RD-MCSA: A Multi-Class Sentiment Analysis Approach Integrating In-Context Classification Rationales and Demonstrations

Anonymous ACL submission

Abstract

Multi-class sentiment analysis (MCSA) poses significant challenges due to its multiple categories and the subtle semantic distinctions between adjacent classes, necessitating substantial amounts of high-quality annotated data, which is often scarce. This paper introduces **RD-MCSA** (**R**ationales and **D**emonstrations based **M**ulti-**C**lass **S**entiment **A**nalysis), an approach that enhances classification performance with limited labeled data. RD-MCSA leverages In-Context Learning (ICL) by integrating classification rationales and demonstration examples, enabling Large Language Models (LLMs) to make more accurate predictions. In RD-MCSA, a representative set of annotated samples is constructed using a balanced Coreset algorithm to guide LLMs in generating classification rationales grounded in linguistic and semantic features. These rationales are then integrated with demonstration examples, selected via a Multi-Kernel Gaussian Process (MK-GP)-based similarity evaluation method, to enhance ICL for MCSA. Experiments on five diverse datasets demonstrate that RD-MCSA outperforms both supervised learning methods and conventional ICL approaches across key evaluation metrics.

1 Introduction

Multi-Class Sentiment Analysis (MCSA) extends beyond basic sentiment polarity classification (e.g., positive or negative) by distinguishing varying levels of emotional intensity (e.g., differentiating between "very positive" and "generally positive"). By capturing finer sentiment distinctions, MCSA enables deeper insights into sentiment expression, making it essential for applications requiring fine-grained sentiment analysis (Wang et al., 2023). For example, in opinion dynamics research, a prerequisite step is categorizing users' natural language expressions into five or more sentiment or opinion categories. (Chuang et al., 2024)

However, the complexity of MCSA arises from the subtle differences between adjacent sentiment intensities, which are often challenging to discern accurately (Mamta and Ekbal, 2023). Moreover, sentiment categorization criteria can vary significantly across applications (Rosenthal et al., 2019). Effectively tackling a new MCSA task typically demands a substantial amount of high-quality labeled data tailored to the task's specific requirements.

Large Language Models (LLMs) have shown strong performance in sentiment analysis, making them a promising approach for MCSA. However, while LLMs perform well in straightforward sentiment classification, they often struggle with nuanced distinctions between sentiment categories (Zhang et al., 2024). In-Context Learning, which enhances LLM performance by providing demonstration examples, has achieved state-of-the-art results in various NLP tasks. Yet, existing research has largely overlooked its effective application to classification tasks with a large number of sentiment categories (Randl et al., 2024). Our experimental analysis further reveals that traditional ICL approaches remain insufficient for MCSA.

In this paper, we propose RD-MCSA, a novel approach to improve ICL performance for MCSA. RD-MCSA leverages explicit category division rationales, generated through LLM-driven reasoning of semantic and linguistic features from a representative set of labeled MCSA samples. This integration enriches the decision-making process in sentiment analysis. Additionally, we introduce a text similarity evaluation method using a multi-kernel Gaussian process to optimize the selection of high-quality demonstration examples for ICL.

In summary, this paper makes the following main contributions:

1. **Integration of Classification Rationales and Demonstrations for ICL**: This approach enhances the performance of ICL for MCSA

by incorporating classification rationales and demonstration examples. These rationales, grounded in linguistic and semantic features, guide LLMs in achieving more accurate and nuanced sentiment classification.

2. **Classification Rationale Generation via Balanced Coreset Selection**: A balanced Coreset algorithm is developed to construct a standard reference set to generate classification rationales, ensuring comprehensive representation and class balance.

3. **MK-GP-based Demonstration Selection**: A text similarity evaluation method leveraging a Multi-Kernel Gaussian Process (MK-GP) is introduced to enhance the selection of high-quality demonstration examples for ICL.

A series of comprehensive experiments on five diverse and representative datasets validate the effectiveness of RD-MCSA, highlighting its advantages and identifying key challenges in MCSA tasks.

2 Related Works

2.1 Multi-class Sentiment Analysis

Multi-class sentiment analysis (MCSA), also known as fine-grained or graded sentiment analysis (Sharma et al., 2024), extends traditional sentiment classification by categorizing sentiments into multiple distinct classes. It refines sentiment intensity beyond polarity classification (e.g., "positive"/"negative") by introducing subcategories like "very positive" and "slightly positive" or rating scales (e.g., 1–5) (AlQahtani, 2021). This provides a more nuanced understanding of sentiment in text.

Traditional MCSA models rely on supervised machine learning (Wang et al., 2023) and are typically applied to texts such as tweets, movie reviews, and product reviews, with sentiment analysis often focused on specific targets or aspects. Common MCSA datasets include SemEval-2017 Task 4 (Rosenthal et al., 2019), SST-5 (Socher et al., 2013), and Amazon Reviews (AlQahtani, 2021).

Another research direction treats sentiment intensity assessment as a regression problem, where sentiment is predicted on a continuous scale. Notable tasks and datasets include SemEval-2017 Task 5 (Cortis et al., 2017), FiQA 2018 (de França Costa and da Silva, 2018), and recent dimABSA tasks at SIGHAN-2024 (Lee et al., 2024).

Despite advances, MCSA still faces challenges such as accuracy limitations and the high cost of large-scale annotation, particularly as sentiment granularity increases (Krosuri and Aravapalli, 2023). Fine-grained sentiment analysis for specific entities often requires distinct annotated datasets, making large-scale implementation impractical.

To address these challenges, we aim to improve MCSA performance with limited labeled data while ensuring a versatile approach applicable to various MCSA scenarios.

2.2 Text Analysis Using LLMs

Large-scale language models outperform smaller models in many NLP tasks, especially when annotation resources are limited (Zhang et al., 2024), making them a promising solution for MCSA.

Recent research on LLMs for text analysis has focused on in-context learning, where carefully selected demonstration examples guide the model's predictions. Common strategies for example selection include similarity-based selection (Liu et al., 2022), diversity-based selection (Levy et al., 2023), LLM feedback (Shi et al., 2022), information-theoretic criteria (Wu et al., 2023), task-level selection (Li and Qiu, 2023), active learning (Zhang et al., 2022a), and contrastive learning (Chen et al., 2024). For MCSA, a recent study (Chuang et al., 2024) applies similarity-based demonstration selection within ICL to analyze opinion dynamics.

Despite their potential, LLMs still face challenges in many NLP tasks. While effective for simpler tasks, they struggle with nuanced sentiment analysis (Zhang et al., 2024). Additionally, few-shot ICL requires further research on optimal prompt design (Liu et al., 2022). To our knowledge, no prior work has explored few-shot prompting for multi-class prediction with a large number of classes (Randl et al., 2024). Long prompts can overload LLMs (Liu et al., 2024), and context window limitations may restrict the effective representation of all classes. Addressing how to **efficiently provide classification information to LLMs** is a key focus of this paper.

3 The Methodology of RD-MCSA

The RD-MCSA framework, as illustrated in Fig. 1, consists of the following key steps: given an annotated dataset $\mathcal{D}$, 1) a balanced-coreset $\mathcal{B}$ is constructed to derive the classification rationale $\mathbb{R}$ (Section 3.1), 2) a multi-kernel Gaussian process $\mathbb{G}$ is

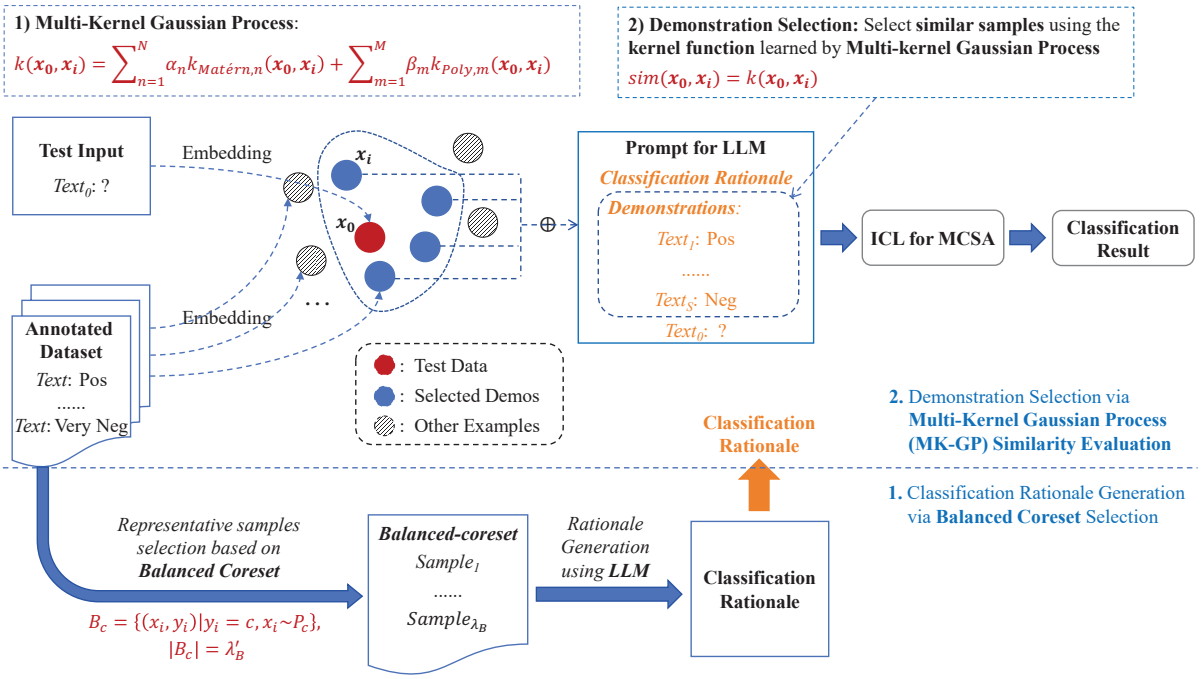

Figure 1: The workflow of RD-MCSA: The lower half of the figure (below the long dashed line) corresponds to Section 3.1, while the upper half (above the long dashed line) corresponds to Section 3.2. The training of the MK-GP (described in Subsection 3.2.2) is omitted in the figure.

trained (Subsection 3.2.2), 3) for MCSA on a new text, ICL is carried out by using a prompt that encompasses the classification rationale $\mathbb{R}$ and a set of demonstration examples selected from $\mathcal{D}$ by $\mathbb{G}$ (Subsection 3.2.3).

3.1 Classification Rationale Generation via Balanced Coreset Selection

To ensure that the generated classification rationale **adequately represents and balances the characteristics of each class**, a balanced-coreset $\mathcal{B}$ is constructed from the annotated dataset $\mathcal{D} = \{(x_i, y_i) \mid 1 \leq i \leq |\mathcal{D}|\}$ using a balanced Coreset algorithm. This algorithm selects a subset of representative samples while preserving class distribution to avoid over-representation of dominant classes.

3.1.1 The Balanced Coreset Algorithm

The Balanced Coreset selection process consists of the following steps:

1) Determining Class-Specific Sample Limits

The coreset size $\lambda_{\mathcal{B}}$ (a hyperparameter) serves as an upper bound on the total selected samples in $\mathcal{B}$. To maintain class balance, the **per-class selection limit** is:

$$\lambda'_{\mathcal{B}} = \left\lceil \frac{\lambda_{\mathcal{B}}}{u} \right\rceil$$

where u is the number of unique classes in $\mathcal{D}$. This prevents any class from being overrepresented.

2) Computing the Sampling Probability

To select the most informative samples, selection probability is assigned based on an importance weight function $w(x_i, y_i)$, which gives higher weight to samples farther from the class centroid:

$$\mu_c = \frac{1}{|\mathcal{D}_c|} \sum_{j:y_j=c} \phi(x_j) \quad (1)$$

where $\phi(x_j)$ represents the embedding of x_j, and $|\mathcal{D}_c|$ is the number of samples in class c. Probabilities are then normalized within each class:

$$P_c(x_i) = \frac{w(x_i, y_i)}{\sum_{j:y_j=c} w(x_j, y_j)} \quad (2)$$

$P_c(x_i)$ represents the normalized selection probability of sample x_i within class c.

3) Stratified Weighted Random Sampling

Stratified weighted random sampling is applied to select representative samples from each class. The selection process follows these rules:

- If $|\mathcal{D}_c| \leq \lambda'_{\mathcal{B}}$, all samples in class c are directly included in $\mathcal{B}$.
- If $|\mathcal{D}_c| > \lambda'_{\mathcal{B}}$, a subset of $\lambda'_{\mathcal{B}}$ samples is selected via weighted random sampling using the computed probabilities $P_c(x_i)$ as follows:

$$\mathcal{B}_c = \{(x_i, y_i) \mid y_i = c, x_i \sim P_c\}, |\mathcal{B}_c| = \lambda'_{\mathcal{B}}$$

where $\mathcal{B}_c$ is the subset of selected samples for class c.

Finally, the balanced-coreset $\mathcal{B}$ is obtained by aggregating the selected subsets from all u classes:

$$\mathcal{B} = \bigcup_{c=1}^{u} \mathcal{B}_c, \quad |\mathcal{B}| \leq \lambda_{\mathcal{B}}$$

This approach ensures that $\mathcal{B}$ remains representative, diverse, and class-balanced.

3.1.2 Classification Rationale Generation

The balanced-coreset $\mathcal{B}$ serves as a reference set for generating classification rationale using LLMs. An LLM is prompted with examples from $\mathcal{B}$ to extract and articulate **class-specific sentiment characteristics**. The prompt guides the LLM to reason about class distinctions based on the following aspects:

1) Linguistic expressions
2) Semantic distinctions
3) The characteristics of the target or aspect of sentiment expression.

The LLM is instructed to derive clear, distinguishing classification criteria and identify **specific words, phrases, and expressions from the example set** to enrich the rationale.

The resulting classification rationale $\mathbb{R}$ is then incorporated into the ICL process to enhance classification accuracy. An example prompt for generating rationale is provided in Appendix A.

3.2 Demonstration Selection via Multi-Kernel Gaussian Process Similarity Evaluation

RD-MCSA leverages a multi-kernel Gaussian process to evaluate text similarity for selecting ICL demonstrations. This method benefits from Multiple Kernel Learning's ability to model and adapt to complex data distributions (Ghasempour and Martínez-Ramón, 2023). Initially, the multi-kernel Gaussian process is trained on the annotated dataset $\mathcal{D}$ to identify the unique characteristics of each category. Subsequently, the trained MK-GP's kernel functions are employed to assess text similarity, enabling the effective selection of demonstrations.

3.2.1 Gaussian Process (GP)

Gaussian Process (GP) (Liu et al., 2021) is a Bayesian non-parametric method that can be applied to model categorical data with C categories ($\mathcal{Y} = \{1, \ldots, C\}$) by introducing a set of latent functions $\{f_c(\boldsymbol{x})\}_{c=1}^{C}$, one for each class. Here, $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^W$ represents the input space. For text sentiment classification, $\mathcal{X}$ corresponds to the embedding space of input sentences.

Each latent function is modeled as an independent Gaussian Process (GP) (Wang, 2023):

$$f_c(\boldsymbol{x}) \sim \mathcal{GP}(e_c(\boldsymbol{x}), k_c(\boldsymbol{x}, \boldsymbol{x'})), \tag{3}$$

where $e_c(\boldsymbol{x})$ denotes the mean function, and $k_c(\boldsymbol{x}, \boldsymbol{x'})$ represents the covariance function (also referred to as the **kernel**) for the c-th class. In this work, the mean function is modeled as a learnable constant without additional constraints, and the covariance function is designed as a multi-kernel function, as detailed in Section 3.2.2.

Although the kernel function parameters and the mean function are inherently independent across categories, this study adopts a shared covariance function $k(\boldsymbol{x_i}, \boldsymbol{x_j})$ and mean function for all categories $c \in \mathcal{Y}$ (Bonilla et al., 2007). This design choice not only reduces computational complexity but also leverages the structural similarities often observed among different categories within the same dataset.

3.2.2 Multi-Kernel Gaussian Process

Multi-Kernel Gaussian Process (MK-GP) extends standard GP by integrating Multiple Kernel Learning (MKL). This approach enhances the model's capability to represent and adapt to complex data distributions through a flexible combination of kernel functions (Ghasempour and Martínez-Ramón, 2023). In this work, we employ a weighted combination of the Matérn kernel (Borovitskiy et al., 2021) and the polynomial kernel (Song et al., 2021) to effectively capture both stationary and non-stationary patterns in the data (Lawler, 2018). The combined kernel function is defined as follows:

$$\begin{aligned} k(\boldsymbol{x_i}, \boldsymbol{x_j}) = &\sum_{n=1}^{N} \alpha_n k_{\text{Matérn},n}(\boldsymbol{x_i}, \boldsymbol{x_j}) + \\ &\sum_{m=1}^{M} \beta_m k_{Poly,m}(\boldsymbol{x_i}, \boldsymbol{x_j}), \end{aligned} \tag{4}$$

where $k_{Matérn,n}(\boldsymbol{x_i}, \boldsymbol{x_j})$ represents the n-th Matérn kernel, and $k_{Poly,m}(\boldsymbol{x_i}, \boldsymbol{x_j})$ denotes the m-th polynomial kernel. The coefficients α_n and β_m are learnable weights constrained to be non-negative ($\alpha_n, \beta_m \geq 0$). The Matérn kernel is de-

fined as:

$$k_{\text{Matérn},n}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \frac{2^{1-\nu_n}}{\Gamma(\nu_n)} \left(\sqrt{2\nu_n} \frac{\|\boldsymbol{x_i} - \boldsymbol{x_j}\|}{\ell_n} \right)^{\nu_n} B_{\nu_n} \left(\sqrt{2\nu_n} \frac{\|\boldsymbol{x_i} - \boldsymbol{x_j}\|}{\ell_n} \right), \quad (5)$$

where ν_n controls the smoothness of the kernel, ℓ_n is the length scale, and both are learnable parameters. Γ is the gamma function, and B_{ν_n} is the modified Bessel function of the second kind, with their specific forms detailed in the Appendix B.

The polynomial kernel is expressed as:

$$k_{Poly,m}(\boldsymbol{x_i}, \boldsymbol{x_j}) = (\gamma_m \langle \boldsymbol{x_i}, \boldsymbol{x_j} \rangle + c_m)^{d_m}, \quad (6)$$

where γ_m is a scaling factor, c_m is an offset (both learnable parameters), and d_m is the degree of the polynomial, treated as a hyper-parameter. Here, $\langle \boldsymbol{x_i}, \boldsymbol{x_j} \rangle$ denotes the dot product of $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$.

Given the annotated dataset $\mathcal{D} = \{\boldsymbol{X}, \boldsymbol{y}\}$, where $\boldsymbol{X}$ represents the input data and $\boldsymbol{y}$ denotes the corresponding labels, an MK-GP model $\mathbb{G}$ is trained. The training process for GP involves optimizing the kernel parameters by minimizing the negative log-marginal likelihood (Artemev et al., 2021). To optimize the kernel parameter vector $\boldsymbol{\theta}$, the gradient of the negative log-marginal likelihood $\mathcal{L}$ with respect to each parameter θ_p is computed as:

$$\frac{\partial \mathcal{L}}{\partial \theta_p} = -\frac{1}{2} \boldsymbol{y}^T \boldsymbol{K}^{-1} \frac{\partial \boldsymbol{K}}{\partial \theta_p} \boldsymbol{K}^{-1} \boldsymbol{y} + \frac{1}{2} \operatorname{Tr} \left(\boldsymbol{K}^{-1} \frac{\partial \boldsymbol{K}}{\partial \theta_p} \right), \quad (7)$$

where $\boldsymbol{K}$ is the covariance matrix with entries $[\boldsymbol{K}]_{ij} = k(\boldsymbol{x_i}, \boldsymbol{x_j})$, $\theta_p \in \boldsymbol{\theta}$ is a kernel learnable parameter, and Tr denotes the trace operator.

3.2.3 Similarity-based Demonstration Selection Based on the Kernel Function

Similarity-based demonstration selection, which identifies the examples most similar to the test sample, has proven to be optimal for ICL (Liu et al., 2022). This paper leverages the kernel function of the trained MK-GP model $\mathbb{G}$ to perform **similarity-based demonstration selection**.

For a given test sample $\boldsymbol{x_0}$, the similarity between $\boldsymbol{x_0}$ and an example $\boldsymbol{x_i} \in \mathcal{D}$ is quantified using the following formulation:

$$sim(\boldsymbol{x_0}, \boldsymbol{x_i}) = k(\boldsymbol{x_0}, \boldsymbol{x_i}), \quad (8)$$

A larger kernel function value signifies a greater similarity between $\boldsymbol{x_0}$ and $\boldsymbol{x_i}$ in the feature space (Thickstun, 2019), with more details in Appendix C.

The S examples with the highest similarity values to $\boldsymbol{x_0}$ are selected as demonstration examples. These examples, along with their corresponding labels, denoted as $\{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_S}, y_S)\}$, are then concatenated with the classification rationale $\mathbb{R}$ to form a 'prompt' for the LLM. This process is defined as follows:

$$\hat{y_0} = \text{LLM}(\boldsymbol{x_0} \oplus \mathbb{R} \oplus (\boldsymbol{x_1}, y_1) \oplus \cdots \oplus (\boldsymbol{x_S}, y_S))$$

where $\hat{y_0}$ is the predicted label for $\boldsymbol{x_0}$, and $\oplus$ represents the concatenation operation.

4 Experimental Setup

4.1 Experimental Datasets

To evaluate RD-MCSA, experiments were conducted on five diverse datasets across various domains and sentiment classification granularities as shown in Table 1:

Dataset	Size	Classes	Granularity & Text type
SST5[1]	11,855	5	Sentence-level Movie Reviews
SemEval17[2]	20,632	5	Topic-based Tweets
PR_Baby[3]	183,531	5	Baby-product Reviews
PR_Software[4]	12,804	5	Software Product Reviews
ABSIA[5]	4,650	7	Restaurant-related Reviews

Table 1: Summary of Experimental Datasets

These datasets cover a range of sentiment classification tasks, from sentence-level analysis to fine-grained aspect-based sentiment analysis, enabling a comprehensive evaluation of RD-MCSA.

4.2 Experimental Implementation Details

In our experiments, we randomly sample 1,000 instances from each dataset to construct the annotated dataset $\mathcal{D}$, ensuring a fair evaluation of RD-MCSA across datasets. This also provides insights into the amount of labeled data required for MCSA tasks, helping determine the annotation needed to outperform traditional classifiers trained on large-scale datasets. The balanced coreset size for generating the classification rationale is set to $\lambda_{\mathcal{B}} = 100$. Taking into account both economy and effectiveness, the number of demonstrations is set to $S = 10$.

We conduct experiments using two LLMs: GPT[6] and DeepSeek[7]. Specifically, GPT-4o is employed for classification rationale generation, while GPT-4o-mini, a cost-efficient model, is utilized for

[1]https://huggingface.co/datasets/SetFit/sst5
[2]https://huggingface.co/datasets/midas/semeval2017
[3]https://snap.stanford.edu/data/web-Amazon-links.html
[4]https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2
[5]https://www.iitp.ac.in/ãi-nlp-ml/resources.html#ABSIA

ICL in MCSA to handle large-scale datasets. For DeepSeek, DeepSeek-R1 is used for classification rationale generation, and DeepSeek-V3 is applied for ICL in MCSA.

In the MK-GP model, we set $n = 9$ for the Matérn Kernel (Formula 5) and $m = 9$ for the Polynomial Kernel (Formula 6) across all datasets. The degrees for the kernels were configured as $d_1, d_2, d_3 = 1$; $d_4, d_5, d_6 = 2$; and $d_7, d_8, d_9 = 3$. The Adam optimizer was employed to minimize the loss function, with a learning rate of 0.01 over 500 training epochs. All other optimizer parameters were set to their default values. The optimal hyperparameters were determined through grid search and cross-validation.

The majority of our experiments were conducted on an NVIDIA GeForce RTX 3080 GPU, where a single unit of this GPU takes approximately 170.86 seconds to complete 500 epochs of Gaussian process training. As for the utilization of LLMs, we employ an API-based approach for their invocation.

4.3 Comparison Models

We select **baseline models** from two categories: (1) classic machine learning and (2) language models for sentiment classification. The chosen models are: 1) **Naïve Bayes** (Rennie, 2001): Multinomial Naive Bayes with Tf-idf features and SMOTE oversampling or random undersampling for data imbalance. 2) **SVM** (Li et al., 2011): Linear kernel Support Vector Classifier with balanced class weights and Tf-idf features. 3) **BERT** (Sun et al., 2019): BERT-base model with 'Focal Loss' to handle data imbalance. 4) **BERTweet** (Nguyen et al., 2020): Pretrained model for English tweets, using 'Focal Loss' for imbalance.

All the baseline models are trained and evaluated across five datasets with an 80%/20% train-test split.

Given that ICL approaches have recently achieved state-of-the-art performance in text classification, several ICL selection approaches are included as **comparison methods**: 1) **Random**: Randomly selects unique in-context examples from the candidate set. 2) **Coreset** (Indyk et al., 2014): Select samples that are representative of the overall diversity present in the full dataset. 3) **Cos-Similarity** (de Vos et al., 2022): Selects the top-S examples based on cosine similarity. 4) **BM25** (Robertson et al., 2009): Selects the top-S examples based on BM25 scoring. 5) **Complex-CoT** (Fu et al., 2022): Selects examples based on their complexity, quantified by newline characters. 6) **Auto-CoT** (Zhang et al., 2022b): Clusters candidate examples and selects those closest to each cluster center.

To ensure a fair comparison, these ICL-based methods use the same 1,000 labeled samples as RD-MCSA, with 100 demonstrations (S=100). Additionally, all prompts include classification rationales generated by the same method.

For further analysis, **ablation studies** are conducted with the following models: 1) **CR-only**: Uses only classification rationales in the prompt, excluding demonstration examples. 2) **DE-only**: Uses only demonstration examples, excluding classification rationales. 3) **LLM-only**: Relies solely on the LLM's inherent reasoning for classification, without classification rationales or demonstration examples. 4) **UnBa-CR**: Excludes label balance in coreset selection for classification rationale generation.

4.4 Evaluation Metric

Due to the multi-class nature of MCSA and the class imbalance in the experimental data, we use the Accuracy and weighted-average F1 score to evaluate each model's performance (Sokolova and Lapalme, 2009).

5 Experimental Results and Analysis

5.1 Main Results

Table 2 summarizes the performance of various models on sentiment classification tasks across three datasets (results for the remaining two datasets are provided in Table 4 in Appendix D). Based on the experimental results, the following conclusions can be drawn:

1) **Effectiveness of In-Context Learning:** Across the experiments conducted on multiple datasets, the In-Context Learning method achieved the best performance in terms of both Accuracy and weighted-average F1 score. This highlights its superior effectiveness for the MCSA task compared to traditional machine learning approaches and language model-based classification methods. Notably, the In-Context Learning method accomplished this using only 1,000 samples as the example pool, whereas baseline methods

[6] https://openai.com/api/
[7] https://www.deepseek.com/

Table 2: Experimental Results of Baseline Models and ICL Comparison Methods.

	Method	SST5		SemEval17		ABSIA	
		Acc (%)	F1 (%)	Acc (%)	F1 (%)	Acc (%)	F1 (%)
Baseline Models	Naïve Bayes	37.2	37.0	44.9	44.0	34.8	31.0
	SVM	37.1	37.0	56.7	58.0	49.9	50.0
	BERT	**49.9**	**50.0**	59.2	61.0	51.2	**52.0**
	BERTweet	48.7	47.0	**63.4**	**65.0**	**52.4**	**52.0**
ICL based on GPT-4o +GPT-4o-mini	Random	55.0	54.9	57.7	60.22	51.6	52.87
	Coreset	55.7	55.44	59.4	62.07	53.2	55.39
	Cos-Similarity	55.6	55.08	60.1	61.92	52.8	53.58
	BM25	56.5	56.02	61.6	63.53	53.0	54.66
	Complex-CoT	56.5	54.3	62.5	63.12	52.9	55.26
	Auto-CoT	56.6	54.18	62.2	63.09	53.4	55.62
	RD-MCSA	**57.6**	**56.03**	**63.9**	**64.69**	**54.3**	**56.01**
ICL based on DeepSeek-R1 +DeepSeek-V3	Random	56.1	55.18	67.2	67.71	51.2	53.26
	Coreset	56.2	55.09	67.6	68.4	52.7	53.98
	Cos-Similarity	56.3	55.21	68.4	**68.62**	53.2	55.41
	BM25	56.6	55.75	67.3	67.99	53.1	54.72
	Complex-CoT	56.1	53.84	67.5	67.31	52.2	53.36
	Auto-CoT	56.3	54.64	67.7	68.11	52.7	54.99
	RD-MCSA	**57.9**	**57.0**	**68.6**	68.55	**54.6**	**56.5**

were trained on 80% of the data (typically tens of thousands of samples). This further underscores the efficiency and effectiveness of In-Context Learning in leveraging limited data for robust performance.

2) **Effectiveness of the RD-MCSA Method:** In the experiments conducted across multiple datasets, the RD-MCSA method achieved the best performance on all metrics, except for the SemEval17 dataset, where the cosine similarity-based In-Context Learning method attained the highest weighted-average F1 score. These results validate the effectiveness of the RD-MCSA method. Further validation of its individual components will be provided in the subsequent ablation experiments.

3) **Comparison of various demonstrations selection methods:** In our experiments across multiple datasets, the Coreset method (diversity-based sample selection), Complex-CoT (sample complexity-based selection), Auto-CoT (cluster center-based selection), and three similarity-based methods (BM25, Cosine Similarity, and RD-MCSA) consistently outperformed random sample selection. This highlights the overall effectiveness of structured example selection strategies. Among the similarity-based methods, RD-MCSA, which leverages Gaussian processes to learn a kernel function, demonstrated superior capability in measuring sample similarity. Compared to BM25 and Cosine Similarity, RD-MCSA more effectively identifies samples closely aligned with the target classification samples, thereby enhancing the classification performance of LLMs in the MCSA task.

5.2 Ablation Analysis

Table 3 presents the results of the ablation study conducted on two datasets. Based on the ablation experiments, we can draw the following conclusions regarding the components of the algorithm:

1) **Effectiveness of In-Context Learning:** Overall, in the experiments conducted on various datasets in this paper, compared to directly querying the LLM for text classification, incorporating classification rationales resulted in an improvement in the LLM's classification performance. This is likely because classi-

Table 3: Experimental Results of Ablation Studies.

	SST5		SemEval17		PR_Baby		PR_Software		ABSIA	
	Acc (%)	F1 (%)	Acc (%)	F1 (%)	Acc (%)	F1 (%)	Acc (%)	F1 (%)	Acc (%)	F1 (%)
GPT-based:										
LLM-only	54.2	52.62	56.0	58.88	57.4	57.5	62.1	63.32	50.2	52.11
CR-only	54.9	54.42	57.0	59.81	57.8	57.9	62.4	63.89	50.9	52.68
DE-only	55.8	54.07	59.1	61.03	59.7	59.81	65.6	65.97	52.0	53.31
UnBa-CR	57.1	54.82	62.0	62.81	59.6	59.62	66.1	65.82	53.7	56.01
Final	**57.6**	**56.03**	**63.9**	**64.69**	**60.1**	**60.32**	**67.0**	**67.22**	**54.3**	**56.01**
DeepSeek-based:										
LLM-only	54.1	53.76	55.8	59.85	55.5	55.5	56.6	58.37	49.2	51.24
CR-only	54.97	52.88	66.5	67.23	55.9	55.96	58.8	60.58	50.1	51.9
DE-only	57.0	56.18	67.9	67.96	56.9	57.09	65.9	66.94	52.6	53.36
UnBa-CR	57.4	55.99	67.8	67.48	57.1	57.3	66.7	66.39	54.1	55.38
Final	**57.9**	**57.0**	**68.6**	**68.55**	**57.5**	**57.7**	**67.7**	**68.11**	**54.6**	**56.5**

fication rationales enable the LLM to better understand the specific meaning of each class label, thereby enhancing its ability to perform text classification.

2) **Effectiveness of Demonstration Examples:** In the experiments conducted on various datasets, compared to directly querying the LLM for text classification, the inclusion of demonstration examples led to a significant improvement in the LLM's classification performance. This suggests that, compared to other modules, demonstration examples play a more crucial role in the LLM's text classification process.

3) **Effectiveness of Label Balance in Classification Rationales Generation:** In the experiments conducted on various datasets in this paper, when the samples used to generate classification rationales were imbalanced in terms of class distribution, the classification performance showed a noticeable decline compared to when class-balanced samples were used. This may be because, when the sample categories are imbalanced, the number of samples from certain classes may be too small, making it difficult for the LLM to truly understand the meaning of those classes, and subsequently harder to generate effective classification rationales.

6 Conclusions

This paper introduces a novel multi-class sentiment analysis (MCSA) framework that combines balanced-coreset-based classification with multi-kernel Gaussian processes (MK-GP) for similarity assessment. The proposed approach effectively tackles critical challenges, including class imbalance and the high cost of large-scale annotation, while capturing subtle and nuanced sentiment expressions. Extensive experiments conducted across five diverse datasets demonstrate the superior performance and robustness of our method.

Future research directions include extending the framework to other sentiment analysis tasks, integrating multimodal data (e.g., audio and visual inputs), enhancing computational efficiency, and developing strategies to mitigate the impact of subjective annotations. These advancements contribute to the field by offering valuable insights and laying the groundwork for more accurate and scalable sentiment analysis systems.

Ethics Statement

Our study uses publicly available datasets, and no personally identifiable information is included. We acknowledge potential biases in sentiment classification tasks and have taken steps to mitigate them, such as dataset balancing and bias analysis. No human subjects were involved in the study, and no additional ethical approval was required. While our method could be used for sentiment analysis applications, we do not foresee direct misuse. We will release the code and models responsibly, ensuring compliance with ethical guidelines.

LLMs (mainly GPT) are applied in our writing to help correct grammatical and word usage errors, but they do not generate any ideas, data, images, or

tables for us.

Limitations

This paper has the following limitations:

1. While our method has been validated on five diverse datasets, its applicability remains limited. In particular, we have not tested it on multimodal datasets, which are increasingly relevant.

2. The overall performance of our method remains suboptimal. Even traditional supervised models trained on tens of thousands of samples struggle to exceed 80% accuracy. A key challenge in MCSA tasks is the inherent subjectivity of annotations—different annotators may assign different labels to the same sample, limiting classification performance. Additionally, the quality of the datasets may not be ideal, but we have yet to conduct an in-depth analysis of this aspect.

3. Although the multi-kernel Gaussian process (MK-GP) method achieves strong results, it is computationally slower than other similarity evaluation approaches. Enhancing its efficiency is an important direction for future work, which we have not yet explored.

References

Arwa SM AlQahtani. 2021. Product sentiment analysis for amazon reviews. *International Journal of Computer Science & Information Technology (IJCSIT) Vol*, 13.

Artem Artemev, David R Burt, and Mark van der Wilk. 2021. Tighter bounds on the log marginal likelihood of gaussian process regression using conjugate gradients. In *International Conference on Machine Learning*, pages 362–372. PMLR.

Edwin V Bonilla, Kian Chai, and Christopher Williams. 2007. Multi-task gaussian process prediction. *Advances in neural information processing systems*, 20.

Viacheslav Borovitskiy, Iskander Azangulov, Alexander Terenin, Peter Mostowsky, Marc Deisenroth, and Nicolas Durrande. 2021. Matérn gaussian processes on graphs. In *International Conference on Artificial Intelligence and Statistics*, pages 2593–2601. PMLR.

Yunmo Chen, Tongfei Chen, Harsh Jhamtani, Patrick Xia, Richard Shin, Jason Eisner, and Benjamin Van Durme. 2024. Learning to retrieve iteratively for in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7156–7168, Miami, Florida, USA. Association for Computational Linguistics.

Yun-Shiuan Chuang, Agam Goyal, Nikunj Harlalka, Siddharth Suresh, Robert Hawkins, Sijia Yang, Dhavan Shah, Junjie Hu, and Timothy Rogers. 2024. Simulating opinion dynamics with networks of LLM-based agents. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3326–3346, Mexico City, Mexico. Association for Computational Linguistics.

Keith Cortis, André Freitas, Tobias Daudert, Manuela Huerlimann, Manel Zarrouk, Siegfried Handschuh, and Brian Davis. 2017. Semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 519–535.

Dayan de França Costa and Nadia Felix Felipe da Silva. 2018. Inf-ufg at fiqa 2018 task 1: predicting sentiments and aspects on financial tweets and news headlines. In *Companion Proceedings of the The Web Conference 2018*, pages 1967–1971.

Isa M Apallius de Vos, Ghislaine L Boogerd, Mara D Fennema, and Adriana D Correia. 2022. Comparing in context: Improving cosine similarity measures with a metric tensor. *arXiv preprint arXiv:2203.14996*.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations*.

Alireza Ghasempour and Manel Martínez-Ramón. 2023. Multiple output sparse gaussian processes with multiple kernel learning for electric load forecasting. In *2023 5th International Conference on Power and Energy Technology (ICPET)*, pages 987–990. IEEE.

Piotr Indyk, Sepideh Mahabadi, Mohammad Mahdian, and Vahab S Mirrokni. 2014. Composable core-sets for diversity and coverage maximization. In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 100–108.

Lakshmi Revathi Krosuri and Rama Satish Aravapalli. 2023. Novel heuristic-based hybrid resnext with recurrent neural network to handle multi class classification of sentiment analysis. *Machine Learning: Science and Technology*, 4(1):015033.

Gregory F Lawler. 2018. *Introduction to stochastic processes*. Chapman and Hall/CRC.

Lung-Hao Lee, Liang-Chih Yu, Suge Wang, and Jian Liao. 2024. Overview of the sighan 2024 shared task for chinese dimensional aspect-based sentiment analysis. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*, pages 165–174.

Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse demonstrations improve in-context compositional generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1401–1422, Toronto, Canada. Association for Computational Linguistics.

Kunlun Li, Jing Xie, Xue Sun, Yinghui Ma, and Hui Bai. 2011. Multi-class text categorization based on lda and svm. *Procedia Engineering*, 15:1963–1967.

Xiaonan Li and Xipeng Qiu. 2023. Finding support examples for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6219–6235, Singapore. Association for Computational Linguistics.

Haitao Liu, Yew-Soon Ong, Ziwei Yu, Jianfei Cai, and Xiaobo Shen. 2021. Scalable gaussian process classification with additive noise for non-gaussian likelihoods. *IEEE transactions on cybernetics*, 52(7):5842–5854.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Mamta and Asif Ekbal. 2023. Service is good, very good or excellent? towards aspect based sentiment intensity analysis. In *European Conference on Information Retrieval*, pages 685–700. Springer.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets. *arXiv preprint arXiv:2005.10200*.

Emilio Porcu, Moreno Bevilacqua, Robert Schaback, and Chris J Oates. 2024. The matérn model: A journey through statistics, numerical analysis and machine learning. *Statistical Science*, 39(3):469–492.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, and Tony Lindgren. 2024. CICLe: Conformal in-context learning for largescale multi-class food risk classification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7695–7715, Bangkok, Thailand. Association for Computational Linguistics.

Jason DM Rennie. 2001. Improving multi-class text classification with naive bayes.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2019. Semeval-2017 task 4: Sentiment analysis in twitter. *arXiv preprint arXiv:1912.00741*.

Neeraj Anand Sharma, ABM Shawkat Ali, and Muhammad Ashad Kabir. 2024. A review of sentiment analysis: tasks, applications, and deep learning techniques. *International journal of data science and analytics*, pages 1–38.

Peng Shi, Rui Zhang, He Bai, and Jimmy Lin. 2022. XRICL: Cross-lingual retrieval-augmented in-context learning for cross-lingual text-to-SQL semantic parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5248–5259, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Marina Sokolova and Guy Lapalme. 2009. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437.

Zhao Song, David Woodruff, Zheng Yu, and Lichen Zhang. 2021. Fast sketching of polynomial kernels of polynomial degree. In *International Conference on Machine Learning*, pages 9812–9823. PMLR.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese computational linguistics: 18th China national conference, CCL 2019, Kunming, China, October 18–20, 2019, proceedings 18*, pages 194–206. Springer.

John Thickstun. 2019. Mercer's theorem. *University of Washington, dostupné na internete (5.2. 2018): https://homes. cs. washington. edu/˜ thickstn/docs/mercer. pdf*.

Jie Wang. 2023. An intuitive tutorial to gaussian processes regression. *Computing in Science & Engineering*.

Zhaoxia Wang, Zhenda Hu, Seng-Beng Ho, Erik Cambria, and Ah-Hwee Tan. 2023. Mimusa—mimicking human language understanding for fine-grained multi-class sentiment analysis. *Neural Computing and Applications*, 35(21):15907–15921.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, Toronto, Canada. Association for Computational Linguistics.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Pan, and Lidong Bing. 2024. Sentiment analysis in the era of large language models: A reality check. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3881–3906, Mexico City, Mexico. Association for Computational Linguistics.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022a. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022b. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

A Prompts for Classification Rationale Generation and MCSA

1) Prompt Template for Classification Rationale Generation

You are an expert in sentiment analysis. Based on the examples provided below, generate detailed descriptions for each sentiment label.

Examples: {examples_str}

Sentiment Labels: {str(label_list)}

For each sentiment label, provide a comprehensive description covering: 1) Linguistic features; 2) Semantic features; 3) Characteristics of the target or aspect of sentiment expression.

Clearly define distinguishing classification criteria and identify specific words, phrases, and expressions from the examples. Each description should be approximately {description_words} words.

2) Prompt Template for MCSA

As a sentiment analysis model, analyze the sentiment of the given text, which contains texts about {target}.

Your classification should be based on the following {len(label_list)} Sentiment labels: {str(label_list)}

Label_description: {label_description}

Demonstration Examples: {examples}

If the input is nonsensical or meaningless, classify it as intermediate values of {str(label_list)}.

Now, analyze the following text about {target}: {query_text}

When performing analysis, label distribution also needs to be considered.

Label Distribution: {label_distribution}

B Definition and Properties of the Matérn Kernel of MK-GP

The Matérn kernel is defined as follows, where ν and ℓ are the kernel parameters:

$$k_{\text{Matérn}}(x_i, x_j) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|x_i - x_j\|}{\ell} \right)^{\nu} B_\nu \left(\sqrt{2\nu} \frac{\|x_i - x_j\|}{\ell} \right),$$

where $\Gamma(\nu)$ represents the Gamma function, defined as:

$$\Gamma(\nu) = \int_0^\infty t^{\nu-1} e^{-t}\, dt.$$

Here, $B_\nu(z)$ denotes the modified Bessel function of the second kind, defined as:

$$B_\nu(z) = \frac{\pi}{2} \frac{I_{-\nu}(z) - I_\nu(z)}{\sin(\nu\pi)},$$

where $I_\nu(z)$ is the modified Bessel function of the first kind, given by:

$$I_\nu(z) = \sum_{k=0}^{\infty} \frac{\left(\frac{z}{2}\right)^{\nu+2k}}{k!\Gamma(\nu + k + 1)}.$$

When the parameter $\nu \to \infty$, the Matérn kernel converges to the Radial Basis Function (RBF) kernel (Porcu et al., 2024):

$$\lim_{\nu\to\infty} k_{\text{Matérn}}(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\ell^2} \right).$$

When the parameter $\nu = \frac{1}{2}$, the Matérn kernel becomes equivalent to the Laplace kernel [54]:

$$k_{\text{Matérn}}(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|}{\ell} \right), \text{when } \nu = \frac{1}{2}.$$

C Similarity Evaluation Based on Kernel Functions of MK-GP

According to Mercer's theorem (Thickstun, 2019), there exists a Hilbert space $\mathcal{H}$ and a mapping $\phi : \mathcal{X} \to \mathcal{H}$ such that the kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$ can be expressed as the inner product in the Hilbert space:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}}, \quad \forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}.$$

Table 4: Experimental Results of Baseline Models and ICL Comparison Methods on Two Datasets.

	Method	PR_Baby		PR_Software	
		Acc (%)	F1 (%)	Acc (%)	F1 (%)
Baseline Models	Naïve Bayes	47.86	47.0	44.8	45.0
	SVM	50.96	51.0	58.1	59.0
	BERT	**58.18**	**58.0**	**60.3**	**61.0**
	BERTweet	57.74	56.0	59.9	58.0
ICL based on GPT-4o +GPT-4o-mini	Random	57.9	57.88	62.3	63.57
	Coreset	58.1	58.06	62.6	63.68
	Cos-Similarity	58.9	59.03	64.7	65.86
	BM25	59.2	59.36	63.1	64.25
	Complex-CoT	58.4	58.46	65.3	66.38
	Auto-CoT	58.8	59.07	62.7	64.08
	RD-MCSA	**60.1**	**60.32**	**67.0**	**67.22**
ICL based on DeepSeek-R1 +DeepSeek-V3	Random	56.0	56.13	61.5	62.94
	Coreset	56.3	56.42	63.5	64.54
	Cos-Similarity	56.6	56.72	64.5	65.91
	BM25	56.6	56.74	63.9	65.09
	Complex-CoT	56.4	56.58	65.7	65.29
	Auto-CoT	56.5	56.64	63.2	64.49
	RD-MCSA	**57.5**	**57.7**	**67.7**	**68.11**

Here, $\phi(\mathbf{x})$ is an implicitly defined mapping, and $\mathcal{H}$ is the corresponding Hilbert space. In $\mathcal{H}$, the Euclidean distance between any two samples $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$ is defined as:

$$\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 = \langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_i)\rangle_{\mathcal{H}} - 2\langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle_{\mathcal{H}} + \langle\phi(\mathbf{x}_j), \phi(\mathbf{x}_j)\rangle_{\mathcal{H}}.$$

By utilizing the definition of the kernel function, $k(\mathbf{x}_i, \mathbf{x}_j) = \langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle_{\mathcal{H}}$, the expression can be rewritten as:

$$\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 = k(\mathbf{x}_i, \mathbf{x}_i) - 2k(\mathbf{x}_i, \mathbf{x}_j) + k(\mathbf{x}_j, \mathbf{x}_j).$$

This represents the distance in the Hilbert space induced by a positive definite kernel function. After normalizing the samples, for the kernel function adopted in this study, the first and third terms in the above equation become constants. Thus, the larger the value of the middle term $k(\mathbf{x}_i, \mathbf{x}_j)$, the smaller the distance between $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$, indicating that the two samples are more similar.

D Experimental Results on Other Datasets

Table D presents the experimental results of baseline models and in-context learning (ICL) comparison models on the remaining two datasets.