GenCLS++: Pushing the Boundaries of Generative Classification in LLMs Through Comprehensive SFT and RL Studies Across Diverse Datasets

Anonymous ACL submission

Abstract

As a fundamental task in machine learning, text classification plays a crucial role in many areas. With the rapid scaling of Large Language Models (LLMs), particularly through reinforcement learning (RL), there is a growing need for more capable discriminators. Consequently, advances in classification are becoming increasingly vital for enhancing the overall capabilities of LLMs. Traditional discriminative methods map text to labels but overlook LLMs' intrinsic generative strengths. Generative classification addresses this by prompting the model to directly output labels. However, existing studies still rely on simple SFT alone, seldom probing the interplay between training and inference prompts, and no work has systematically leveraged RL for generative text classifiers and unified SFT, RL, and inference-time prompting in one framework. We bridge this gap with **GenCLS++**, a framework that jointly optimizes SFT and RL while systematically exploring five high-level strategy dimensions—in-context learning variants, category definitions, explicit uncertainty labels, semantically irrelevant numeric labels, and perplexity-based decoding—during both training and inference. After an SFT "policy warm-up," we apply RL with a rule-based reward, yielding sizable extra gains. **Across seven datasets, GenCLS++ achieves an average accuracy improvement of 3.46% relative to the naive SFT baseline; on public datasets, this improvement rises to 4.00%.** Notably, unlike reasoning-intensive tasks that benefit from explicit thinking processes, we find that classification tasks perform better without such reasoning steps. These insights into the role of explicit reasoning provide valuable guidance for future LLM applications.

1 Introduction

With the rapid advancement of Large Language Models (LLMs) (Anthropic, 2024; Google, 2024; OpenAI, 2024), remarkable progress has been achieved in enhancing their generative capabilities, particularly in the domain of reasoning. Throughout this development, well-designed discriminators play a crucial role, whether in aligning model outputs with human preferences (Schulman et al., 2017; Ziegler et al., 2019; Ouyang et al., 2022) or scaling model capabilities through effective reward signals (Guo et al., 2025; Yu et al., 2025; Seed, 2025). The emergence of DeepSeek-R1 (Guo et al., 2025) highlights the effectiveness of rule-based rewards in domains such as mathematics and code. However, in broader scenarios where golden answers are not readily available, learned discriminators remain indispensable for providing reliable reward signals (Seed, 2025; Liu et al., 2025).

Building on this insight, we explore methods to enhance the performance of discriminator models by focusing on the closely related task of classification. Traditional discriminative approaches (Ruan et al., 2024; Muennighoff et al., 2022; Cobbe et al., 2021; Yu et al., 2023) typically involve using a randomly initialized value head with a pre-trained language model to map text to labels, relying on the representation token to predict class probabilities. Although this method is widely adopted, it introduces an inherent mismatch between the randomly initialized value head and the carefully optimized language model, potentially leading to suboptimal performance (Zhang et al., 2024; Ye et al., 2024). This discrepancy may hinder the model from fully exploiting the generative capabilities already embedded within LLMs.

Recent advancements in prompt-based learning offer an alternative by guiding LLMs to perform classification through language generation (Parikh et al., 2023; Rouzegar and Makrehchi, 2024). This generative approach naturally aligns with the intrinsic training paradigm of LLMs, effectively leveraging their native language understanding and generation capabilities. Compared to traditional methods, this approach offers several advantages:

- Benefits from LLM Improvement: Generative methods enable classification tasks to benefit from ongoing advancements in LLM capabilities, naturally scaling classification accuracy with improved underlying LLM performance.

- Greater Flexibility: Generative methods allow the addition of classes without extensive training or altering model architecture. Traditional methods require adjusting dimensions and retraining when new labels are introduced.

Despite its intuitive appeal, most methods adopt a simple, and identical prompt strategy for both training and inference. The systematic exploration of diverse prompt strategies for both stages remains limited, with the effects of using different prompts during these stages not yet thoroughly investigated or quantified. To address this gap, we propose **GenCLS++**, a prompt-based generative classification framework that systematically explores five high-level strategy dimensions: In-Context Learning (ICL) variants (semantic retrieval vs. fixed exemplars, and varying shot counts), category definitions, explicit uncertainty labels, semantically irrelevant labels, and perplexity-based decoding, during both supervised fine-tuning and inference. Furthermore, inspired by recent advances, we integrate reinforcement learning (RL) into GenCLS++, resulting in additional performance gains and underscoring the potential of unifying supervised and reinforcement learning paradigms. We empirically evaluate GenCLS++ across seven diverse datasets, comprising both publicly available and internal data. Our findings reveal that GenCLS++ significantly enhances classification accuracy, achieving an average accuracy improvement of 3.46% relative to the commonly used naive SFT baseline. This improvement rises to 4.00% on public datasets.

Interestingly, our results challenge assumptions derived from related reasoning-intensive tasks. While explicit reasoning steps have shown significant performance improvements in such tasks, we find that classification tasks often achieve optimal results without explicit reasoning prompts, consistent with some studies (Li et al., 2025). These findings offer new insights into the role and necessity of explicit reasoning in classification contexts.

Our contributions can be summarized as follows:

- We conduct a comprehensive analysis of a wide range of prompt strategies for classification tasks. Our findings reveal that specific combinations can significantly outperform the naive SFT approach, highlighting their effectiveness in enhancing model performance.

- We integrate RL to further boost performance. Our experiments indicate that supervised fine-tuning for warm-up initialization delivers a significant relative improvement in accuracy, with an average gain of 18.18% compared to training directly from the base model.

- Motivated by recent investigations into reasoning-based inference and the observation that RL tends to produce shorter responses, we find that models achieve better performance on classification tasks by directly predicting answers without explicit reasoning steps.

2 Related Work

LLMs for Classification Compared to the traditional discriminative approach of employing a value head to map text to labels, recent studies have explored a generative strategy in which LLMs perform classification through prompt engineering (Qin et al., 2023; Sun et al., 2023; Peskine et al., 2023; Milios et al., 2023), augmenting the prompt with few-shot examples and category definitions. However, few studies have taken the next step of fine-tuning LLMs to generate class labels (Parikh et al., 2023), and several reports indicate that the generative approach underperforms on certain classification benchmarks (Ruan et al., 2024). Our study advances prior work by systematically examining various combinations of training-time and inference-time strategies. With this framework we achieve consistently higher accuracy across these datasets, providing strong evidence of the generative paradigm's potential for classification tasks.

RL for LLM Training Reinforcement learning (RL) now plays a pivotal role in training LLMs. It is used not only to align outputs with human preferences through Reinforcement Learning from Human Feedback (RLHF) (Ziegler et al., 2019; Ouyang et al., 2022), but also to enhance models' reasoning abilities, as recently demonstrated by DeepSeek-R1 (Guo et al., 2025). These applications underscore RL's potential to drive further advancements in LLMs. In this paper, we investigate how RL can improve performance on classification tasks and present several noteworthy empirical findings. Given that PPO (Schulman et al., 2017) incurs substantial computational overhead and rule-based

rewards have shown to be effective, we conduct most of our experiments using the more efficient Reinforce++ algorithm (Hu, 2025), along with carefully designed rule-based reward functions.

3 Method

Traditional LLM-based classifiers do not fully utilize the text generation capabilities of pretrained LLMs. To address this issue, we propose training generative classifiers using standard next token prediction. Specifically, instead of obtaining each category's probability through a representative token, the language model predicts categories using its own probability distribution over tokens. This approach preserves the model's generative abilities, since classification is merely another token prediction, while also offering several advantages that naturally arise from LLMs, such as a unified paradigm for pretraining and classification, and the ability to scale inference time compute. The overview of our method is shown in Figure 1.

3.1 Exploring Different Strategy Combinations in SFT and Inference

Let $\mathbf{x}$ denote the input to be classified. A generative classifier π_θ predicts the gold label $\mathbf{y}_{\text{gold}}$ using tokens. This is achieved by maximizing $\log \pi_\theta\big(\mathbf{y}_{\text{gold}}|(\mathbf{p},\mathbf{x})\big)$, where $\mathbf{p}$ represents a particular prompt strategy from the strategy pool $\mathcal{P}$. To do so, we minimize the supervised fine-tuning loss on the dataset $\mathcal{D}$, which contains input–class pairs:

$$\mathcal{L}_{\text{SFT}}(\theta,\mathcal{D}) = -\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\sum_{t=1}^{|\mathbf{y}|}\log \pi_\theta\big(y_t \mid \mathbf{p},\mathbf{x},\mathbf{y}_{<t}\big)\right].$$

We aim to explore how different prompt strategies, applied during both training and inference, affect classification performance within a purely generative paradigm. Below, we describe the types of prompt strategies in $\mathcal{P}$ that we employed; examples of each type are provided in Appendix A.

Zero-shot The model receives only a general task description (e.g., "Classify the following text"), without labeled examples or detailed definitions.

N-shot We include N labeled examples (randomly selected from the training set) in the prompt $\big(N \in \{1,3,5\}\big)$, providing the model with exemplar input–label pairs to guide classification.

Fixed-3-shot The same three labeled examples appear in the prompt for every test case.

Similar-3-shot We retrieve the three training examples most similar to the new input (based on textual similarity) and include them in the prompt to provide more contextually relevant guidance.

Definition We prepend concise text definitions of each category to the prompt, allowing the model to reference these definitions when generating predictions. These definitions are generated by prompting a LLM to provide explanations for each class label. For example, for an emotion classification task with label "anger", the generated definition is: "anger: contains strong negative feelings like anger, annoyance, indignation, involving injustice, conflict, frustration, etc." These definitions are then incorporated into the prompt to provide clear semantic meanings of the categories before classification. For our implementation, we use "GPT-4o-2024-11-20" as the LLM to generate these definitions.

Definition with 1-shot In addition to including category definitions, we add a single labeled example to the prompt for further guidance.

Numerical (semantically irrelevant labels) We assign each category a numerical label and prompt the model to output the corresponding number. This approach is non-semantic, relying purely on arbitrary number assignments rather than meaningful category descriptions.

Uncertainty We introduce an "Uncertain" category for any example misclassified by our two top zero-shot models. Each example is relabeled as "Uncertain" if both models disagree with its ground truth; otherwise it keeps its original label. To avoid overloading the training data, we cap the uncertain cases at 10% of the corpus. If more than 10% qualify, we retain only those with the lowest average prediction confidence. This yields a modified dataset containing up to 10% ambiguous examples alongside the correctly labeled ones. We then fine-tune the base model on this set using a training prompt that includes the "Uncertain" option. At inference, the classifier is restricted to the original label set and will never output "Uncertain".

Perplexity For each candidate class $\mathbf{y}_i$, we append it to the input and compute its perplexity $\text{PPL}(\mathbf{y}_i)$ as follows:

$$\text{PPL}(\mathbf{y}_i) = \exp\Big\{-\frac{1}{|\mathbf{y}_i|}\sum_{t=1}^{|\mathbf{y}_i|}\log \pi_\theta\big(y_t|\mathcal{P}_{\text{base}}(\mathbf{x}),\mathbf{y}_{<t}\big)\Big\}$$

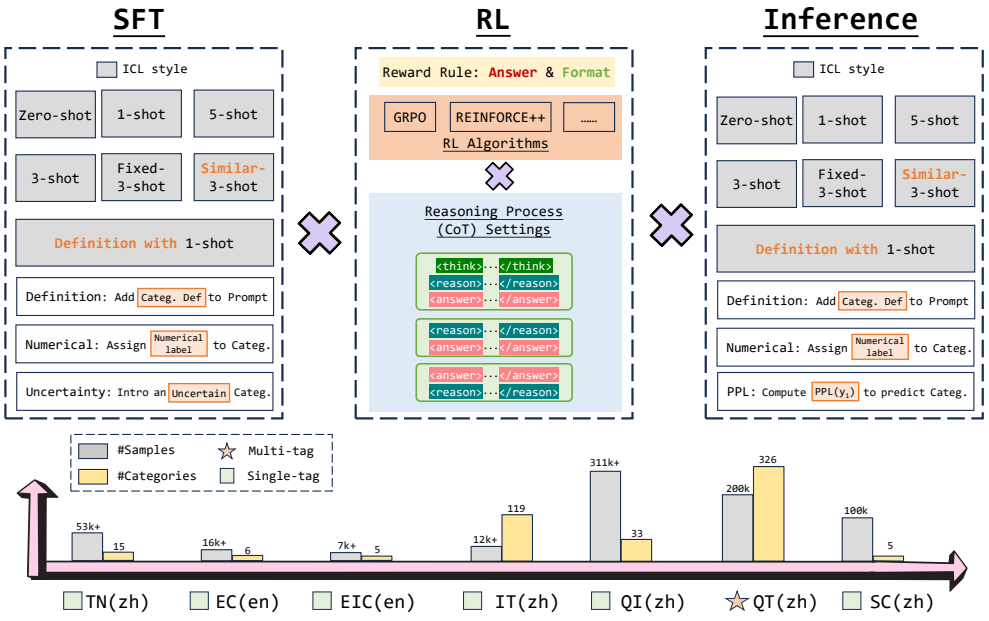

Figure 1: An overview of the GenCLS++ framework. It explores diverse combinations of training and inference strategies for classification tasks and incorporates RL to further enhance performance. We conduct comprehensive experiments on seven datasets, encompassing different languages, varying numbers of categories and data types.

where $\mathcal{P}_{\text{base}}(\mathbf{x})$ is a base prompt. We then select the class with the lowest perplexity as our prediction:

$$\hat{y} = \arg\min_{\mathbf{y}_i \in \mathcal{C}} \text{PPL}(\mathbf{y}_i)$$

This strategy is employed only at inference time.

We apply various strategies to train the model and subsequently evaluate each resulting model with different prompt strategies (e.g., trained with definitions, evaluated in a zero-shot setting), as illustrated in Figure 1. In contrast to traditional few-shot learning, which uses the same prompt type for both training and inference, our approach enables a more fine-grained analysis of how different strategies affect performance at each stage.

3.2 Reinforcement Learning

Building on the success of DeepSeek-R1 (Guo et al., 2025), which shows that reinforcement learning (RL) can markedly enhance the reasoning ability of language models, we explore RL for generative classification. Specifically, we fine-tune our model with a rule-based reward function to gauge the effectiveness of RL in this setting.

3.2.1 Policy Warm-up

During the warm-up phase, we equip the policy model with foundational classification capabilities by performing supervised fine-tuning on the dataset $\mathcal{D}$. We find that this phase has a significant impact on the subsequent performance of RL. Furthermore, we investigate how different start models affect the final performance. Detailed results and discussions are presented in Sections 5.1 and 5.2.

3.2.2 RL with Reasoning

System Prompt We first follow Guo et al. (2025)'s paradigm, encouraging models to engage in a reasoning (thinking) process before producing the final answer. The prompt is defined as follows: "Please output your answer in the format: <reason> reasoning process here </reason> <answer> answer here </answer>."

Reward Function Similarly, we design a two-part rule-based reward function: format reward and accuracy reward. The format reward verifies that the response follows the required structured format, ensuring that every part appears in the correct order and is enclosed in the appropriate tags:

$$R_{\text{format}} = \begin{cases} 1, & \text{if the format is correct,} \\ 0, & \text{otherwise.} \end{cases}$$

The accuracy reward measures whether the model's prediction matches the gold label $\mathbf{y}_{\text{gold}}$:

$$R_{\text{accuracy}} = \begin{cases} 1, & \text{if } y = \mathbf{y}_{\text{gold}}, \\ 0, & \text{otherwise.} \end{cases}$$

The final reward function R is a combination of the two rewards: $R = R_{\text{format}} + R_{\text{accuracy}}$

3.2.3 RL without Reasoning

Unlike reasoning-driven tasks such as mathematics and code generation, we observed that during the RL process in classification tasks, the response length fluctuates and may even decrease rapidly. Comparative experiments further revealed that the inclusion of rationale does not seem to contribute to performance improvement, as discussed in Section 5.2. This phenomenon has also been observed in other tasks, such as commonsense question answering (Jiang et al., 2025; Sprague et al., 2024; Sui et al., 2025) and vision classification (Li et al., 2025). These findings suggest that chain-of-thought (CoT) reasoning may not be essential for all tasks. Motivated by these insights, we investigate RL in classification tasks without reasoning.

System Prompt Unlike most current RL-based scaling methods, which encourage models to repeatedly reason and verify, the prompt in our method directs the model to output the result directly, e.g., "Please output your answer."

Reward Function Since we no longer need to distinguish between reasoning and the answer, we eliminate the need for conventional format rewards. Instead, we solely use an accuracy reward, which checks whether the model's output matches the ground truth exactly. It is defined as follows:

$$R_{\text{accuracy}} = \begin{cases} 1, & \text{if } y = \mathbf{y}_{\text{gold}}, \\ 0, & \text{otherwise.} \end{cases}$$

We adopt Reinforce++ (Hu, 2025) as our reinforcement learning algorithm. In Section 5.3, we compare it with several widely used baselines, such as GRPO (Shao et al., 2024), and find that Reinforce++ consistently delivers higher accuracy while requiring less training time, demonstrating advantages in both performance and efficiency.

4 Experiments

4.1 Experimental Setup

Datasets We conducted comprehensive experiments on seven datasets, including four public benchmarks (EC, EIC, IFLYTEK, and TNEWS) and three proprietary datasets (Query Intent, Search Correlation, and Query Taxonomy). EC focuses on sentiment detection, whereas EIC classifies the type of edits between sentence pairs, a task on which generative classifiers have previously performed poorly (Ruan et al., 2024). IFLYTEK assigns app descriptions to as many as 120 categories, and TNEWS categorizes news headlines by topic; both are widely used multi-class benchmarks. Our proprietary datasets further extend the evaluation: Query Intent (QI) predicts user intent at both coarse and fine granularities across roughly 30 labels, Search Correlation (SC) evaluates the relevance between a query and a text passage, and Query Taxonomy (QT) performs multi-label semantic tagging, since a single query may map to multiple categories. More detailed descriptions of all datasets are provided in Appendix B.

Metrics The performance of our models is evaluated using accuracy (Acc.) and macro-F1. Accuracy measures the ratio of correct predictions to total predictions, while macro-F1 is the average of per-class F1-scores, assigning equal weight to each class. For each inference strategy, we report five metrics: *fmt-suc ratio* (percentage of format-matched outputs), *fmt-suc accuracy* and *fmt-suc macro-F1* (computed only on format-matched outputs), and *overall accuracy* and *overall macro-F1* (calculated over all predictions). The **overall accuracy** and **overall macro-F1** serve as the primary indicators of task performance. When *fmt-suc ratio* is less than 100%, format-success metrics are highlighted only if their corresponding overall metrics also achieve best performance.

Parameter Setting We used Qwen-2.5-7B-Instruct (Yang et al., 2024) in our experiments. This open-source model achieves non-trivial performance on the classification task while still leaving room for improvement, making it an ideal testbed for our study. We constructed the training dataset using the prompt strategy described in Section 3.1 and tested each trained model across all these prompt types, yielding approximately 10×10 total combinations. For RL, we used Reinforce++ and its training framework OpenRLHF (Hu, 2025).

Baselines Since our approach employs generative classification, we adopt the traditional discriminative method using value head on public datasets as a robust baseline. Specifically, we utilize the results reported by (Ruan et al., 2024) for the EC and EIC datasets, and by (Xu et al., 2020) for the IFLYTEK and TNEWS datasets. Additionally, to illustrate that combining different prompt strategies during training and inference can yield superior performance, we introduce an additional commonaly used naive SFT baseline, using a zero-shot prompt

Table 1: Experimental Results (Accuracy & macro-F1 Score, %). Gray indicates that the training strategy is not aligned with the best inference strategy. **Bold** indicates the best result, and underline indicates the second best. * reported by Ruan et al. (2024); †reported by Xu et al. (2020).

(a) Part 1: Results on Public Datasets

	EC		EIC		IFLYTEK		TNEWS	
Training Method	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1
Base model	68.75	33.94	55.02	49.85	57.41	25.93	58.13	22.92
Discriminative method	94.10*	89.60*	84.40*	82.20*	62.98†	-	59.46†	-
Naive SFT	93.70	90.17	82.74	81.73	61.18	45.42	60.83	59.39
Zero-shot	93.75	90.31	84.04	82.93	62.83	46.69	62.12	59.16
1-shot	93.15	89.45	83.82	82.90	63.33	47.15	61.98	60.49
3-shot	93.45	89.13	85.03	83.75	62.91	48.51	62.06	60.53
5-shot	94.15	89.83	84.13	83.39	62.52	44.93	62.54	58.41
Fixed-3-shot	93.80	89.92	83.17	82.49	63.52	45.26	62.25	58.94
Similar-3-shot	93.90	89.44	82.18	79.35	62.83	47.69	63.30	61.31
Definition	93.30	88.31	83.26	82.30	63.64	44.41	61.37	59.26
Definition with 1-shot	93.80	89.70	84.34	82.79	63.37	47.47	62.20	60.65
Numerical	93.65	89.93	83.17	81.89	62.29	46.29	61.24	57.09
Uncertainty	93.55	90.01	85.08	83.74	63.76	47.85	61.97	58.15
GenCLS++ (RL)	**94.50**	**90.57**	**85.86**	**84.72**	**64.91**	**49.27**	**64.04**	**62.35**

(b) Part 2: Results on Proprietary Datasets

	Query Intent		Search Correlation		Query Taxonomy	
Training Method	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1
Base model	74.91	18.59	41.91	34.24	26.51	14.70
Naive SFT	92.28	86.33	67.43	58.64	51.43	43.10
Zero-shot	92.30	86.27	65.27	54.44	53.25	43.13
1-shot	92.48	**87.34**	67.37	59.10	53.09	44.02
3-shot	92.44	86.55	67.53	59.23	53.38	43.84
5-shot	92.52	86.95	66.73	56.42	53.95	44.52
Fixed-3-shot	92.36	86.17	64.76	50.08	54.03	43.60
Similar-3-shot	92.22	86.40	67.63	60.01	52.99	44.11
Definition	92.23	85.91	68.60	62.25	-	-
Definition with 1-shot	92.41	78.02	67.40	58.51	-	-
Numerical	92.52	86.34	64.40	47.89	51.26	42.87
Uncertainty	92.36	86.51	65.57	53.18	50.21	38.35
GenCLS++ (RL)	**92.62**	86.86	**68.94**	**65.08**	**54.31**	**46.18**

strategy for both stages (i.e. training and evaluating the model exclusively with the zero-shot prompt).

4.2 Main Results

We adopt a generative paradigm based on a LLM. The base model is fine-tuned with various prompt strategies and evaluated under each strategy at inference time. Since a dataset can yield nearly one hundred training–inference combinations, Table 1 reports only the best result for each training strategy to enable comprehensive analysis. The full combination results are provided in Appendix C.

As shown in Table 1, GenCLS++ surpasses every discriminative baseline on the public datasets, underscoring the strength of generative approaches to classification. Moreover, for most training prompts, **switching to an alternative inference prompt yields additional gains in both accuracy and macro-F1**. Figure 4 visualizes these improvements, demonstrating that—**regardless of the strategy employed during training—experimenting with a different inference strategy typically leads to superior performance**. Moreover, applying RL to a model that has already undergone SFT yields additional gains. Although the naive SFT baselines for EC and Query Intent are already strong, exceeding 90% accuracy, GenCLS++ still achieves an average relative accuracy improvement of 3.46% on all seven datasets and 4.00% on the four public datasets. Notably, GenCLS++ delivers a 6.10% relative accuracy improvement on the IFLYTEK dataset, underscoring its effectiveness.

Further analysis reveals a consistent pattern: adding labeled examples to the training prompt

Table 2: Experimental Results (Accuracy & macro-F1 Score, %). **Bold** indicates the best result.

	EC		EIC		IFLYTEK		TNEWS	
Training Method	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1
Base model	68.75	33.94	55.02	49.85	57.41	25.93	58.13	22.92
+ RL	76.90	69.52	62.28	47.87	59.79	29.88	61.92	59.56
+ warm up	94.15	89.83	85.08	83.74	63.76	47.85	63.30	61.31
+ RL	**94.50**	**90.57**	**85.86**	**84.72**	**64.91**	**49.27**	**64.04**	**62.35**

Table 3: Experimental Results on the EIC dataset (Accuracy & macro-F1 Score, %). Blue indicates the best inference strategy for current training method. **Bold** indicates the best result.

	Class→Reason		Reason→Class		Think→Reason→Class	
Training Method	Acc.	macro-F1	Acc.	macro-F1	Acc.	macro-F1
Base model	17.95	13.86	30.54	5.69	29.46	7.27
+ RL	53.33	42.46	62.28	47.87	46.58	19.60
SFT (Class→Reason)	76.73	76.12	57.05	51.46	57.40	52.34
SFT (Think→Reason→Class)	37.98	36.15	53.81	47.14	57.35	50.45
SFT (Reason→Class)	64.62	64.67	71.37	66.26	70.11	45.60
+ RL	77.12	76.00	79.50	78.06	78.63	76.00
SFT (No-CoT)	76.60	75.64	69.81	62.03	74.57	33.76
+ RL	**85.86**	**84.72**	85.64	84.52	84.60	83.26
+ SFT (Reason→Class)	63.62	52.13	76.08	73.63	-	-
+ RL	0.43	0.43	84.04	82.67	33.48	31.90

(few-shot learning) consistently outperforms training without examples, and this advantage holds in both zero-shot and few-shot evaluations. Additionally, randomly sampled examples yield higher scores than a fixed set of examples. Although the optimal inference prompt is not always identical to the training prompt, two clear tendencies emerge: 1) If the training prompt includes few-shot examples, the highest scores are achieved when the inference prompt also provides examples. 2) If the training prompt omits examples, a zero-shot inference prompt is usually the stronger choice.

These findings underscore that prompt design should be considered jointly for training and inference, rather than in isolation. Furthermore, using reinforcement learning to further enhance the performance of generative models on classification tasks is a promising approach, which we will analyze in more detail in Section 5.

5 Analysis

5.1 Effectiveness of the Policy Warm-up

To equip the policy model with fundamental classification capabilities, we first apply fine-tuning on the training data, which we refer to as the "warm-up" phase. We then conduct an ablation study using Qwen-2.5-7B-Instruct on several public benchmarks to evaluate the impact of this phase. As reported in Table 2, incorporating a warm-up phase provides a significant performance boost in subsequent RL training, with an average relative accuracy improvement of 18.18% over initializing RL directly from the base model. This indicates that allowing the policy model to acquire essential classification skills through supervised fine-tuning (SFT) before RL effectively raises the ceiling for achievable performance, demonstrating the importance of pre-training in enhancing RL outcomes.

5.2 Does the reasoning help classification?

In this subsection, we present further discussions on the reasoning process in fine-tuning for classification. We explored the following settings: <Class→Reason>, <Reason→Class>, and <Think→Reason→Class>, which define the order in which the model outputs its responses. For example, in the <Reason→Class> setting, the model first explains its reasoning and then predicts the classification result. Here, “Think” represents a longer, more elaborate thought, while “Reason” represents a more concise explanation.

We chose the EIC dataset for our research because identifying the categories of editing intent requires comparing the changes before and after sentences, which requires the model to use

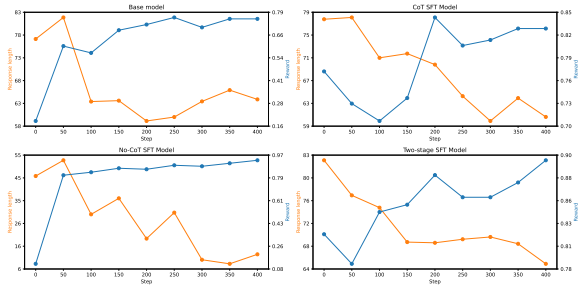

Figure 2: Comparison of Different Models: Response Length vs. Reward over Steps

reasoning ability more than other classification tasks. We used DeepSeek-R1 (Guo et al., 2025) to generate reasoning for the training set in the <Think→Reason→Class> format and then manually converted these outputs into the three settings mentioned above. For each question, we sampled three times; the reasoning process was considered valid only if all three outputs were correct. We then performed supervised fine-tuning (SFT) of the base model on these three datasets, and the results are shown in Table 3. Interestingly, contrary to intuition, having the model provide its classification result first led to higher accuracy compared to the other two approaches.

To delve deeper into the reasoning process for classification, we ran RL from several starting checkpoints: the base model, a SFT model without reasoning (No-CoT SFT), an SFT model with reasoning (CoT SFT), and a two-stage model obtained by further fine-tuning the No-CoT SFT model on CoT data. The results indicate the following: 1) Regardless of the starting model, RL consistently enhances classification performance. 2) Before applying CoT data for SFT, conducting an initial SFT stage using No-CoT data leads to better performance improvements following RL. 3) Interestingly, the best performance is attained by directly applying RL to the No-CoT SFT model and allowing the model to predict the answer without performing reasoning beforehand.

We subsequently examined how the model's output length and behavior evolved during RL training. As shown in Figure 2, the length of the generated "reason" gradually decreased, indicating that the model pruned away unnecessary reasoning steps. These results suggest that the model progressively learns to simplify its reasoning and that extensive deliberation is not always beneficial for producing correct answers. In classification tasks in particular, the RL signal appears to teach the model that directly producing the answer is sufficient.

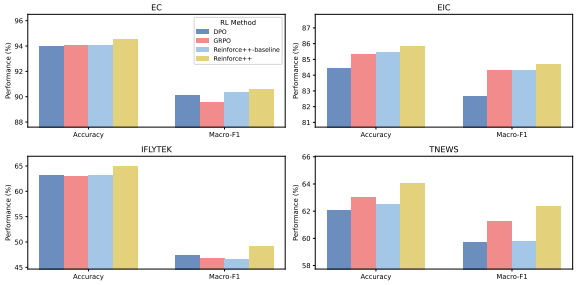

Figure 3: Comparison of model performance across different RL algorithms.

5.3 Different RL Algorithms

We further explored the impact of different RL algorithms on model performance by analyzing GRPO (Shao et al., 2024), Reinforce++-baseline (Hu, 2025), and Reinforce++ (Hu, 2025), with DPO (Rafailov et al., 2023) included as an off-policy comparator. As shown in Table 3, all on-policy methods outperform DPO. Notably, Reinforce++ yields the largest gains and, unlike GRPO, requires no batch sampling of candidate responses during training—making it the most efficient choice.

6 Conclusion

In this paper, we investigate the use of LLMs as generative classifiers. By systematically exploring a variety of prompt strategies during both training and inference, coupled with the integration of RL, we enhance the intrinsic generative capabilities of LLMs for classification tasks. GenCLS++ achieves an average relative accuracy improvement of +3.46% across seven benchmark datasets compared to the naive SFT baseline. Notably, our experiments show that while explicit reasoning steps enhance performance on complex tasks, they do not yield significant benefits in classification settings. In future work, we aim to evaluate whether these findings generalize to models of varying scales and to explore novel techniques that can further push the performance limits of generative classifiers.

Limitations

While GenCLS++ is effective for classification tasks, its potential applications as a verifier or reward model were not explored in this work. Furthermore, because we employed Qwen-2.5-7B-Instruct,

we have not tested whether our findings generalize to models of different scales or to other architectures, such as LLaMA. We leave these investigations, along with the exploration of novel techniques to further advance the performance of generative classifiers, for future work.

References

Anthropic. 2024. Claude 3.5 sonnet. https://www.anthropic.com/news/claude-3-5-sonnet.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Google. 2024. Our next-generation model: Gemini 1.5. https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*.

Mingqian He, Yongliang Shen, Wenqi Zhang, Zeqi Tan, and Weiming Lu. 2024. Advancing process verification for large language models via tree-based preference learning. *arXiv preprint arXiv:2407.00390*.

Ruixin Hong, Hongming Zhang, Xinyu Pang, Dong Yu, and Changshui Zhang. 2023. A closer look at the self-verification abilities of large language models in logical reasoning. *arXiv preprint arXiv:2311.07954*.

Jian Hu. 2025. Reinforce++: A simple and efficient approach for aligning large language models. *arXiv preprint arXiv:2501.03262*.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2023. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanwei Li, Yu Qi, Xinyan Chen, Liuhui Wang, Jianhan Jin, Claire Guo, Shen Yan, and 1 others. 2025. Mme-cot: Benchmarking chain-of-thought in large multimodal models for reasoning quality, robustness, and efficiency. *arXiv preprint arXiv:2502.09621*.

Ming Li, Shitian Zhao, Jike Zhong, Yuxiang Lai, and Kaipeng Zhang. 2025. Cls-rl: Image classification with rule-based reinforcement learning. *arXiv preprint arXiv:2503.16188*.

Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. 2025. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*.

Aristides Milios, Siva Reddy, and Dzmitry Bahdanau. 2023. In-context learning for text classification with many labels. *arXiv preprint arXiv:2309.10954*.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

OpenAI. 2024. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Soham Parikh, Quaizar Vohra, Prashil Tumbade, and Mitul Tiwari. 2023. Exploring zero and few-shot techniques for intent classification. *arXiv preprint arXiv:2305.07157*.

Youri Peskine, Damir Korenčić, Ivan Grubisic, Paolo Papotti, Raphael Troncy, and Paolo Rosso. 2023. Definitions matter: Guiding gpt for multi-label classification. In *EMNLP 2023, Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Jie Ren, Yao Zhao, Tu Vu, Peter J Liu, and Balaji Lakshminarayanan. 2023. Self-evaluation improves selective generation in large language models. In *Proceedings on*, pages 49–64. PMLR.

Hamidreza Rouzegar and Masoud Makrehchi. 2024. Enhancing text classification through llm-driven active learning and human annotation. *arXiv preprint arXiv:2406.12114*.

Qian Ruan, Ilia Kuznetsov, and Iryna Gurevych. 2024. Are large language models good classifiers? a study on edit intent classification in scientific document revisions. *arXiv preprint arXiv:2410.02028*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

ByteDance Seed. 2025. Seed-thinking-v1.5: Advancing superb reasoning models with reinforcement learning. https://github.com/ByteDance-Seed/Seed-Thinking-v1.5.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Hanjie Chen, Xia Hu, and 1 others. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023. Text classification via large language models. *arXiv preprint arXiv:2305.08377*.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, and 1 others. 2020. Clue: A chinese language understanding evaluation benchmark. *arXiv preprint arXiv:2004.05986*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Ziyi Ye, Xiangsheng Li, Qiuchi Li, Qingyao Ai, Yujia Zhou, Wei Shen, Dong Yan, and Yiqun Liu. 2024. Beyond scalar reward model: Learning generative judge from preference data. *arXiv preprint arXiv:2410.03742*.

Fei Yu, Anningzhe Gao, and Benyou Wang. 2023. Ovm, outcome-supervised value models for planning in mathematical reasoning. *arXiv preprint arXiv:2311.09724*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. 2024. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Prompt Example

To illustrate, we use the EC dataset to showcase the prompt strategies outlined in Section 3.1.

Zero-shot & Few-shot We adopt the 3-shot setting as a representative example for both the zero-shot and few-shot series.

> You are a professional sentiment classification expert. There is now a piece of text that requires your sentiment classification.
> Optional categories: [sadness, joy, love, anger, fear, surprise]
> Format requirement: Please output in the format `Category: xxx` (where `xxx` is the corresponding category label).
>
> **Example 1:**
> Text: i feel now i am not giving all of me to christ and i want to be devoted
> Category: love
> **Example 2:**
> Text: i find myself feeling shocked hearing that word spoken out loud in my own lounge room
> Category: surprise
> **Example 3:**
> Text: i feel pathetic and that i shouldn't make myself feel this way
> Category: sadness
>
> **Current case:**
> Text: i feel like a low life mooching off everyone
> Please output the category for the text according to the format requirement.

Definition We prepend concise text definitions of each target category to the prompt and further explore an alternative strategy by adding an extra example.

> You are a professional sentiment classification expert. There is now a piece of text that requires your sentiment classification.
> Optional categories: [sadness, joy, love, anger, fear, surprise]
> Format requirement: Please output in the format `Category: xxx` (where `xxx` is the corresponding category label).
>
> Sentiment category definitions:
> **sadness**: expresses loss, sorrow, frustration, etc., involving farewells, failures, regrets, etc.
> **joy**: conveys happiness, cheerfulness, satisfaction, etc., including celebrations, success, and pleasure from good things.
> **love**: reflects romantic love, familial love, friendship, etc., involving care, admiration, attachment, etc.
> **anger**: contains strong negative feelings like anger, annoyance, indignation, involving injustice, conflict, frustration, etc.
> **fear**: shows fear, worry, anxiety, etc., involving danger, uncertainty, psychological pressure, etc.
> **surprise**: expresses the unexpected, astonishment, amazement, etc., including sudden events or information beyond expectations.
>
> **Example 1:**
> Text: i walk in the door to my house i feel happy
> Category: joy
>
> **Current case:**
> Text: i was feeling like a beluga whale and quite grouchy
> Please output the category for the text according to the format requirement.

Numerical For this type, we assign a numerical label to each category and instruct the model to output the corresponding number.

> You are a professional sentiment classification expert. There is now a piece of text that requires your sentiment classification.
> Optional categories: sadness: 0, joy: 1, love: 2, anger: 3, fear: 4, surprise: 5
> Format requirement: Please output in the format `Category: xxx` (where `xxx` is the corresponding numeric label).
>
> **Current case:**
> Text: i reply i do my best to reply to questions but feel free to contact me via twitter isobelmeg xx
> Please output the category for the text according to the format requirement.

Uncertainty We introduce a new class, "Uncertain," and employ a fine-tuned model to label training examples that cannot be classified with high confidence. This prompt strategy is used exclusively during dataset construction; during inference, the model is restricted to predicting only the original classes.

> You are a professional sentiment classification expert. There is now a piece of text that requires your sentiment classification.
> Optional categories: [sadness, joy, love, anger, fear, surprise]
> Format requirement: Please output in the format `Category: xxx` (where `xxx` is the corresponding category label; if unsure, please reply `"Category: uncertain"`).
>
> **Current case:**
> Text: i forgive myself that i have accepted and allowed myself to forget that i decide and thus i was decided to feel groggy this morning
> Please output the category for the text according to the format requirement.

B Data Statistics

The detailed statistics for all datasets are shown in Table 4 and 5

C SFT Results

C.1 Performance Gains from Prompt Strategy Switching

We visualize the improvement brought by using the best-performing inference prompt strategy compared to reusing the same strategy as in training in

Figure 4. As shown, our approach consistently improves both accuracy and macro-F1 across nearly all tasks and training prompt strategies.

C.2 Effect of Retrieval Relevance in Few-Shot Inference

We further compared the impact of two different inference strategies on model performance. As shown in Figure 5, using similar few-shot examples does not always lead to better results across all tasks. We found that when the retrieval strategy is related to the category, as is the case with the TNEWS dataset, the retrieved examples can enhance performance. However, for tasks involving relationships between multiple texts, where the retrieval strategy is unrelated to classification, including seemingly similar examples may actually degrade performance.

C.3 Perplexity-Based Strategy Evaluation

Recent methods (Hao et al., 2023; Ren et al., 2023) adopt perplexity as a confidence score for LLMs—for example, using the probability of the "A" token to gauge answer confidence in multiple-choice questions. Similarly, we compared this perplexity-based strategy with a fixed 3-shot prompt. As shown in Figure 6, relying on perplexity markedly degrades model performance on most tasks, which is consistent with prior findings (Huang et al., 2023; Hong et al., 2023; He et al., 2024). In other words, perplexity alone is not a sufficiently reliable confidence measure.

C.4 Performance Across Training–Inference Combinations

The performance of different training and inference combinations for each dataset is presented in the tables below.

Table 4: Average token length statistics across different tasks and prompt types. Both prompt tokens and response tokens represent average values per example.

Dataset	Zero-shot		1-shot		3-shot		5-shot		Definition	
	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens
Query Intent	282.4	6.0	305.9	6.0	344.8	6.0	383.8	6.0	719.4	6.0
Query Taxonomy	1225.4	7.7	1250.5	7.7	1292.6	7.7	1334.8	7.7	-	-
Search Correlation	458.6	5.2	843.0	5.2	1605.9	5.2	2367.5	5.2	1261.6	5.2
EIC	141.6	4.1	222.1	4.1	375.9	4.1	523.6	4.1	287.6	4.1
EC	86.0	3.5	121.4	3.5	184.4	3.5	246.9	3.5	229.0	3.5
TNEWS	119.8	4.7	152.3	4.7	209.3	4.7	266.4	4.7	631.8	4.7
IFLYTEK	786.9	3.6	984.3	3.6	1370.2	3.6	1761.1	3.6	3481.9	3.6
Dataset	**Numerical**		**Similar-3-shot**		**Fixed-3-shot**		**Uncertainty**		**1-shot w/ Def**	
	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens	Prompt tokens	Response tokens
Query Intent	446.4	3.7	344.2	6.0	342.4	6.0	291.4	6.0	741.9	6.0
Query Taxonomy	2746.4	6.9	1292.0	7.7	1294.4	7.7	1225.4	6.6	-	-
Search Correlation	475.6	4.0	1921.0	5.2	1740.6	5.2	473.1	5.1	1653.0	5.2
EIC	155.6	3.0	400.6	4.1	596.5	4.1	145.6	4.1	368.1	4.1
EC	103.0	3.0	163.4	3.5	186.0	3.5	95.0	3.4	264.4	3.5
TNEWS	193.8	5.0	210.7	4.7	194.8	4.7	128.8	4.5	665.3	4.7
IFLYTEK	1272.9	4.0	1476.9	3.6	1215.9	3.6	795.9	3.5	3682.3	3.6

Table 5: Dataset Statistics

Dataset	Train samples	Test samples	Class count	Classification type
Query Intent	100,000	10,000	33	Single-label
Query Taxonomy	200,000	10,000	326	Multi-label
Search Correlation	311,446	2,997	5	Single-label
EIC	7,478	2,312	5	Single-label
EC	16,000	2,000	6	Single-label
TNEWS	53,360	10,000	15	Single-label
IFLYTEK	12,133	2,599	119	Single-label

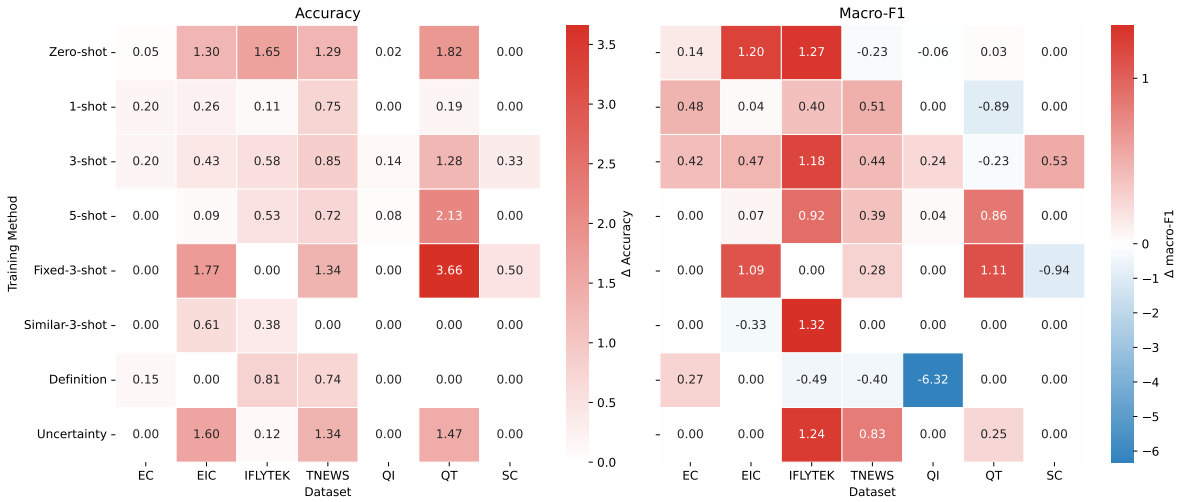

Figure 4: Visualization of the improvement achieved by changing the inference prompt strategy. Left: improvement in accuracy. Right: improvement in macro-F1. The x-axis represents the dataset, and the y-axis represents the different training strategies. Improvements are highlighted in red, while decreases are shown in blue.

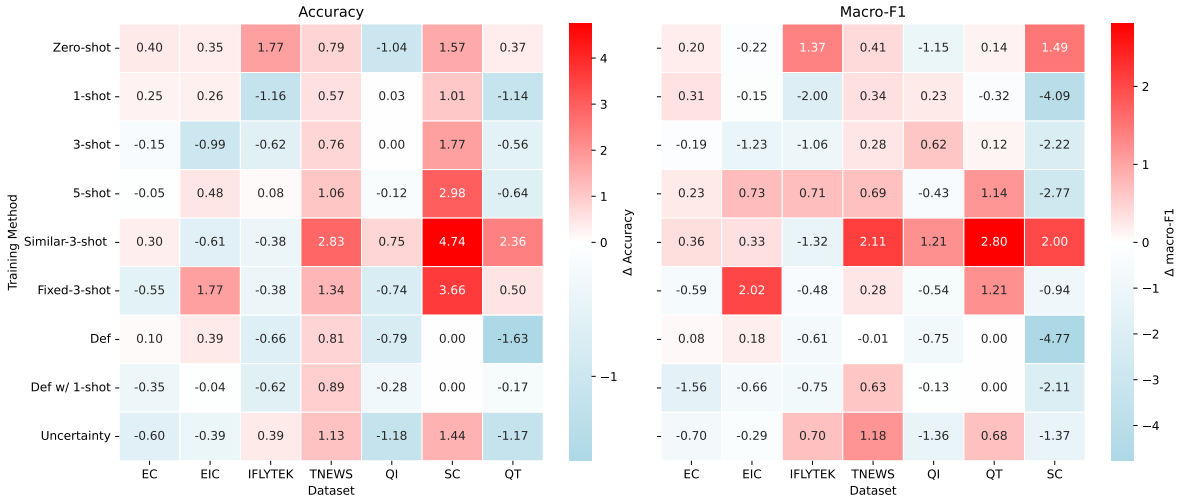

Figure 5: Visualization of the improvement achieved by changing the inference prompt strategy from fixed-3-shot to similar-3-shot. Left: improvement in accuracy. Right: improvement in macro-F1. The x-axis represents the dataset, and the y-axis represents the different training strategies. Improvements are highlighted in red, while decreases are shown in blue.

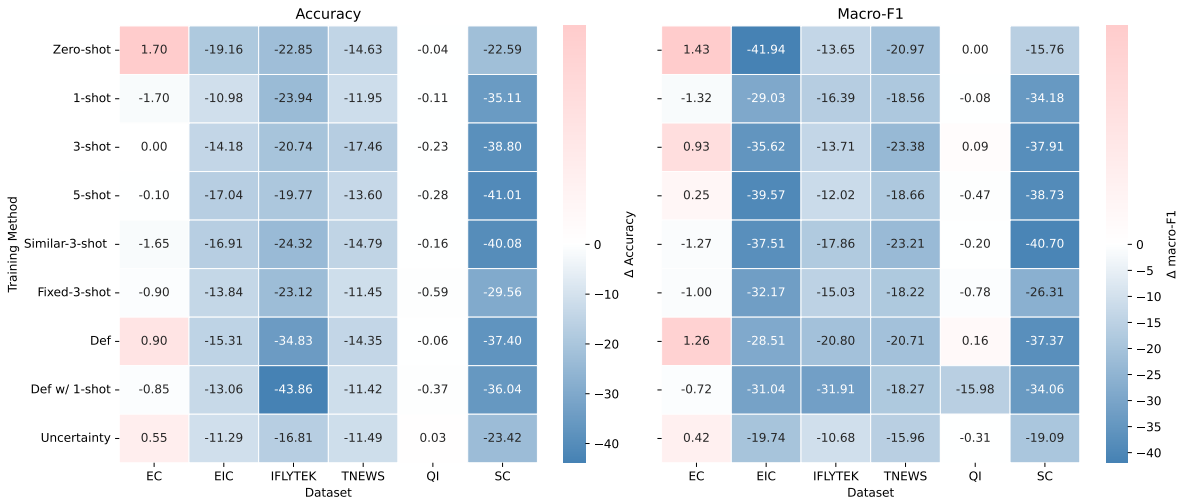

Figure 6: Visualization of the improvement achieved by changing the inference prompt strategy from fixed-3-shot to perplexity. Left: improvement in accuracy. Right: improvement in macro-F1. The x-axis represents the dataset, and the y-axis represents the different training strategies. Improvements are highlighted in red, while decreases are shown in blue.

Table 6: Experimental results on the EIC dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	83.43	82.40	83.43	82.40	100	83.39	82.67	83.39	82.67	100	82.74	82.22	82.74	82.22
1-shot	100	83.56	82.86	83.56	82.86	100	83.82	82.90	83.82	82.90	100	82.74	81.95	82.74	81.95
3-shot	100	84.08	82.88	84.08	82.88	100	84.60	83.28	84.60	83.28	100	85.03	83.75	85.03	83.75
5-shot	100	83.82	82.91	83.82	82.91	100	84.13	83.39	84.13	83.39	100	83.39	82.61	83.39	82.61
Definition	100	83.09	82.15	83.09	82.15	100	83.09	82.11	83.09	82.11	100	82.27	81.25	82.27	81.25
Numerical	100	38.41	4.83	38.41	4.83	100	48.18	6.61	48.18	6.61	100	50.56	7.69	50.56	7.69
Similar-3-shot	100	81.19	78.28	81.19	78.28	100	81.88	79.26	81.88	79.26	100	82.18	79.35	82.18	79.35
Fixed-3-shot	100	81.75	80.64	81.75	80.64	100	82.01	81.02	82.01	81.02	100	81.40	80.47	81.40	80.47
Uncertainty	100	83.48	81.13	83.48	81.13	100	84.39	82.65	84.39	82.65	100	84.26	82.13	84.26	82.13
1-shot w/ Def	100	84.34	82.79	84.34	82.79	100	83.95	82.44	83.95	82.44	100	83.65	82.11	83.65	82.11

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	84.04	82.93	84.04	82.93	100	82.22	81.44	82.22	81.44	0	–	–	–	–
1-shot	100	83.61	82.85	83.61	82.85	100	83.09	81.99	83.09	81.99	0	–	–	–	–
3-shot	100	84.95	**84.01**	84.95	**84.01**	100	81.79	20.30	81.79	20.30	0	–	–	–	–
5-shot	100	84.04	83.32	84.04	83.32	100	83.87	82.55	83.87	82.55	2.51	82.76	31.31	2.08	0.79
Definition	100	83.09	82.19	83.09	82.19	100	83.26	82.30	83.26	82.30	26.38	82.13	34.14	21.67	9.01
Numerical	100	54.67	10.89	54.67	10.89	100	12.41	8.79	12.41	8.79	100	83.17	81.89	83.17	81.89
Similar-3-shot	100	81.75	78.96	81.75	78.96	100	80.19	77.20	80.19	77.20	19.20	76.80	48.53	14.75	9.32
Fixed-3-shot	100	82.53	81.73	82.53	81.73	100	81.92	81.00	81.92	81.00	1.04	79.17	29.46	0.82	0.31
Uncertainty	100	84.60	82.77	84.60	82.77	100	84.47	83.09	84.47	83.09	0	–	–	–	–
1-shot w/ Def	100	84.17	82.47	84.17	82.47	100	84.04	83.16	84.04	83.16	0.26	83.33	45.45	0.22	0.12

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	83.09	82.00	83.09	82.00	100	82.74	81.73	82.74	81.73	100	63.58	40.28	63.58	40.28
1-shot	100	83.00	81.80	83.00	81.80	100	83.04	81.95	83.04	81.95	100	71.76	52.92	71.76	52.92
3-shot	100	84.04	82.52	84.04	82.52	100	83.09	80.90	83.09	80.90	100	70.85	48.13	70.85	48.13
5-shot	100	83.87	83.34	83.87	83.34	100	83.61	82.62	83.61	82.62	100	66.35	43.04	66.35	43.04
Definition	100	82.66	81.43	82.66	81.43	100	82.79	81.71	82.79	81.71	100	66.96	52.74	66.96	52.74
Numerical	100	57.18	8.77	57.18	8.77	100	10.68	4.02	10.68	4.02	100	75.30	70.38	75.30	70.38
Similar-3-shot	100	81.57	79.68	81.57	79.68	100	81.49	77.98	81.49	77.98	100	65.27	41.84	65.27	41.84
Fixed-3-shot	100	83.17	82.49	83.17	82.49	100	82.70	81.56	82.70	81.56	100	67.56	48.30	67.56	48.30
Uncertainty	100	83.87	81.84	83.87	81.84	100	**85.08**	83.74	**85.08**	83.74	100	72.97	62.39	72.97	62.39
1-shot w/ Def	100	83.61	81.45	83.61	81.45	100	83.78	82.84	83.78	82.84	100	70.59	51.07	70.59	51.07

Table 7: Experimental results on the **Query Intent** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	89.23	83.29	89.23	83.29	100	91.08	84.99	91.08	84.99	100	92.09	86.14	92.09	86.14
1-shot	100	92.48	**87.34**	92.48	**87.34**	100	92.47	87.20	92.47	87.20	100	92.38	86.97	92.38	86.97
3-shot	100	92.29	85.98	92.29	85.98	100	92.30	86.31	92.30	86.31	100	92.24	85.91	92.24	85.91
5-shot	100	92.41	86.78	92.41	86.78	100	**92.52**	86.95	**92.52**	86.95	100	92.45	87.03	92.45	87.03
Definition	100	91.51	84.62	91.51	84.62	100	91.90	85.26	91.90	85.26	100	92.19	85.62	92.19	85.62
Numerical	100	0	0	0	0	100	0	0	0	0	100	0	0	0	0
Similar-3-shot	100	91.33	84.56	91.33	84.56	100	91.43	85.03	91.43	85.03	100	91.47	85.19	91.47	85.19
Fixed-3-shot	100	92.09	85.58	92.09	85.58	100	92.19	85.65	92.19	85.65	100	92.36	86.17	92.36	86.17
Uncertainty	100	89.82	83.71	89.82	83.71	100	91.15	85.24	91.15	85.24	100	92.24	86.41	92.24	86.41
1-shot w/ Def	100	92.41	78.02	92.41	78.02	100	92.28	77.88	92.28	77.88	100	92.29	86.35	92.29	86.35

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	91.43	85.35	91.43	85.35	100	92.30	86.27	92.30	86.27	0	–	–	–	–
1-shot	100	92.43	87.12	92.43	87.12	100	92.39	87.05	92.39	87.05	0	–	–	–	–
3-shot	100	92.44	86.55	92.44	86.55	100	92.26	86.06	92.26	86.06	0	–	–	–	–
5-shot	100	92.44	86.91	92.44	86.91	100	92.41	86.81	92.41	86.81	0	–	–	–	–
Definition	100	92.10	85.91	92.10	85.91	100	92.23	85.75	92.23	85.75	0	–	–	–	–
Numerical	100	0	0	0	0	100	0	0	0	0	100	**92.52**	86.34	**92.52**	86.34
Similar-3-shot	100	91.58	85.34	91.58	85.34	100	91.60	85.36	91.60	85.36	0	–	–	–	–
Fixed-3-shot	100	92.06	85.61	92.06	85.61	100	92.14	85.58	92.14	85.58	0	–	–	–	–
Uncertainty	100	91.41	85.37	91.41	85.37	100	92.29	86.03	92.29	86.03	0.11	81.82	48.72	0.09	0.05
1-shot w/ Def	100	92.36	86.57	92.36	86.57	100	92.26	86.37	92.26	86.37	0	–	–	–	–

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	91.05	84.99	91.05	84.99	100	92.28	86.33	92.28	86.33	100	92.05	86.14	92.05	86.14
1-shot	100	92.41	87.20	92.41	87.20	100	92.39	87.13	92.39	87.13	100	92.27	86.89	92.27	86.89
3-shot	100	92.24	86.53	92.24	86.53	96.73	92.10	86.28	89.09	83.46	100	92.01	86.00	92.01	86.00
5-shot	100	92.33	86.60	92.33	86.60	99.77	92.44	86.81	92.23	86.61	100	92.17	86.56	92.17	86.56
Definition	100	91.40	84.87	91.40	84.87	100	92.21	85.67	92.21	85.67	100	92.13	85.78	92.13	85.78
Numerical	100	0	0	0	0	100	0	0	0	0	100	56.21	26.25	56.21	26.25
Similar-3-shot	100	92.22	86.40	92.22	86.40	99.91	91.50	85.01	91.42	84.93	100	91.31	84.99	91.31	84.99
Fixed-3-shot	100	91.62	85.63	91.62	85.63	99.87	92.09	85.51	91.97	85.40	100	91.77	85.39	91.77	85.39
Uncertainty	100	91.06	85.05	91.06	85.05	100	92.36	86.51	92.36	86.51	100	92.27	86.10	92.27	86.10
1-shot w/ Def	100	92.01	86.22	92.01	86.22	100	92.09	67.53	92.09	67.53	100	91.92	70.37	91.92	70.37

Table 8: Experimental results on the **Search Correlation** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	63.86	56.02	63.86	56.02	100	65.30	56.16	65.30	56.16	100	64.16	53.56	64.16	53.56
1-shot	100	67.37	59.10	67.37	59.10	100	66.43	58.19	66.43	58.19	100	66.37	57.33	66.37	57.33
3-shot	100	67.17	58.45	67.17	58.45	100	67.20	58.70	67.20	58.70	100	67.23	58.20	67.23	58.20
5-shot	100	66.43	55.75	66.43	55.75	100	66.43	56.06	66.43	56.06	100	66.27	55.51	66.27	55.51
Definition	100	67.17	59.77	67.17	59.77	100	66.73	59.16	66.73	59.16	100	67.10	59.15	67.10	59.15
Numerical	100	0	0	0	0	100	0	0	0	0	100	0	0	0	0
Similar-3-shot	100	64.66	57.27	64.66	57.27	100	64.80	57.57	64.80	57.57	100	65.27	58.01	65.27	58.01
Fixed-3-shot	100	64.36	50.86	64.36	50.86	100	64.06	51.47	64.06	51.47	100	64.26	51.02	64.26	51.02
Uncertainty	100	61.66	48.82	61.66	48.82	100	61.53	47.44	61.53	47.44	100	62.06	47.17	62.06	47.17
1-shot w/ Def	100	67.40	58.51	67.40	58.51	100	66.57	57.35	66.57	57.35	100	65.70	55.47	65.70	55.47

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	64.73	55.34	64.73	55.34	100	66.70	58.68	66.70	58.68	4.14	64.52	44.25	2.67	1.83
1-shot	100	66.23	47.94	66.23	47.94	100	66.57	58.19	66.57	58.19	3	3.33	1.29	0.10	0.04
3-shot	100	67.53	59.23	67.53	59.23	100	66.87	58.52	66.87	58.52	6.14	19.57	23.63	1.20	1.45
5-shot	100	66.73	56.42	66.73	56.42	100	66.47	56.54	66.47	56.54	49.58	56.86	45.83	28.19	22.73
Definition	100	66.17	58.35	66.17	58.35	100	**68.60**	**62.25**	**68.60**	**62.25**	0	–	–	–	–
Numerical	100	0	0	0	0	100	0	0	0	0	100	64.40	47.89	64.40	47.89
Similar-3-shot	100	65.30	58.12	65.30	58.12	100	64.80	57.40	64.80	57.40	0.13	75.00	42.86	0.10	0.06
Fixed-3-shot	100	64.40	51.80	64.40	51.80	100	63.23	48.88	63.23	48.88	37.64	34.22	29.29	12.88	11.02
Uncertainty	100	61.76	47.77	61.76	47.77	100	62.73	48.62	62.73	48.62	0	–	–	–	–
1-shot w/ Def	100	66.40	57.13	66.40	57.13	100	67.40	59.07	67.40	59.07	21.05	64.18	44.11	13.51	9.29

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	59.73	47.63	59.73	47.63	100	65.27	54.44	65.27	54.44	100	41.07	33.73	41.07	33.73
1-shot	100	65.23	53.24	65.23	53.24	100	66.40	57.71	66.40	57.71	100	31.26	23.15	31.26	23.15
3-shot	100	66.67	55.98	66.67	55.98	100	67.00	58.47	67.00	58.47	100	28.43	20.29	28.43	20.29
5-shot	100	65.63	52.74	65.63	52.74	100	65.80	54.10	65.80	54.10	100	25.26	16.78	25.26	16.78
Definition	100	65.47	54.38	65.47	54.38	100	68.20	60.99	68.20	60.99	100	29.70	21.78	29.70	21.78
Numerical	100	0	0	0	0	100	0	0	0	0	100	32.73	23.72	32.73	23.72
Similar-3-shot	100	67.63	60.01	67.63	60.01	100	65.97	57.31	65.97	57.31	100	25.19	17.31	25.19	17.31
Fixed-3-shot	100	64.76	50.08	64.76	50.08	100	63.50	48.29	63.50	48.29	100	34.70	24.71	34.70	24.71
Uncertainty	100	60.89	45.80	60.89	45.80	100	65.57	53.18	65.57	53.18	100	38.64	28.08	38.64	28.08
1-shot w/ Def	100	65.53	53.36	65.53	53.36	100	67.03	57.99	67.03	57.99	100	29.66	21.41	29.66	21.41

Table 9: Experimental results on the **Query Taxonomy** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	51.53	42.6	51.53	42.6	100	51.47	42.92	51.47	42.92	100	51.68	42.99	51.68	42.99
1-shot	100	52.9	**44.91**	52.9	**44.91**	100	52.43	44.56	52.43	44.56	100	52.08	44.34	52.08	44.34
3-shot	100	51.25	43.56	51.25	43.56	100	52.1	44.07	52.1	44.07	100	51.61	43.72	51.61	43.72
5-shot	100	51.31	43.44	51.31	43.44	100	51.11	43.25	51.11	43.25	100	50.97	43.38	50.97	43.38
Numerical	100	0	0	0	0	100	0	0	0	0	100	0	0	0	0
Similar-3-shot	100	48.56	40.75	48.56	40.75	100	49.46	41.81	49.46	41.81	100	48.25	41.31	48.25	41.31
Fixed-3-shot	100	50.74	42.49	50.74	42.49	100	49.62	42.32	49.62	42.32	100	50.37	42.39	50.37	42.39
Uncertainty	100	48.13	37.42	48.13	37.42	100	49.16	38.13	49.16	38.13	100	48.77	37.67	48.77	37.67

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	51.3	42.77	51.3	42.77	–	–	–	–	–	100	0	0	0	0
1-shot	100	51.83	44.3	51.83	44.3	–	–	–	–	–	100	0	0	0	0
3-shot	100	51.54	43.44	51.54	43.44	–	–	–	–	–	56.13	0	0	0	0
5-shot	100	51.82	43.66	51.82	43.66	–	–	–	–	–	93.11	0	0	0	0
Numerical	100	0	0	0	0	–	–	–	–	–	100	51.26	42.87	51.26	42.87
Similar-3-shot	100	47.99	41.23	47.99	41.23	–	–	–	–	–	84.55	0	0	0	0
Fixed-3-shot	100	50.48	42.44	50.48	42.44	–	–	–	–	–	55.38	0	0	0	0
Uncertainty	100	47.66	37.52	47.66	37.52	–	–	–	–	–	100	0	0	0	0

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	53.25	43.13	53.25	43.13	100	51.43	43.1	51.43	43.1	–	–	–	–	–
1-shot	100	53.09	44.02	53.09	44.02	100	52.09	44.59	52.09	44.59	–	–	–	–	–
3-shot	100	53.38	43.84	53.38	43.84	13.38	56.65	51.14	7.58	6.84	–	–	–	–	–
5-shot	100	53.95	44.52	53.95	44.52	52.18	58.63	51.97	30.6	27.12	–	–	–	–	–
Numerical	100	0	0	0	0	100	0	0	0	0	–	–	–	–	–
Similar-3-shot	100	52.99	44.11	52.99	44.11	43.95	52.54	45.77	23.09	20.11	–	–	–	–	–
Fixed-3-shot	100	**54.03**	43.6	**54.03**	43.6	22.42	53.72	47.8	12.04	10.72	–	–	–	–	–
Uncertainty	100	50.21	38.35	50.21	38.35	100	48.74	38.1	48.74	38.1	–	–	–	–	–

Table 10: Experimental results on the **EC** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	83.30	78.97	83.30	78.97	100	92.60	88.73	92.60	88.73	100	91.95	88.53	91.95	88.53
1-shot	100	92.95	88.97	92.95	88.97	100	92.80	88.76	92.80	88.76	100	92.75	88.62	92.75	88.62
3-shot	100	93.40	89.10	93.40	89.10	100	93.25	88.71	93.25	88.71	100	93.20	88.46	93.20	88.46
5-shot	100	93.90	89.05	93.90	89.05	100	93.75	88.99	93.75	88.99	100	93.65	88.94	93.65	88.94
Definition	100	78.85	73.68	78.85	73.68	100	92.35	87.19	92.35	87.19	100	92.00	87.04	92.00	87.04
Numerical	100	38.80	1.67	38.80	1.67	100	56.55	2.58	56.55	2.58	100	43.05	1.71	43.05	1.71
Similar-3-shot	100	93.20	88.43	93.20	88.43	100	93.25	88.55	93.25	88.55	100	93.60	89.08	93.60	89.08
Fixed-3-shot	100	93.60	89.56	93.60	89.56	100	93.45	89.04	93.45	89.04	100	93.80	89.92	93.80	89.92
Uncertainty	100	77.25	73.31	77.25	73.31	100	91.45	88.06	91.45	88.06	100	92.50	89.07	92.50	89.07
1-shot w/ Def	100	93.60	89.43	93.60	89.43	100	93.80	89.70	93.80	89.70	100	93.35	89.65	93.35	89.65

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	93.50	89.92	93.50	89.92	100	93.75	**90.31**	93.75	**90.31**	0	–	–	–	–
1-shot	100	92.85	88.62	92.85	88.62	63.60	94.50	90.98	60.10	57.86	0	–	–	–	–
3-shot	100	93.20	88.63	93.20	88.63	19.55	94.88	95.94	18.55	18.76	0	–	–	–	–
5-shot	100	**94.15**	89.83	**94.15**	89.83	99.85	93.84	89.48	93.70	89.34	0	–	–	–	–
Definition	100	92.85	88.05	92.85	88.05	100	93.15	88.04	93.15	88.04	0	–	–	–	–
Numerical	100	63.50	3.12	63.50	3.12	100	62.45	5.70	62.45	5.70	100	93.65	89.93	93.65	89.93
Similar-3-shot	100	93.55	88.71	93.55	88.71	3.05	78.69	89.79	2.40	2.74	6.25	87.20	73.43	5.45	4.59
Fixed-3-shot	100	93.35	89.06	93.35	89.06	85.25	93.61	89.11	79.80	75.97	0	–	–	–	–
Uncertainty	100	92.75	89.27	92.75	89.27	87.30	92.50	89.50	80.75	78.14	0	–	–	–	–
1-shot w/ Def	100	93.35	88.88	93.35	88.88	100	93.80	89.95	93.80	89.95	0	–	–	–	–

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	92.35	88.73	92.35	88.73	100	93.70	90.17	93.70	90.17	100	93.65	89.96	93.65	89.96
1-shot	100	93.00	88.93	93.00	88.93	100	93.15	89.45	93.15	89.45	100	91.05	87.30	91.05	87.30
3-shot	100	93.05	88.27	93.05	88.27	99.95	93.50	89.17	93.45	89.13	100	93.20	89.39	93.20	89.39
5-shot	100	93.60	89.17	93.60	89.17	100	93.70	89.06	93.70	89.06	100	93.55	89.19	93.55	89.19
Definition	100	92.10	87.12	92.10	87.12	100	93.30	88.31	93.30	88.31	100	92.90	88.30	92.90	88.30
Numerical	100	63.20	2.65	63.20	2.65	100	55.35	2.88	55.35	2.88	100	92.45	88.74	92.45	88.74
Similar-3-shot	100	93.90	89.44	93.90	89.44	39.65	94.70	92.81	37.55	36.80	100	91.95	87.81	91.95	87.81
Fixed-3-shot	100	93.25	89.33	93.25	89.33	100	93.20	89.14	93.20	89.14	100	92.90	88.92	92.90	88.92
Uncertainty	100	91.90	88.37	91.90	88.37	100	93.55	90.01	93.55	90.01	100	93.05	89.49	93.05	89.49
1-shot w/ Def	100	93.00	88.09	93.00	88.09	100	93.80	89.81	93.80	89.81	100	92.50	88.93	92.50	88.93

Table 11: Experimental results on the **IFLYTEK** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	60.95	44.95	60.95	44.95	100	60.95	45.33	60.95	45.33	100	61.06	45.32	61.06	45.32
1-shot	100	63.22	46.75	63.22	46.75	100	63.33	47.15	63.33	47.15	100	63.22	47.15	63.22	47.15
3-shot	100	62.91	**48.51**	62.91	**48.51**	100	62.33	47.33	62.33	47.33	100	62.91	47.22	62.91	47.22
5-shot	100	61.91	44.60	61.91	44.60	100	61.75	44.05	61.75	44.05	100	61.75	43.26	61.75	43.26
Definition	100	63.41	44.86	63.41	44.86	100	63.64	44.41	63.64	44.41	100	63.26	44.49	63.26	44.49
Numerical	100	53.67	22.11	53.67	22.11	100	54.83	24.27	54.83	24.27	100	54.41	22.55	54.41	22.55
Similar-3-shot	100	62.10	46.43	62.10	46.43	100	62.68	47.19	62.68	47.19	100	62.83	47.69	62.83	47.69
Fixed-3-shot	100	63.49	45.88	63.49	45.88	100	63.37	45.22	63.37	45.22	100	63.52	45.26	63.52	45.26
Uncertainty	100	63.33	46.91	63.33	46.91	100	63.26	46.73	63.26	46.73	100	63.06	46.45	63.06	46.45
1-shot w/ Def	100	63.06	46.47	63.06	46.47	100	63.06	46.62	63.06	46.62	100	63.37	47.47	63.37	47.47

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	61.02	45.27	61.02	45.27	100	61.56	46.93	61.56	46.93	0	–	–	–	–
1-shot	100	63.18	46.45	63.18	46.45	99.65	63.17	47.91	62.95	47.75	0.15	75.00	42.86	0.12	0.07
3-shot	100	62.41	47.34	62.41	47.34	98.46	63.03	48.74	62.06	47.99	2.12	41.82	15.85	0.88	0.34
5-shot	100	61.99	44.01	61.99	44.01	100	62.52	44.93	62.52	44.93	0.15	50.00	22.22	0.08	0.03
Definition	100	63.45	44.35	63.45	44.35	100	62.83	44.90	62.83	44.90	0	–	–	–	–
Numerical	100	53.79	22.34	53.79	22.34	100	50.29	27.23	50.29	27.23	100	62.29	46.29	62.29	46.29
Similar-3-shot	100	62.64	47.29	62.64	47.29	100	62.37	47.72	62.37	47.72	0.27	42.86	12.00	0.12	0.03
Fixed-3-shot	100	63.29	45.08	63.29	45.08	100	63.10	46.43	63.10	46.43	0.08	0.00	0.00	0.00	0.00
Uncertainty	100	63.45	46.42	63.45	46.42	100	**63.76**	47.85	**63.76**	47.85	0	–	–	–	–
1-shot w/ Def	100	63.26	46.46	63.26	46.46	100	62.52	46.36	62.52	46.36	0	–	–	–	–

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	62.83	46.69	62.83	46.69	100	61.18	45.42	61.18	45.42	100	38.21	31.67	38.21	31.67
1-shot	100	62.06	45.15	62.06	45.15	99.35	63.44	46.39	63.02	46.09	100	39.28	30.76	39.28	30.76
3-shot	100	62.29	46.16	62.29	46.16	97.81	62.90	47.41	61.52	46.37	100	42.17	33.51	42.17	33.51
5-shot	100	61.83	43.97	61.83	43.97	100	61.87	43.96	61.87	43.96	100	41.98	31.24	41.98	31.24
Definition	100	62.60	43.88	62.60	43.88	100	63.45	45.41	63.45	45.41	100	28.43	23.69	28.43	23.69
Numerical	100	58.64	29.57	58.64	29.57	100	50.98	20.52	50.98	20.52	100	46.44	38.71	46.44	38.71
Similar-3-shot	100	62.45	46.37	62.45	46.37	100	62.49	47.45	62.49	47.45	100	38.51	29.83	38.51	29.83
Fixed-3-shot	100	63.14	44.78	63.14	44.78	100	62.72	43.99	62.72	43.99	100	40.40	30.23	40.40	30.23
Uncertainty	100	63.45	47.15	63.45	47.15	100	63.64	46.61	63.64	46.61	100	46.25	35.77	46.25	35.77
1-shot w/ Def	100	62.75	46.72	62.75	46.72	100	63.18	46.73	63.18	46.73	100	19.51	15.56	19.51	15.56

Table 12: Experimental results on the **TNEWS** dataset (Accuracy & macro-F1 Score, %). Blue highlights the best inference strategy for each training method, while **bold** denotes the overall best performance across all settings.

Method	1-shot					3-shot					fix_3_shot				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	61.31	58.96	61.31	58.96	100	61.31	58.70	61.31	58.70	100	61.33	58.75	61.33	58.75
1-shot	100	61.23	59.98	61.23	59.98	100	61.45	60.08	61.45	60.08	100	61.41	60.15	61.41	60.15
3-shot	100	61.07	59.56	61.07	59.56	100	61.21	60.09	61.21	60.09	100	61.30	60.25	61.30	60.25
5-shot	100	61.73	57.66	61.73	57.66	100	61.48	57.40	61.48	57.40	100	61.48	57.72	61.48	57.72
Definition	100	60.48	59.00	60.48	59.00	100	60.84	59.56	60.84	59.56	100	60.56	59.27	60.56	59.27
Numerical	100	54.12	2.65	54.12	2.65	100	55.07	3.89	55.07	3.89	100	55.86	5.05	55.86	5.05
Similar-3-shot	100	60.60	59.02	60.60	59.02	100	60.59	59.07	60.59	59.07	100	60.47	59.20	60.47	59.20
Fixed-3-shot	100	61.11	57.92	61.11	57.92	100	61.07	58.45	61.07	58.45	100	60.91	58.66	60.91	58.66
Uncertainty	100	60.54	56.71	60.54	56.71	100	60.77	56.86	60.77	56.86	100	60.84	56.97	60.84	56.97
1-shot w/ Def	100	61.09	59.92	61.09	59.92	100	61.32	60.09	61.32	60.09	100	61.31	60.02	61.31	60.02

Method	5_shot					category_definition					numerical				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	61.52	59.12	61.52	59.12	100	61.52	58.98	61.52	58.98	0	–	–	–	–
1-shot	100	61.35	60.08	61.35	60.08	100	61.29	59.92	61.29	59.92	0	–	–	–	–
3-shot	100	61.39	60.22	61.39	60.22	100	61.16	59.98	61.16	59.98	0	–	–	–	–
5-shot	100	61.82	58.02	61.82	58.02	100	61.76	57.61	61.76	57.61	0	–	–	–	–
Definition	100	60.89	59.73	60.89	59.73	100	60.63	59.38	60.63	59.38	0	–	–	–	–
Numerical	100	55.10	4.64	55.10	4.64	100	31.56	0.43	31.56	0.43	100	61.24	57.09	61.24	57.09
Similar-3-shot	100	60.59	59.17	60.59	59.17	100	60.67	58.89	60.67	58.89	0	–	–	–	–
Fixed-3-shot	100	61.01	58.40	61.01	58.40	100	60.80	57.19	60.80	57.19	0	–	–	–	–
Uncertainty	100	60.80	56.96	60.80	56.96	100	60.73	56.85	60.73	56.85	0	–	–	–	–
1-shot w/ Def	100	61.35	60.19	61.35	60.19	100	61.22	59.85	61.22	59.85	0	–	–	–	–

Method	similar_3_shot					zero_shot					ppl				
	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1	fmt-suc ratio	fmt-suc acc	fmt-suc macro-f1	overall acc	overall macro-f1
Zero-shot	100	**63.30**	**61.31**	**63.30**	**61.31**	100	60.83	59.39	60.83	59.39	100	45.68	35.99	45.68	35.99
1-shot	100	61.98	60.49	61.98	60.49	100	61.10	59.75	61.10	59.75	100	49.46	41.59	49.46	41.59
3-shot	100	62.06	60.53	62.06	60.53	100	61.15	60.07	61.15	60.07	100	43.84	36.87	43.84	36.87
5-shot	100	62.54	58.41	62.54	58.41	100	61.94	57.85	61.94	57.85	100	47.88	39.06	47.88	39.06
Definition	100	61.37	59.26	61.37	59.26	100	60.63	59.66	60.63	59.66	100	46.21	38.56	46.21	38.56
Numerical	100	59.64	12.99	59.64	12.99	100	39.16	0.54	39.16	0.54	100	53.68	50.42	53.68	50.42
Similar-3-shot	100	**63.30**	**61.31**	**63.30**	**61.31**	100	60.83	59.39	60.83	59.39	100	45.68	35.99	45.68	35.99
Fixed-3-shot	100	62.25	58.94	62.25	58.94	100	60.96	58.37	60.96	58.37	100	49.46	40.44	49.46	40.44
Uncertainty	100	61.97	58.15	61.97	58.15	100	61.12	57.32	61.12	57.32	100	49.35	41.01	49.35	41.01
1-shot w/ Def	100	62.20	60.65	62.20	60.65	100	61.30	59.90	61.30	59.90	100	49.89	41.75	49.89	41.75

D Packing Results

A common optimization technique in the pre-training stage of LLMs is **packing**, where multiple training samples are concatenated into a single sequence to improve computational efficiency. When applied to SFT for classification tasks, packing introduces two effects: (i) it increases the effective batch size and context length, and (ii) it allows samples within a packed sequence to attend to preceding samples—referred to as contaminated attention. We hypothesize that this second effect may mimic the behavior of ICL training.

To test this hypothesis, we conducted experiments on seven datasets under three conditions: (i) no packing, (ii) packing, and (iii) packing with attention mask to prevent cross-sample contamination.

From the results in Table 13, on QI, SC, EC, IT datasets, we observe that **neat packing yields higher zero-shot accuracy, while standard packing achieves better few-shot performance**—providing empirical support for our hypothesis that cross-sample attention mimics in-context learning. Moreover, the no-packing setting yields the best performance on five of the seven datasets, specifically the QI, SC, EC, EIC, and TN.

Table 13: Performance of packing strategies across all datasets. Accuracy and macro-F1 (%) are reported for 1-shot, 3-shot, 5-shot, and zero-shot settings. Cells shaded in green denote cases where standard packing outperforms neat packing, whereas yellow shading indicates the opposite. **Bold** numbers mark the best results in each column.

Dataset	Method	1-shot		3-shot		5-shot		zero-shot	
		Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1
QI	No Packing	89.23	83.29	91.08	84.99	91.43	85.35	**92.28**	**86.33**
	Packing	90.93	83.69	91.30	84.65	91.29	84.85	91.20	84.28
	Neat Packing	91.00	79.38	91.29	80.01	91.15	83.99	91.58	84.94
SC	No Packing	63.86	56.02	65.30	56.16	64.73	55.34	**67.43**	**58.64**
	Packing	64.90	51.97	64.76	51.37	65.10	51.96	65.07	51.47
	Neat Packing	61.93	51.20	62.16	49.36	61.63	48.12	65.27	54.44
QT	No Packing	51.53	42.60	51.47	42.92	51.30	42.77	51.43	43.10
	Packing	46.99	41.50	47.94	42.21	49.38	**43.38**	50.10	41.79
	Neat Packing	51.19	42.24	**51.82**	42.71	50.24	42.49	51.32	42.76
EC	No Packing	83.30	78.97	92.60	88.73	93.50	89.92	**93.70**	**90.17**
	Packing	91.55	87.94	91.20	86.80	91.60	87.03	91.95	87.05
	Neat Packing	76.75	70.92	89.55	84.73	90.85	85.91	92.85	88.70
EIC	No Packing	83.43	82.40	83.39	82.67	**84.04**	**82.93**	82.74	81.73
	Packing	81.66	79.20	81.96	79.62	82.74	81.05	81.62	74.95
	Neat Packing	82.61	80.80	83.52	82.66	83.22	82.09	83.39	82.77
IT	No Packing	61.31	44.95	60.95	45.33	61.02	45.27	61.18	45.42
	Packing	63.41	47.14	63.83	48.15	**64.06**	47.56	63.95	48.41
	Neat Packing	63.49	47.50	62.75	46.91	63.18	47.24	63.99	**48.99**
TN	No Packing	61.31	58.96	61.31	58.70	61.52	59.12	**61.71**	**59.51**
	Packing	60.74	56.62	60.70	56.46	60.58	56.36	60.62	56.66
	Neat Packing	60.86	58.33	60.65	57.41	60.84	57.42	61.40	58.58