ye, W.; Yang, B.; Zhang, L.; Ren, X.; Liu, D.; Sun, J.; Zhang, S.; Zhang, H.; and Zhao, W. 2022. Frequency-aware contrastive learning for neural machine translation. In *Proc. of AAAI*, volume 36, 11712–11720.

Zhang, X.; and Yang, Q. 2023. Xuanyuan 2.0: A large chinese financial chat model with hundreds of billions parameters. In *Proc. of CIKM*, 4435–4439.

Zhong, H.; Xiao, C.; Tu, C.; Zhang, T.; Liu, Z.; and Sun, M. 2020. JEC-QA: a legal-domain question answering dataset. In *Proc. of AAAI*, volume 34, 9701–9708.