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

**Haihua Xie**[1*], **Yinzhu Cheng**[2,1*], **Yaqing Wang**[1], **Miao He**[1], **Mingming Sun**[1]
[1] Beijing Institute of Mathematical Sciences and Applications, Beijing, China
[2] Institute of Statistics and Big Data, Renmin University of China, Beijing, China
{haihua.xie, chengyinzhu, wangyaqing, miaohe, sunmingming}@bimsa.cn

## Abstract

This paper addresses the important yet underexplored task of **multi-class sentiment analysis (MCSA)**, which remains challenging due to the subtle semantic differences between adjacent sentiment categories and the scarcity of high-quality annotated data. To tackle these challenges, we propose **RD-MCSA** (**R**ationales and **D**emonstrations-based **M**ulti-**C**lass **S**entiment **A**nalysis), an In-Context Learning (ICL) framework designed to enhance MCSA performance under limited supervision by integrating classification rationales with adaptively selected demonstrations. First, semantically grounded classification rationales are generated from a representative, class-balanced subset of annotated samples selected using a tailored balanced coreset algorithm. These rationales are then paired with demonstrations chosen through a similarity-based mechanism powered by a **multi-kernel Gaussian process (MK-GP)**, enabling large language models (LLMs) to more effectively capture fine-grained sentiment distinctions. Experiments on five benchmark datasets demonstrate that RD-MCSA consistently outperforms both supervised baselines and standard ICL methods across various evaluation metrics.

## 1 Introduction

Multi-class sentiment analysis (MCSA) goes beyond basic sentiment polarity classification (e.g., *positive* or *negative*) by distinguishing varying levels of emotional intensity (e.g., *very positive* vs. *generally positive*). By capturing finer sentiment distinctions, MCSA provides deeper insights into sentiment expression, making it essential for applications that require fine-grained sentiment analysis (Wang et al., 2023). For example, in opinion dynamics research, a key step involves categorizing users' natural language expressions into five or more sentiment categories (Chuang et al., 2024).

Despite its importance, MCSA remains challenging due to subtle semantic differences between adjacent sentiment levels, which are often difficult to distinguish accurately (Mamta and Ekbal, 2023). Moreover, sentiment categorization criteria can vary considerably across domains and applications (Rosenthal et al., 2019), further complicating the modeling process. Addressing a new MCSA task typically requires a substantial amount of high-quality, task-specific annotated data (Krosuri and Aravapalli, 2023), which is often scarce in low-resource settings.

Large language models (LLMs) have demonstrated strong performance in sentiment analysis, making them a promising tool for MCSA. However, although LLMs perform well in basic sentiment classification, they often struggle with nuanced distinctions between adjacent sentiment categories (Zhang et al., 2024). In-context learning (ICL), which enhances LLM capabilities through a few task-specific demonstrations, has achieved state-of-the-art performance across various NLP tasks. Nevertheless, its application to classification settings involving multiple sentiment categories remains underexplored (Randl et al., 2024). Moreover, our experimental results show that conventional ICL approaches are insufficient for effectively addressing the complexity of MCSA.

To address these limitations, this paper proposes **RD-MCSA**, a novel framework aimed at improving ICL performance for MCSA. RD-MCSA refines the two core components of ICL—*prompt design* and *demonstration selection*—by incorporating **classification rationales** and an **adaptive example selection** mechanism. This design enables LLMs to better capture fine-grained sentiment distinctions and improve classification accuracy.

The main contributions of this paper are summarized as follows:

1. **Rationale-Augmented ICL**: An ICL frame-

---

[*] Equal contribution.

*Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21700–21723
November 4-9, 2025 ©2025 Association for Computational Linguistics

work that integrates classification rationales and demonstration examples is proposed, enabling LLMs to more effectively capture fine-grained sentiment distinctions in MCSA.

2. **Classification Rationale Generation via Tailored Balanced Coreset**: A rationale generation strategy is designed that guides LLMs to produce linguistically and semantically rich classification rationales, based on representative and class-balanced samples selected through a tailored balanced Coreset algorithm.

3. **Adaptive Demonstration Selection via MK-GP**: A novel demonstration selection method based on a multi-kernel Gaussian process (MK-GP) is proposed, enabling adaptive similarity modeling beyond fixed metrics such as cosine similarity, marking the first use of kernel-based selection in the ICL setting.

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

## 2 Related Work

### 2.1 Multi-Class Sentiment Analysis

Multi-class sentiment analysis (MCSA), also referred to as fine-grained or graded sentiment analysis (Sharma et al., 2024), extends traditional sentiment classification by mapping texts into multiple sentiment categories rather than a simple polarity distinction. Unlike binary sentiment analysis (e.g., *positive* vs. *negative*), MCSA captures variations in sentiment intensity by introducing subcategories such as *very positive* or *slightly positive*, or by employing rating scales (e.g., 1–5) (AlQahtani, 2021). This finer granularity enables a more nuanced understanding of sentiment and serves as a fundamental task in diverse applications, including business intelligence, public opinion monitoring, and disaster assessment (Sharma et al., 2024).

Traditional MCSA models rely on supervised machine learning (Wang et al., 2023) and are commonly applied to texts such as tweets, movie reviews, and product reviews. In many cases, sentiment analysis also focuses on specific targets or aspects. Widely used 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 ongoing advances, MCSA still faces key challenges, such as limited accuracy and the high cost of large-scale annotation, especially as sentiment granularity increases (Krosuri and Aravapalli, 2023). Fine-grained sentiment analysis for specific entities often requires distinct annotated datasets, making large-scale deployment impractical.

To address these challenges, this study aims to enhance MCSA performance under limited labeled data conditions, while maintaining broad applicability across diverse MCSA scenarios.

### 2.2 Text Analysis Using LLMs

LLMs have demonstrated strong few-shot learning capabilities through in-context learning, which provides a promising direction for achieving efficient MCSA (Wu et al., 2025).

Recent research on LLM-based text analysis has focused on ICL, where carefully selected demonstration examples guide model predictions. Common strategies for selecting examples include similarity-based selection (Liu et al., 2022), diversity-based selection (Zhang et al., 2022), information-theoretic criteria (Wu et al., 2023), task-level selection (Li and Qiu, 2023) and LLM feedback (Liu et al., 2024a). For MCSA, a recent study (Chuang et al., 2024) employs similarity-based demonstration selection within ICL to analyze opinion dynamics.

Despite their potential, LLMs still face challenges across many NLP tasks. They often struggle with fine-grained sentiment analysis (Zhang et al., 2024). Moreover, few-shot ICL requires further study on optimal prompt design (Liu et al., 2022), and, to our knowledge, no prior work has examined few-shot prompting for multi-class prediction with a large number of categories (Randl et al., 2024). In addition, long prompts may overload LLMs (Liu et al., 2024b), while limited context windows can restrict the effective representation of all classes.

This study focuses on two key components of ICL, prompt construction and demonstration selection, addressing how to **effectively provide classification information to LLMs** and how to adapt both components to better serve MCSA tasks.

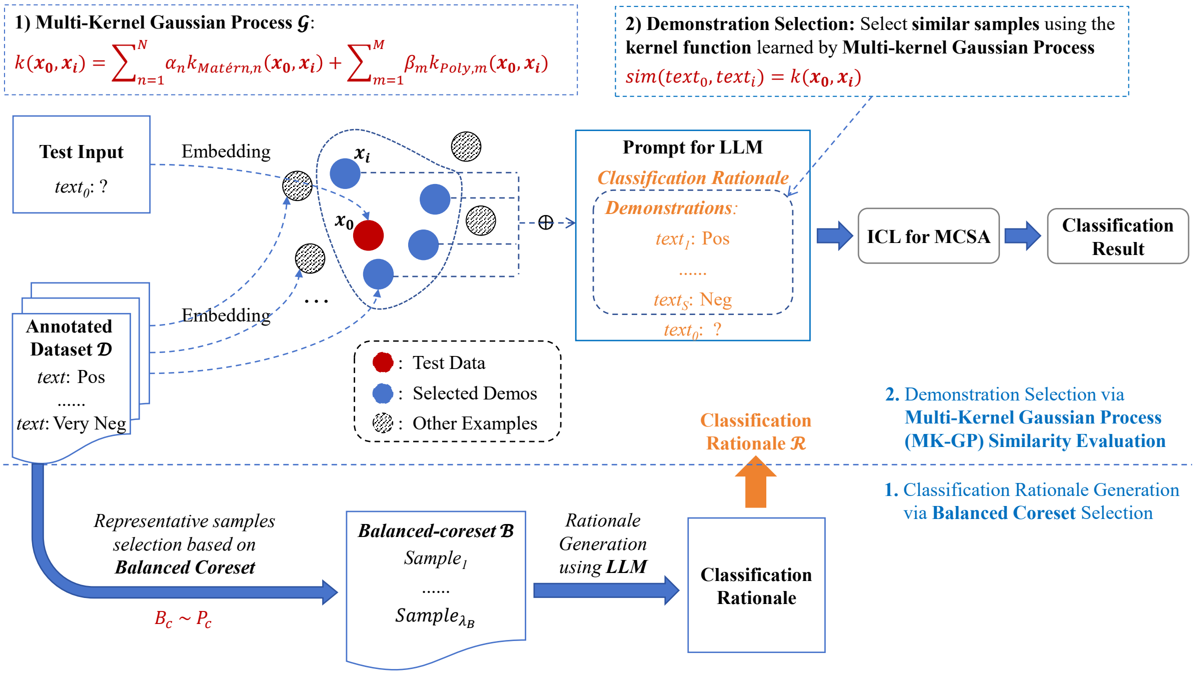

1) Multi-Kernel Gaussian Process $\mathcal{G}$:
$k(\boldsymbol{x_0}, \boldsymbol{x_i}) = \sum_{n=1}^{N} \alpha_n k_{Matérn,n}(\boldsymbol{x_0}, \boldsymbol{x_i}) + \sum_{m=1}^{M} \beta_m k_{Poly,m}(\boldsymbol{x_0}, \boldsymbol{x_i})$
2) Demonstration Selection: Select similar samples using the kernel function learned by Multi-kernel Gaussian Process
$sim(text_0, text_i) = k(\boldsymbol{x_0}, \boldsymbol{x_i})$
Test Input
$text_0$: ?
Embedding
$\boldsymbol{x_i}$
$\boldsymbol{x_0}$
Annotated Dataset $\mathcal{D}$
text: Pos
......
text: Very Neg
Embedding
...
Prompt for LLM
Classification Rationale
Demonstrations:
$text_1$: Pos
......
$text_S$: Neg
$text_0$: ?
ICL for MCSA
Classification Result
: Test Data
: Selected Demos
: Other Examples
Classification Rationale $\mathcal{R}$
2. Demonstration Selection via Multi-Kernel Gaussian Process (MK-GP) Similarity Evaluation
1. Classification Rationale Generation via Balanced Coreset Selection
Representative samples selection based on Balanced Coreset
$B_c \sim P_c$
Balanced-coreset $\mathcal{B}$
$Sample_1$
......
$Sample_{\lambda_\mathcal{B}}$
Rationale Generation using LLM
Classification Rationale

Figure 1: The framework 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.

## 3 The Methodology of RD-MCSA

The RD-MCSA framework, illustrated in Fig. 1, consists of the following key components. Given an annotated MCSA dataset $\mathcal{D}$: 1) a balanced Coreset $\mathcal{B}$ is constructed to generate classification rationales $\mathcal{R}$ (Section 3.1); 2) a multi-kernel Gaussian process $\mathcal{G}$ is trained (Subsection 3.2.2) to model adaptive similarity; 3) for MCSA on a new input, ICL is performed using a prompt that incorporates both $\mathcal{R}$ and a set of demonstrations selected from $\mathcal{D}$ via $\mathcal{G}$ (Subsection 3.2.3).

### 3.1 Classification Rationale Generation via Balanced Coreset Selection

The classification rationales $\mathcal{R}$ are generated by an LLM through reasoning over the semantic and linguistic features of a representative subset of $\mathcal{D}$. To ensure that this subset (denoted as $\mathcal{B}$) preserves the semantic diversity and key distinguishing characteristics of each sentiment class—while also mitigating class imbalance—a **balanced Coreset selection algorithm** is proposed.

#### 3.1.1 The Balanced Coreset Algorithm

The proposed algorithm extends the classical Coreset formulation (Sener and Savarese, 2017) by incorporating **importance-weighted sampling** and **class-aware stratification**, ensuring that the selected subset $\mathcal{B}$ maintains both intra-class diversity and inter-class balance, thereby facilitating higher-quality rationale generation.

To enforce class balance, the number of selected samples per class is capped by $\lambda'_\mathcal{B} = \left\lceil \frac{\lambda_\mathcal{B}}{u} \right\rceil$, where $u$ denotes the number of unique sentiment classes in $\mathcal{D}$, and $\lambda_\mathcal{B}$ is a hyperparameter specifying the total Coreset size.

**1) Importance-Weighted Sampling Probability.** To prioritize semantically informative and potentially ambiguous instances, each sample is assigned a score based on its distance from the centroid of its respective class (Cohen-Addad et al., 2021).

For a given text sample $(t_i, y_i) \in \mathcal{D}$, let $\boldsymbol{x}(t_i) \in \mathbb{R}^d$ denote the embedding[1] of $t_i$, where $y_i = c$ is its class label. The centroid $\boldsymbol{\mu}_c$ of class $c$ is computed as $\boldsymbol{\mu}_c = \frac{1}{|\mathcal{D}_c|} \sum_{j:y_j=c} \boldsymbol{x}(t_j)$, where $\mathcal{D}_c \subset \mathcal{D}$ denotes the set of samples belonging to class $c$. The **importance weight** is defined as the squared Euclidean distance $w(t_i, y_i) = \|\boldsymbol{x}(t_i) - \boldsymbol{\mu}_c\|_2^2$.

Within each class, importance weights are normalized to form a probability distribution. The **sampling probability** of $t_i$, denoted as $P_c(t_i)$, is

---

[1] This paper employs Sentence-BERT (Reimers and Gurevych, 2019) to generate text embeddings.

defined as:

$$P_c(t_i) = \frac{w(t_i, y_i)}{\sum_{j:y_j=c} w(t_j, y_j)}. \tag{1}$$

**2) Stratified Weighted Random Sampling.** Sample selection is performed independently for each class $1 \leq c \leq u$, based on the corresponding sampling probabilities:

- If $|\mathcal{D}_c| \leq \lambda'_{\mathcal{B}}$, all instances from class $c$ are included in $\mathcal{B}$.
- If $|\mathcal{D}_c| > \lambda'_{\mathcal{B}}$, a subset of $\lambda'_{\mathcal{B}}$ samples is drawn from $\mathcal{D}_c$ via weighted sampling with $P_c(t_i)$, forming the subset $\mathcal{B}_c$:

$$\mathcal{B}_c \subset \mathcal{D}_c, \quad |\mathcal{B}_c| = \lambda'_{\mathcal{B}}, \quad \mathcal{B}_c \sim P_c.$$

The final balanced Coreset $\mathcal{B}$ is obtained by aggregating all class-specific subsets $\mathcal{B}_c$.

### 3.1.2 Classification Rationale Generation via LLM Reasoning

To extract class-discriminative knowledge from the coreset $\mathcal{B}$, classification rationales $\mathcal{R}$ are generated using an LLM guided by a carefully designed prompt, as illustrated in Figure 2. Leveraging the advanced reasoning abilities of LLMs (Wang, 2025), this approach provides a scalable and semantically informed alternative to manual annotation.

Furthermore, because LLMs are subsequently employed for ICL in downstream MCSA tasks, generating classification rationales with models from the same family promotes better alignment between rationale formulation and model interpretation.

> Based on the representative examples provided below, generate detailed descriptions for each sentiment label.
>
> **Examples:** {Balanced Coreset $\mathcal{B}$}
> **Sentiment Labels:** {*str(label_list)*}
>
> For each sentiment label, provide a comprehensive description covering:
> - **Lexical Patterns**
> - **Semantic-Pragmatic Features**
> - **Domain-Attribute Associations**

Figure 2: Prompt template for generating classification rationales using the balanced coreset $\mathcal{B}$.

The prompt instructs the LLM to identify key linguistic and semantic features that distinguish sentiment classes, with a focus on: 1) **Lexical Patterns**: characteristic sentiment-bearing words, phrases, and affective expressions; 2) **Semantic-Pragmatic Features**: contextual meaning shifts and pragmatic implications across classes; 3) **Domain-Attribute Associations**: domain-specific entities and properties linked to sentiment expression.

The LLM is further guided to ground its analysis in representative examples from $\mathcal{B}$, referencing specific lexical or syntactic patterns. This ensures that the resulting rationales are both interpretable and empirically supported.

## 3.2 Demonstration Selection via Multi-Kernel Gaussian Process Similarity Evaluation

RD-MCSA leverages a **multi-kernel Gaussian process for text similarity evaluation** to select ICL demonstrations. This method benefits from Multiple Kernel Learning's ability to model and adapt to complex data distributions (Ghasempour and Martínez-Ramón, 2023).

### 3.2.1 Gaussian Process

Gaussian Process (GP) (Liu et al., 2021) can be applied to model categorical data with $u$ categories by introducing a set of latent functions $\{f_c(\boldsymbol{x})\}_{c=1}^{u}$, one for each class. Each latent function is modeled as an independent Gaussian Process (Wang, 2023):

$$f_c(\boldsymbol{x}) \sim \mathcal{GP}(e_c(\boldsymbol{x}), k_c(\boldsymbol{x}, \boldsymbol{x'})), \tag{2}$$

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.

Following prior work such as (Bonilla et al., 2007), this study adopts a shared kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and a shared mean function across all categories. This design choice not only reduces computational complexity but also capitalizes on structural similarities commonly observed among different classes within the same dataset. In this framework, the mean function is modeled as a learnable constant, and the kernel is defined as a multi-kernel function, as described in Section 3.2.2.

### 3.2.2 Multi-Kernel Gaussian Process

The Multi-Kernel Gaussian Process (MK-GP) extends the standard Gaussian Process framework by incorporating Multiple Kernel Learning, and has demonstrated effectiveness in sentiment and emotion analysis on text (S et al., 2017b,a; Angel Deborah et al., 2021). In this work, we adopt a weighted combination of the Matérn kernel (Borovitskiy et al., 2021) and the polynomial kernel (Song et al., 2021), enabling the model to **capture both stationary and non-stationary patterns in the data** (Lawler, 2018). The resulting composite kernel is formally defined as follows:

$$k(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{n=1}^{N} \alpha_n k_{\text{Matérn},n}(\boldsymbol{x_i}, \boldsymbol{x_j}) + \sum_{m=1}^{M} \beta_m k_{Poly,m}(\boldsymbol{x_i}, \boldsymbol{x_j}), \tag{3}$$

where $k_{\text{Matérn},n}(\boldsymbol{x_i}, \boldsymbol{x_j})$ denotes the $n$-th Matérn kernel, and $k_{\text{Poly},m}(\boldsymbol{x_i}, \boldsymbol{x_j})$ denotes the $m$-th polynomial kernel. The coefficients $\alpha_n$ and $\beta_m$ are learnable weights constrained to be non-negative ($\alpha_n, \beta_m \geq 0$). Additional details are provided in Appendix A.1.

Let $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^{K}$ denote the training data and $\boldsymbol{y}$ the corresponding labels. Let $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), \ldots, f_u(\boldsymbol{x})]^T$ denote the vector of latent function values at input $\boldsymbol{x}$, where $u$ denotes the number of classes in the dataset, and let $\boldsymbol{f} = \{\boldsymbol{f}(\boldsymbol{x}_i)\}_{i=1}^{K}$ denote the collection of latent outputs over the training set. An MK-GP model $\mathcal{G}$ is trained by minimizing the **negative log-marginal likelihood** (Artemev et al., 2021):

$$\mathcal{L} = -\log \int p(\boldsymbol{y} \mid \boldsymbol{f})\, p(\boldsymbol{f} \mid \boldsymbol{X})\, d\boldsymbol{f}. \tag{4}$$

Further details regarding the training procedure of MK-GP are provided in Appendix A.2.

#### 3.2.3 Similarity-Based Demonstration Selection via the Kernel Function

Similarity-based demonstration selection, which selects examples most similar to the test sample, has proven effective for ICL (Margatina et al., 2023). In this work, we adopt a similarity-based approach leveraging the kernel function of the trained MK-GP model $\mathcal{G}$ to guide demonstration selection. Given a test sample $t_0$, its similarity to a candidate example $t_i \in \mathcal{D}$ is computed as:

$$sim(t_0, t_i) = k(\boldsymbol{x}(t_0), \boldsymbol{x}(t_i)), \tag{5}$$

where $\boldsymbol{x}(t_0)$ and $\boldsymbol{x}(t_i)$ (or, for brevity, $\boldsymbol{x}_0$ and $\boldsymbol{x}_i$) are the embeddings of $t_0$ and $t_i$, respectively. As shown in Figure 3, the embeddings are mapped into a Hilbert space via a kernel function. With a well-chosen kernel, the transformed representations exhibit improved class separability relative to the original embedding space (Elen et al., 2022). This enhanced structure enables more discriminative similarity computation for ICL. A higher kernel value (as learned in Section 3.2.2) reflects greater similarity between examples in the feature space (Thickstun, 2019). Additional implementation details are provided in Appendix A.3.

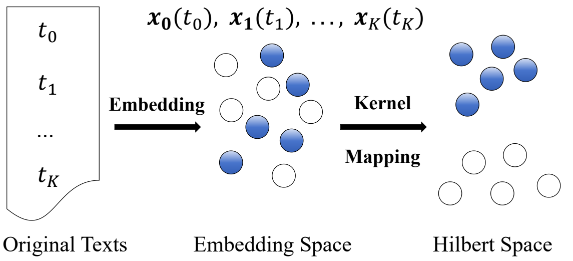


Figure 3: Kernel mapping enhances class separability. Circles in two different colors represent samples from distinct classes.

The $S$ examples most similar to $t_0$ are selected as demonstration examples. These examples, along with their corresponding labels, denoted as $\{(t_1, y_1), \ldots, (t_S, y_S)\}$, are then concatenated with the classification rationale $\mathcal{R}$ to form a *prompt* (as shown in Figure 4) for the LLM. This process is defined as follows:

$$\hat{y_0} = \text{LLM}(t_0 \oplus \mathcal{R} \oplus (t_1, y_1) \oplus \cdots \oplus (t_S, y_S)),$$

where $\hat{y_0}$ is the predicted label for $t_0$, and $\oplus$ represents the concatenation operation.

Analyze the sentiment expressed in the given **Query Text** toward the specified target {*target*}. The sentiment label must be selected from the following set: {*str(label_list)*}. Refer to the provided label descriptions and example demonstrations to guide your classification.

**Label Descriptions:** {Rationales $\mathcal{R}$}
**Demonstrations:** $\{(t_1, y_1), \ldots, (t_S, y_S)\}$

**Query Text:** {*query_text*}

Figure 4: Prompt template of ICL for MCSA.

## 4 Experimental Setup

### 4.1 Experimental Datasets

To comprehensively evaluate RD-MCSA, five datasets spanning different domains and sentiment classification granularities were employed. The characteristics of these datasets are summarized in Table 1.

Table 1: Summary of experimental datasets.

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

These datasets encompass a range of sentiment classification tasks, ranging from sentence-level analysis to topic-based sentiment analysis, thus enabling a comprehensive evaluation of RD-MCSA.

### 4.2 Comparison Models

**Baseline models** were drawn from two categories: (1) classic machine learning algorithms and (2) pretrained language models for sentiment classification. Specifically: 1) **Naïve Bayes** (Rennie, 2001): Multinomial Naïve Bayes with TF-IDF features, incorporating class weighting to mitigate imbalance. 2) **SVM** (Li et al., 2011): Support Vector Classifier with a linear kernel, balanced class weights, and TF-IDF features. 3) **BERT** (Sun et al., 2019): BERT-base model fine-tuned with Focal Loss to alleviate class imbalance. 4) **BERTweet** (Nguyen et al., 2020): Pretrained model for English tweets, optimized with Focal Loss to address imbalance.

Given the recent success of ICL approaches in text classification, several ICL-based selection strategies were included as **comparison methods**: 1) **Random**: In-context examples were selected randomly from the candidate set. 2) **Coreset** (Indyk et al., 2014): Representative samples were selected to capture dataset diversity, using a distance-based probabilistic sampling strategy in the embedding space. 3) **Cos-Similarity** (de Vos et al., 2022): The top-$S$ examples with the highest cosine similarity to the query were selected. 4) **BM25** (Robertson et al., 2009): The top-$S$ examples were selected based on BM25 relevance scores. 5) **Complex-CoT** (Fu et al., 2022): Examples were selected according to complexity, measured by the number of newline characters. 6) **Auto-CoT** (Zhang et al., 2022): Candidate examples were clustered using K-Means ($K = 10$), and the examples closest to each cluster centroid were selected. 7) **Se**$^{2}$ (Liu et al., 2024a): A sequential example selection method that leverages LLM feedback and beam search to construct informative example sequences.

### 4.3 Experimental Implementation Details

In the experiments, dataset division was carried out as follows. Each dataset was split into 80% training and 20% test sets using the `sklearn.model_selection` method.

**1) Baseline models.** All baseline models were trained on the complete training set and evaluated on the full test set.

**2) ICL-based methods (including seven comparison methods and RD-MCSA).**

- From the training set, 1,000 labeled examples were sampled using the Balanced Coreset algorithm, forming the annotated dataset $\mathcal{D}$. This dataset was used both as (i) the demonstration pool for all ICL methods and (ii) the training data for the MK-GP model.

  MK-GP was trained exclusively on these 1,000 samples to ensure a fair comparison with ICL methods, which were also restricted to the same pool of labeled data. This design additionally provided insight into the amount of labeled data required for MCSA tasks, assisting in determining the annotation budget necessary to outperform traditional classifiers trained on large-scale datasets.

- From the test set, 1,000 instances were randomly selected using Python's `random` library. For the ABSIA dataset, where the test set is smaller, all available samples were used. These subsets were employed for ICL performance evaluation. Subset-based evaluation provided a balance between estimation reliability and computational efficiency.

- In RD-MCSA, the Balanced Coreset size for rationale generation was set to $\lambda_{\mathcal{B}} = 100$.

- The number of ICL demonstrations was fixed at $S = 10$, reflecting a balance between efficiency and effectiveness.

Experiments were conducted using three groups of LLMs: GPT[6], DeepSeek[7], and ERNIE[8]. For each group, the more capable (and expensive) model (GPT-4o, DeepSeek-R1, and ERNIE X1 Turbo) was employed for classification rationale

---

[1] https://huggingface.co/datasets/SetFit/sst5

[2] https://huggingface.co/datasets/midas/semeval2017

[3] https://www.iitp.ac.in/ãi-nlp-ml/resources.html#ABSIA

[4] https://snap.stanford.edu/data/web-Amazon-links.html

[5] https://cseweb.ucsd.edu/˜jmcauley/datasets/amazon_v2

Table 2: Experimental results of baseline methods and ICL approaches across five datasets. Values are shown as percentage numerators, with the best method in each category highlighted in bold.

| | **Method** | **SST5** | | **SemEval17** | | **ABSIA** | | **PR_Baby** | | **PR_Software** | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Acc. | F1 | Acc. | F1 | Acc. | F1 | Acc. | F1 | Acc. | F1 |
| Baseline Models | Naïve Bayes | 37.2 | 37.0 | 44.9 | 44.0 | 34.8 | 31.0 | 47.9 | 47.0 | 44.8 | 45.0 |
| | SVM | 37.1 | 37.0 | 56.7 | 58.0 | 49.9 | 50.0 | 51.0 | 51.0 | 58.1 | 59.0 |
| | BERT | **49.9** | **50.0** | 59.2 | 61.0 | 51.2 | **52.0** | **58.2** | **58.0** | **60.3** | **61.0** |
| | BERTweet | 48.7 | 47.0 | **63.4** | **65.0** | **52.4** | **52.0** | 57.7 | 56.0 | 59.9 | 58.0 |
| **ICL** based on GPT-4o +GPT-4o-mini | Random | 55.0 | 54.9 | 57.7 | 60.2 | 51.6 | 52.9 | 57.9 | 57.9 | 62.3 | 63.6 |
| | Coreset | 55.7 | 55.4 | 59.4 | 62.1 | 53.2 | 55.4 | 58.1 | 58.1 | 62.6 | 63.7 |
| | Cos-Similarity | 55.6 | 55.1 | 60.1 | 61.9 | 52.8 | 53.6 | 58.9 | 59.0 | 64.7 | 65.9 |
| | BM25 | 56.5 | 56.0 | 61.6 | 63.5 | 53.0 | 54.7 | 59.2 | 59.4 | 63.1 | 64.3 |
| | Complex-CoT | 56.5 | 54.3 | 62.5 | 63.1 | 52.9 | 55.3 | 58.4 | 58.5 | 65.3 | 66.4 |
| | Auto-CoT | 56.6 | 54.2 | 62.2 | 63.1 | 53.4 | 55.6 | 58.8 | 59.1 | 62.7 | 64.1 |
| | Se$^2$ | 55.9 | 54.2 | 61.9 | 63.3 | 53.1 | 55.9 | 58.2 | 58.6 | 64.4 | 64.0 |
| | **RD-MCSA** | **57.6** | **56.0** | **63.9** | **64.7** | **54.3** | **56.0** | **60.1** | **60.3** | **67.0** | **67.2** |
| **ICL** based on DeepSeek-R1 +DeepSeek-V3 | Random | 56.1 | 55.2 | 67.2 | 67.7 | 51.2 | 53.3 | 56.0 | 56.1 | 61.5 | 62.9 |
| | Coreset | 56.2 | 55.1 | 67.6 | 68.4 | 52.7 | 54.0 | 56.3 | 56.4 | 63.5 | 64.5 |
| | Cos-Similarity | 56.3 | 55.2 | 68.4 | **68.6** | 53.2 | 55.4 | 56.6 | 56.7 | 64.5 | 65.9 |
| | BM25 | 56.6 | 55.8 | 67.3 | 68.0 | 53.1 | 54.7 | 56.6 | 56.7 | 63.9 | 65.1 |
| | Complex-CoT | 56.1 | 53.8 | 67.5 | 67.3 | 52.2 | 53.4 | 56.4 | 56.6 | 65.7 | 65.3 |
| | Auto-CoT | 56.3 | 54.6 | 67.7 | 68.1 | 52.7 | 55.0 | 56.5 | 56.6 | 63.2 | 64.5 |
| | Se$^2$ | 56.3 | 54.8 | 67.9 | 68.2 | 53.1 | 55.2 | 56.5 | 56.1 | 66.2 | 66.2 |
| | **RD-MCSA** | **57.9** | **57.0** | **68.6** | **68.6** | **54.6** | **56.5** | **57.5** | **57.7** | **67.7** | **68.1** |
| **ICL** based on ERNIE X1 Turbo +ERNIE 4.5 Turbo | Random | 51.3 | 48.8 | 67.2 | 66.9 | 50.5 | 50.6 | 55.8 | 55.1 | 62.7 | 62.3 |
| | Coreset | 53.3 | 52.2 | 67.4 | 67.0 | 51.2 | 52.2 | 56.0 | 56.5 | 64.1 | 64.1 |
| | Cos-Similarity | 55.1 | 53.3 | 67.5 | 67.0 | 52.9 | 52.2 | 56.6 | 56.7 | 64.6 | 65.1 |
| | BM25 | 54.7 | 53.2 | 67.7 | 67.1 | 52.8 | 52.5 | 56.9 | 56.2 | 64.7 | 64.8 |
| | Complex-CoT | 56.1 | 53.7 | 67.9 | 67.3 | 52.1 | 52.4 | 56.2 | 56.3 | 65.5 | 65.2 |
| | Auto-CoT | 52.2 | 51.5 | 67.6 | 67.2 | 52.7 | 52.5 | 56.7 | 56.5 | 66.0 | 66.2 |
| | Se$^2$ | 55.2 | 54.3 | 68.2 | 67.3 | 52.1 | 52.0 | 56.8 | 56.2 | 65.7 | 65.8 |
| | **RD-MCSA** | **57.1** | **56.0** | **69.1** | **68.3** | **53.4** | **53.5** | **57.8** | **56.9** | **66.5** | **67.5** |

generation, whereas the more cost-efficient variant (GPT-4o-mini, DeepSeek-V3, and ERNIE 4.5 Turbo) was utilized for ICL in MCSA tasks.

The following settings were applied uniformly across all datasets: $N = 9$ and $M = 9$ were used in the MK-GP model (Equation (3)). The Adam optimizer was adopted with a learning rate of 0.01 over 500 training epochs, and all other optimizer parameters were set to their default values. Optimal hyperparameters were selected via grid search and cross-validation.

Most experiments were conducted on an NVIDIA GeForce RTX 3080 GPU. On average, a single unit of this GPU required 170.86 seconds to complete 500 epochs of Gaussian process training across various datasets. For API-based models, remote inference was employed instead.

### 4.4 Evaluation Metric

Due to the multi-class nature of MCSA and the class imbalance in the experimental data, Accuracy and weighted-average F1 score were used to evaluate performance (Sokolova and Lapalme, 2009).

## 5 Experimental Results and Analysis

### 5.1 Main Results

Table 2 summarizes the performance of various methods on five datasets. The following observations can be made:

**1) Effectiveness of ICL.** ICL achieved the highest accuracy and weighted F1 scores across all datasets, outperforming both traditional machine learning models and language model classifiers. Notably, ICL required only 1,000 labeled examples—substantially fewer than those needed by baseline methods—demonstrating both superior efficiency and effectiveness.

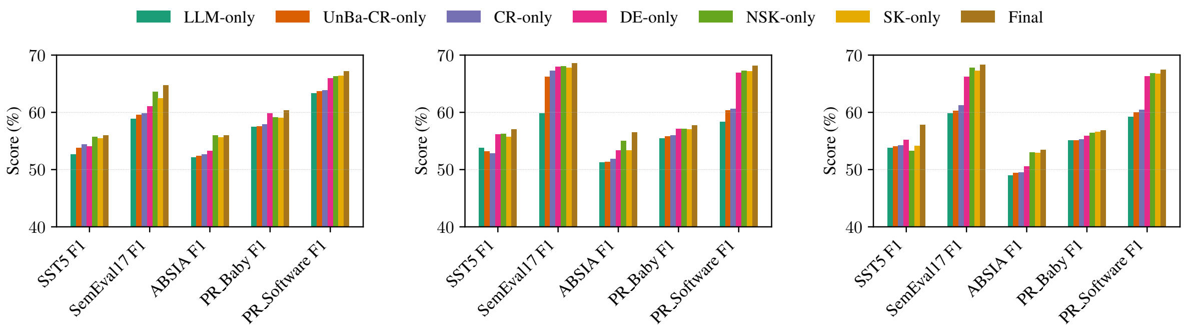
**2) Effectiveness of RD-MCSA.** RD-MCSA consistently outperformed comparision methods across

---

[6] https://openai.com/api/

[7] https://www.deepseek.com/

[8] https://yiyan.baidu.com/



(a) GPT-based (b) DeepSeek-based (c) ERNIE-based

Figure 5: Experimental results from ablation studies across all datasets demonstrate that the removal of any component from the RD-MCSA algorithm leads to a measurable decline in performance.

most datasets. These results highlight the robustness and effectiveness of RD-MCSA, further supported by ablation studies. Moreover, the analysis in Appendix B.1 confirms that the performance gains are statistically significant.

**3) Comparison of Demonstration Selection Methods.** Structured demonstration selection strategies, such as Coreset, Auto-CoT, and similarity-based approaches (including BM25, Cosine, and RD-MCSA), consistently outperformed random sampling. Among these, RD-MCSA proved most effective in identifying informative examples for ICL, thereby delivering the strongest overall performance.

## 5.2 Ablation Analysis

For further analysis, **ablation studies** were conducted with the following model variants: 1) **LLM-only**: Relied solely on the LLM's inherent reasoning for classification, without classification rationales or demonstration examples. 2) **UnBa-CR-only**: Used only classification rationales without considering class balance during generation, excluding demonstration examples. 3) **CR-only**: Used only classification rationales in the prompt, excluding demonstration examples. 4) **DE-only**: Used only demonstration examples, excluding classification rationales. 5) **NSK-only**: Employed only non-stationary kernel functions in the MK-GP algorithm. 6) **SK-only**: Employed only stationary kernel functions in the MK-GP algorithm.

Figure 5 shows the ablation study results (F1 scores) on five datasets, from which the following conclusions can be drawn:

**1) Effectiveness of Rationales:** Incorporating classification rationales led to improved performance compared to direct classification. Rationales enhanced the LLM's ability to interpret label meanings, thereby improving classification accuracy.

**2) Effectiveness of Demonstrations:** Including demonstration examples significantly boosted performance compared to direct classification. These demonstrations served as concrete references that guided the LLM's decision-making process.

**3) Impact of Label Imbalance in Rationale Generation:** Generating classification rationales from imbalanced training samples resulted in noticeable performance degradation. The scarcity of examples from minority classes impaired the LLM's ability to generalize and reduced the quality of the generated rationales.

**4) Effectiveness of Combined Stationary and Non-Stationary Kernels:** Combining stationary and non-stationary kernels outperformed either type alone. This combination more effectively captured structural complexity and enabled the selection of more similar examples in ICL, thereby improving classification accuracy.

## 5.3 Time Cost Analysis

The computational overhead of RD-MCSA comprises two main components: (1) the *offline stage*, which involves Coreset pool construction, rationale generation, and MK-GP training; and (2) the *ICL inference stage*. Statistical analysis based on Table 3, which reports the per-sample average inference time of various ICL methods across five datasets, indicates that there is no statistically significant difference in inference-time cost among the evaluated algorithms. Detailed results are presented in Appendix B.2. Therefore, the additional computational overhead introduced by RD-MCSA is limited to the offline preprocessing stage.

Table 3: Per-sample average inference time (in seconds) of various ICL methods on five datasets.

| **Backbone** | **Method** | **SST5** | **SemEval17** | **ABSIA** | **PR_Baby** | **PR_Software** |
|---|---|---|---|---|---|---|
| **ICL** based on GPT-4o +GPT-4o-mini | Random | 8.72 | 9.05 | 7.58 | 9.01 | 9.79 |
| | Coreset | 8.83 | 8.93 | 7.62 | 9.22 | 9.13 |
| | Cos-Similarity | 8.86 | 9.15 | 7.72 | 7.95 | 9.71 |
| | BM25 | 9.02 | 9.22 | 7.81 | 9.31 | 7.72 |
| | Complex-CoT | 8.74 | 9.13 | 7.25 | 8.83 | 9.18 |
| | Auto-CoT | 8.81 | 9.21 | 7.43 | 7.76 | 10.12 |
| | $\text{Se}^2$ | 8.62 | 9.61 | 8.53 | 8.92 | 9.27 |
| | RD-MCSA | 8.91 | 9.17 | 7.73 | 8.97 | 9.54 |
| **ICL** based on DeepSeek-R1 +DeepSeek-V3 | Random | 12.90 | 12.61 | 7.78 | 11.36 | 12.19 |
| | Coreset | 13.14 | 13.34 | 8.42 | 12.31 | 12.88 |
| | Cos-Similarity | 13.50 | 13.20 | 8.23 | 13.31 | 14.12 |
| | BM25 | 13.21 | 13.64 | 8.61 | 13.48 | 13.79 |
| | Complex-CoT | 12.97 | 12.81 | 8.25 | 11.21 | 13.11 |
| | Auto-CoT | 13.12 | 12.78 | 8.11 | 11.17 | 12.99 |
| | $\text{Se}^2$ | 11.96 | 14.21 | 7.95 | 11.59 | 13.25 |
| | RD-MCSA | 13.17 | 13.82 | 8.57 | 12.21 | 12.92 |
| **ICL** based on ERNIE X1 Turbo +ERNIE 4.5 Turbo | Random | 10.98 | 11.01 | 7.66 | 10.59 | 11.98 |
| | Coreset | 11.07 | 11.21 | 7.79 | 11.32 | 12.17 |
| | Cos-Similarity | 11.82 | 11.17 | 8.21 | 10.27 | 11.82 |
| | BM25 | 11.23 | 11.61 | 8.33 | 12.21 | 12.55 |
| | Complex-CoT | 11.19 | 11.32 | 8.91 | 11.64 | 12.88 |
| | Auto-CoT | 11.11 | 11.49 | 7.98 | 11.71 | 11.76 |
| | $\text{Se}^2$ | 11.24 | 12.89 | 7.63 | 11.92 | 12.21 |
| | RD-MCSA | 11.36 | 11.44 | 8.11 | 12.55 | 11.72 |

## 6 Conclusions

This paper presents a novel framework for multi-class sentiment analysis that leverages in-context learning by integrating classification rationale generation based on balanced Coreset sampling and demonstration selection using multi-kernel Gaussian processes. The proposed approach effectively addresses key challenges such as class imbalance and the high cost of large-scale annotation, while also capturing subtle and nuanced sentiment expressions. Extensive experiments across five diverse datasets demonstrate the superior performance, robustness, and generalizability of the method.

Future research directions include extending the framework to other sentiment analysis tasks, incorporating multimodal data, improving computational efficiency, and designing strategies to mitigate the effects of subjectivity in annotation. These advancements are expected to further contribute to the development of more accurate, efficient, and scalable sentiment analysis systems.

## Acknowledgements

Yaqing Wang is sponsored by Beijing Nova Program.

## Limitations

This paper has several limitations:

**1) Limited Applicability.** Although validated on five diverse datasets, the method has not been evaluated on multimodal datasets, which are increasingly relevant in real-world applications.

**2) Moderate Performance.** While promising, the overall performance remains limited. Even supervised models trained on tens of thousands of samples often struggle to exceed 80%. A key challenge in MCSA tasks lies in annotation subjectivity—different annotators may assign different labels to the same sample. In addition, benchmark dataset quality varies, and this factor has not yet been systematically examined.

**3) High Computational Cost.** Although the MK-GP approach achieves strong results, it is computationally more demanding than some similarity-based methods, particularly in the offline stage. Enhancing computational efficiency remains an important direction for future work.

## 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.

With respect to the use of LLMs, beyond their role in experimental evaluation, LLMs (primarily GPT) were used solely as writing aids for grammar correction and word refinement. They were not employed to generate research ideas, data, images, or tables.

## References

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

S Angel Deborah, TT Mirnalinee, and S Milton Rajendram. 2021. Emotion analysis on text using multiple kernel gaussian... *Neural Processing Letters*, 53(2):1187–1203.

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.

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.

Vincent Cohen-Addad, David Saulpic, and Chris Schwiegelshohn. 2021. A new coreset framework for clustering. In *Annual ACM SIGACT Symposium on Theory of Computing*, pages 169–182.

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 *International Workshop on Semantic Evaluation*, 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 *The Web Conference*, 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*.

Abdullah Elen, Selçuk Baş, and Cemil Közkurt. 2022. An adaptive gaussian kernel for support vector machine. *Arabian Journal for Science and Engineering*, 47(8):10579–10588.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022. Complexity-based prompting for multi-step reasoning. In *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 *International Conference on Power and Energy Technology*, pages 987–990. IEEE.

Piotr Indyk, Sepideh Mahabadi, Mohammad Mahdian, and Vahab S Mirrokni. 2014. Composable core-sets for diversity and coverage maximization. In *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 *SIGHAN Workshop on Chinese Language Processing*, pages 165–174.

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.

Haoyu Liu, Jianfeng Liu, Shaohan Huang, Yuefeng Zhan, Hao Sun, Weiwei Deng, Furu Wei, and Qi Zhang. 2024a. $se^2$: Sequential example selection for in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5262–5284, Bangkok, Thailand. Association for Computational Linguistics.

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. 2024b. 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.

Katerina Margatina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu. 2023. Active learning principles for in-context learning with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5011–5034, Singapore. Association for Computational Linguistics.

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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. 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*.

Angel Deborah S, S Milton Rajendram, and T T Mirnalinee. 2017a. SSN_MLRG1 at SemEval-2017 task 4: Sentiment analysis in Twitter using multi-kernel Gaussian process classifier. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 709–712, Vancouver, Canada. Association for Computational Linguistics.

Angel Deborah S, S Milton Rajendram, and T T Mirnalinee. 2017b. SSN_MLRG1 at SemEval-2017 task 5: Fine-grained sentiment analysis using multiple kernel Gaussian process regression model. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 823–826, Vancouver, Canada. Association for Computational Linguistics.

Ozan Sener and Silvio Savarese. 2017. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*.

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.

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 *China National Conference on Chinese Computational Linguistics*, 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*.

Jun Wang. 2025. A tutorial on LLM reasoning: Relevant methods behind ChatGPT o1. *Preprint*, arXiv:2502.10867.

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.

Shiguang Wu, Yaqing Wang, and Quanming Yao. 2025. Why in-context learning models are good few-shot learners? In *The Thirteenth International Conference on Learning Representations*.

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.

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

# A Mathematical Properties and Applications of Kernel Functions

## A.1 Properties of Kernel Functions

The polynomial kernel is expressed as:

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

where $\gamma_m$ 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 $\boldsymbol{x_j}$.

The Matérn kernel $k_{\text{Matérn}}(x_i, x_j)$ is defined as follows, where $\nu$ and $\ell$ are the kernel parameters:

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

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:

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

The Matérn kernel approaches the RBF and Laplace kernels under different parameter conditions. 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:

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

## A.2 MK-GP Model Training

Given the training inputs $\boldsymbol{X}$, the latent function values for class $c \in \{1, \ldots, u\}$ are

$$\boldsymbol{f}_c = \left[ f_c(\boldsymbol{x}_1), \ldots, f_c(\boldsymbol{x}_K) \right]^\top,$$

which follow a multivariate Gaussian distribution:

$$p(\boldsymbol{f}_c \mid \boldsymbol{X}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_{\boldsymbol{XX}}), \qquad (6)$$

where $\boldsymbol{K}_{\boldsymbol{XX}} \in \mathbb{R}^{K\times K}$ is the kernel (Gram) matrix with entries

$$\left[\boldsymbol{K}_{\boldsymbol{XX}}\right]_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j), \qquad (7)$$

and $k(\cdot,\cdot)$ is the multi-kernel defined in (3). Assuming independence across classes, the joint prior factorizes as

$$p(\boldsymbol{f} \mid \boldsymbol{X}) = \prod_{c=1}^{u} \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_{\boldsymbol{XX}}). \qquad (8)$$

The likelihood term $p(\boldsymbol{y} \mid \boldsymbol{f})$ links the latent functions to the observed labels. Each latent function is transformed into a probability through a logistic (sigmoid) link and combined with a Bernoulli likelihood. For input $\boldsymbol{x}_i$ and class $c$, the conditional distribution is

$$p\Big(y_i^{(c)} \mid f_c(\boldsymbol{x}_i)\Big) = \mathrm{Ber}\Big(y_i^{(c)};\, \sigma(f_c(\boldsymbol{x}_i))\Big), \qquad (9)$$

with

$$\sigma(t) = \frac{1}{1 + e^{-t}}. \tag{10}$$

Assuming conditional independence across data points and classes, the likelihood factorizes as

$$p(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_{c=1}^{u} \prod_{i=1}^{K} p\Big(y_i^{(c)} \mid f_c(\boldsymbol{x}_i)\Big). \tag{11}$$

An MK-GP model $\mathcal{G}$ is trained by minimizing the **negative log-marginal likelihood**:

$$\mathcal{L} = -\log \int p(\boldsymbol{y} \mid \boldsymbol{f})\, p(\boldsymbol{f} \mid \boldsymbol{X})\, d\boldsymbol{f}, \tag{12}$$

where $p(\boldsymbol{f} \mid \boldsymbol{X})$ and $p(\boldsymbol{y} \mid \boldsymbol{f})$ are defined in Equations (8) and (11), respectively.

### A.3 Similarity Evaluation Based on Kernel Functions of MK-GP

By 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 $k(\mathbf{x}_i, \mathbf{x}_j)$ equals the inner product in $\mathcal{H}$:

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

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:

$$\begin{aligned}
&\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 \\
&= \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_i) \rangle_{\mathcal{H}} - 2\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}} \\
&\quad + \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}} \\
&= k(\mathbf{x}_i, \mathbf{x}_i) - 2k(\mathbf{x}_i, \mathbf{x}_j) + k(\mathbf{x}_j, \mathbf{x}_j).
\end{aligned}$$

After normalizing the samples, the first and third terms become constants for the kernel used. Hence, a larger middle term $k(\mathbf{x}_i, \mathbf{x}_j)$ implies a smaller distance between $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$, indicating greater sample similarity.

## B Statistical Analysis of the Results

### B.1 Accuracy and F1 Analysis Based on Mixed-Effects Modeling

The superiority of **RD-MCSA** over competing ICL selection methods was assessed using a linear mixed-effects model (LMM). Algorithm ($A = 8$), Backbone ($B = 3$), and Dataset ($C = 5$) were considered. For each triple $(i, j, k)$ with $i \in \{1, \dots, 8\}$, $j \in \{1, \dots, 3\}$, and $k \in \{1, \dots, 5\}$, a score $Y_{ijk}$ was recorded. The LMM was specified as

$$Y_{ijk} = \mu + \alpha_i + u_j + v_k + \varepsilon_{ijk},$$

$$u_j \sim \mathcal{N}(0, \sigma_u^2),\ v_k \sim \mathcal{N}(0, \sigma_v^2),\ \varepsilon_{ijk} \sim \mathcal{N}(0, \sigma^2),$$

where $\alpha_i$ denoted the fixed effect of the $i$-th algorithm, and $u_j$ and $v_k$ represented random effects induced by backbone and dataset. The hypothesis testing problem was formulated as:

$$H_0 : \alpha_{\text{RD}} = \frac{1}{A-1} \sum_{i \neq \text{RD}} \alpha_i$$

$$\text{vs.} \quad H_1 : \alpha_{\text{RD}} > \frac{1}{A-1} \sum_{i \neq \text{RD}} \alpha_i,$$

where $H_1$ indicates that RD-MCSA achieves a higher mean performance than the average of the other algorithms. The Wald-$t$ test results are reported in Table 4.

Table 4: Mixed-effects contrast: RD-MCSA vs. mean of the other ICL methods.

| Metric | $t$ | $p$ |
|---|---|---|
| Accuracy | 9.355 | $< 10^{-18}$ |
| F1 | 8.223 | $9.92 \times 10^{-17}$ |

At the significance level $\alpha = 0.05$, both $p$-values were far below $\alpha$. Therefore, the null hypothesis was rejected, indicating that the improvements of **RD-MCSA** significantly outperformed the other ICL methods in Accuracy and F1.

### B.2 Time Cost Analysis Based on Variance Analysis

Per-sample inference time was analyzed using a three–main-effects ANOVA (without interactions) with Algorithm ($a = 8$), Model ($b = 3$), and Dataset ($c = 5$):

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + \varepsilon_{ijk},\ \varepsilon_{ijk} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2),$$

where $i = 1, \dots, 8$, $j = 1, \dots, 3$, and $k = 1, \dots, 5$ . The hypothesis testing problem was formulated as:

$$H_0 : \ \alpha_1 = \alpha_2 = \cdots = \alpha_8$$

$$\text{vs.} \quad H_1 : \ \exists\, i \neq j \ \text{ such that } \ \alpha_i \neq \alpha_j.$$

The ANOVA results are reported in Table 5.

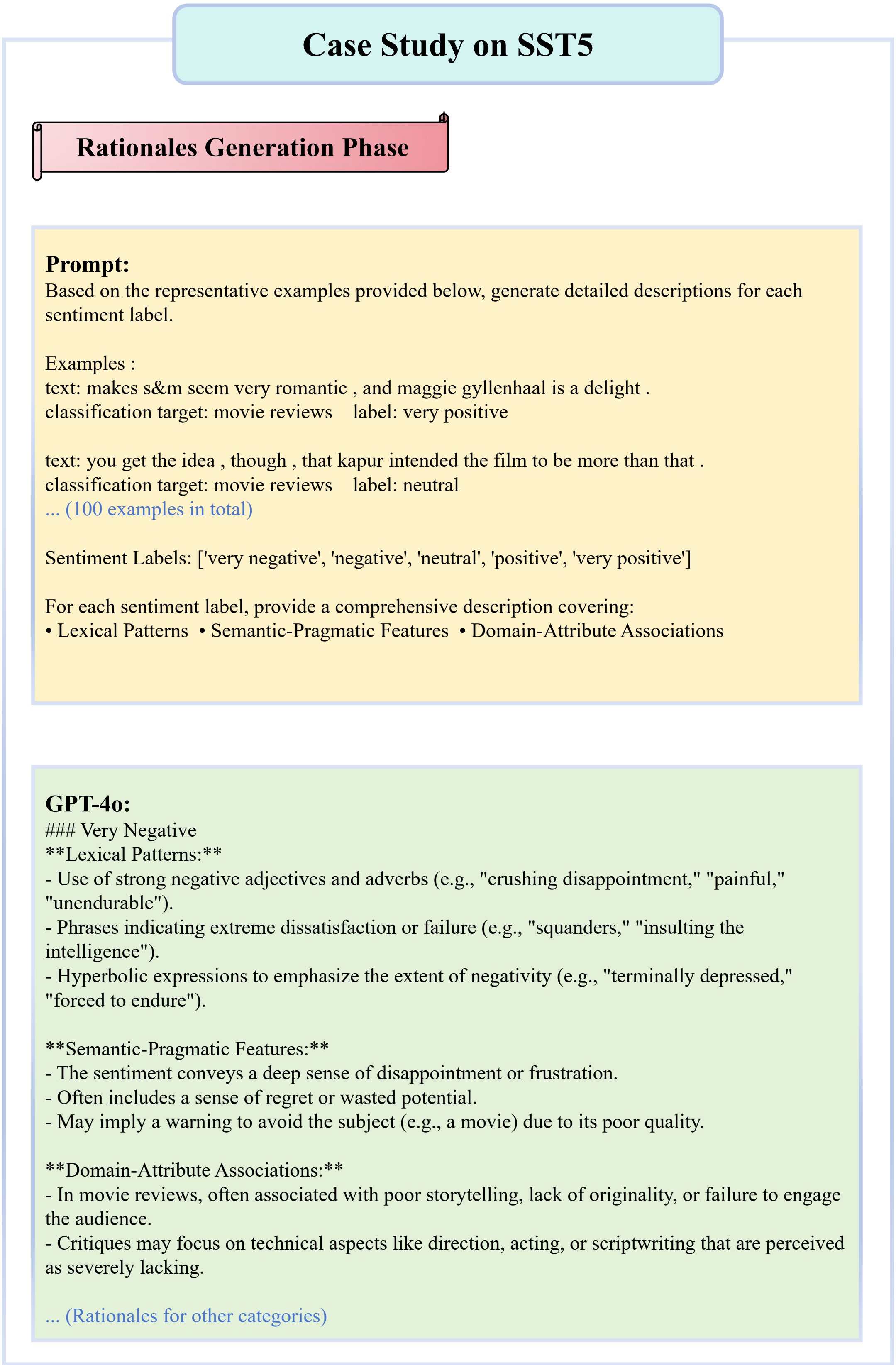
At the significance level $\alpha = 0.05$, the $p$-value was greater than $\alpha$. Therefore, the null hypothesis could not be rejected, indicating that no statistically significant differences in inference time were detected across the eight algorithms.

Table 5: ANOVA of per-sample inference time.

| Factor | SS | df | $F$ | $p$ |
|---|---|---|---|---|
| Algorithm effect | 3.52 | 7 | 0.79 | 0.597 |
| Residual (Error) | 67.44 | 106 | – | – |

## C Case Studies

This appendix presents case studies from the five datasets, each comprising: (i) classification rationale generation and (ii) in-context classification with rationales. The analysis explains why the LLM's predictions were correct or incorrect. Query text, predicted labels, and ground-truth labels are highlighted in red, with elided portions in blue.



## Case Study on SST5

### Rationales Generation Phase

**Prompt:**
Based on the representative examples provided below, generate detailed descriptions for each sentiment label.

Examples :
text: makes s&m seem very romantic , and maggie gyllenhaal is a delight .
classification target: movie reviews    label: very positive

text: you get the idea , though , that kapur intended the film to be more than that .
classification target: movie reviews    label: neutral
... (100 examples in total)

Sentiment Labels: ['very negative', 'negative', 'neutral', 'positive', 'very positive']

For each sentiment label, provide a comprehensive description covering:
• Lexical Patterns  • Semantic-Pragmatic Features  • Domain-Attribute Associations

**GPT-4o:**
### Very Negative
**Lexical Patterns:**
- Use of strong negative adjectives and adverbs (e.g., "crushing disappointment," "painful," "unendurable").
- Phrases indicating extreme dissatisfaction or failure (e.g., "squanders," "insulting the intelligence").
- Hyperbolic expressions to emphasize the extent of negativity (e.g., "terminally depressed," "forced to endure").

**Semantic-Pragmatic Features:**
- The sentiment conveys a deep sense of disappointment or frustration.
- Often includes a sense of regret or wasted potential.
- May imply a warning to avoid the subject (e.g., a movie) due to its poor quality.

**Domain-Attribute Associations:**
- In movie reviews, often associated with poor storytelling, lack of originality, or failure to engage the audience.
- Critiques may focus on technical aspects like direction, acting, or scriptwriting that are perceived as severely lacking.

... (Rationales for other categories)

Figure 6: Case Study on SST5 (Rationales Generation Phase)

## Case Study on SST5

### In-context Learning with Rationale

**Prompt:**
Analyze the sentiment expressed in the given **Query Text** toward the specified target *movie reviews*
The sentiment label must be selected from the following set: ['very negative', 'negative', 'neutral', 'positive', 'very positive'].
Refer to the provided label descriptions and example demonstrations to guide your classification.
**Label Descriptions**:
... (Previously generated Rationales )

**Demonstrations**:
text: uneven performances and a spotty script add up to a biting satire that has no teeth .
classification target: movie reviews    label: negative
... (A total of 10 examples selected by the MK-GP algorithm)

**Query Text**:
this is as lax and limp a comedy as i 've seen in a while , a meander through worn-out material .

**GPT-4o-mini:**
The query text conveys clear disapproval, using phrases like “lax,” “limp,” and “worn-out material” to criticize the film’s lack of energy and originality. The tone is dissatisfied but not extreme, fitting the negative category rather than very negative.

Looking at the example demonstrations, the query text is similar to those labeled as "negative," where the critiques focus on specific elements that detract from the overall experience, such as poor execution or lack of originality.

Label: negative
**label: negative**

**Analysis:**
True label: negative | Predicted: negative

Correctness Category: Combined use of rationales and demonstrations

Reason: The model correctly identified the sentiment as negative. It applied the rationale by noting critical adjectives like “lax” and “limp” along with the phrase “worn-out material”, which signal dissatisfaction but not extreme negativity. It also aligned with demonstration examples labeled negative, where critiques focus on weak execution and lack of originality, leading to the accurate classification.

Figure 7: Case Study on SST5 (In-context Learning with Rationale)

## Case Study on SemEval17

### Rationales Generation Phase

**Prompt:**
Based on the representative examples provided below, generate detailed descriptions for each sentiment label.

Examples :
text: Real Madrid were held to a goalless draw by newly-promoted Sporting Gijon on Saturday. Sportsmail's RIK SHARMA reveals the five things he
classification target: real madrid    label: 0

text: "Carvel Ice Cream is celebrating National Ice Cream Day, Sunday, July 19, with buy one, get one soft serve ones....
classification target: national ice cream day    label: 1

... (100 examples in total)

Sentiment Labels: [-2, -1, 0, 1, 2]

For each sentiment label, provide a comprehensive description covering:
• Lexical Patterns  • Semantic-Pragmatic Features  • Domain-Attribute Associations

**DeepSeek-R1:**
### Sentiment Label Descriptions
### **Sentiment Label Descriptions**
Based on the provided examples and inferred patterns, here are comprehensive descriptions for each sentiment label:
#### **Label: -2 (Extreme Negative)**
**Lexical Patterns**:
- Extreme derogatory language, profanity, slurs, or violent threats (e.g., "terrorist," "disgusting," "death to X").
- Hyperbolic negation (e.g., "never," "worst ever," "utterly vile").
- Explicit calls for harm, exclusion, or annihilation.

**Semantic-Pragmatic Features**:
- Expresses unmitigated hatred, dehumanization, or severe condemnation.
- Often targets groups, ideologies, or individuals with unambiguous malice.
- Contexts include hate speech, extremist rhetoric, or overtly hostile propaganda.

**Domain-Attribute Associations**:
- Topics: Terrorism, genocide, hate crimes, systemic oppression.
- Attributes: Bigotry, extremism, irredeemable harm.

... (Rationales for other categories)

Figure 8: Case Study on SemEval17 (Rationales Generation Phase)

## Case Study on SemEval17

### In-context Learning with Rationale

**Prompt:**
Analyze the sentiment expressed in the given **Query Text** toward the specified target *paul mccartney*
The sentiment label must be selected from the following set: [-2, -1, 0, 1, 2].
Refer to the provided label descriptions and example demonstrations to guide your classification.
**Label Descriptions**:
... (Previously generated Rationales )

**Demonstrations**:
text: Justin Bieber' appeared on Thursday 24 at the 6th place in the Top20 of Norway's Trends: #trndnl
classification target: justin bieber    label: 1
... (A total of 10 examples selected by the MK-GP algorithm)

**Query Text**:
Did you know that 'Hulk Hogan' was Trending Topic on Friday 24 for 23 hours in Portland?#trndnl

**DeepSeek-V3:**
**Reasoning based on label descriptions:**
The text is purely factual, noting Hulk Hogan trended in Portland for 23 hours, with no approval or criticism. The target is only mentioned incidentally, fitting Label 0 (Neutral/Informational).

**Reasoning based on example demonstrations:**
"Ran into wrestling legend Hulk Hogan at the world famous Patsy's Italian Restaurant in NYC Thursday night!" (label: 0) - A neutral, factual mention.

**Label: 0**

**Analysis:**
True label: 0 | Predicted: 0

Correctness Category: Combined use of rationales and demonstrations

Reason: The model correctly classified the query as Label 0 (Neutral/Informational) by applying the rationale that it is a purely factual statement without sentiment. It also aligned with the demonstration example mentioning Hulk Hogan in a neutral, incidental way, reinforcing the accurate prediction.

Figure 9: Case Study on SemEval17 (In-context Learning with Rationale)

## Case Study on ABSIA

### Rationales Generation Phase

**Prompt:**
Based on the representative examples provided below, generate detailed descriptions for each sentiment label.

Examples :
text: if you're not familiar with the cuisine, you may encounter problems ordering here and be sure to bring plenty of cash with you as they don't accept credit cards.
classification target: cuisine    label: -1

text: really tasty spring rolls and noodles for a good price though.
classification target: noodles    label: 1

... (100 examples in total)

Sentiment Labels: [-3, -2, -1, 0, 1, 2, 3]

For each sentiment label, provide a comprehensive description covering:
• Lexical Patterns  • Semantic-Pragmatic Features  • Domain-Attribute Associations

**ERNIE X1 Turbo:**
### Sentiment Label Descriptions
#### **Label -3 (Strongly Negative)**
**Lexical Patterns**:
- Intensifiers: "highly disappointed," "sick of," "terrible," "snotty," "obnoxious."
- Negative verbs: "got sick," "back talk," "burn-outs," "over the top" (context-dependent).
- Critical adjectives: "terrible," "heavy," "dry," "flavorless."

**Semantic-Pragmatic Features**:
- Expresses extreme dissatisfaction, often involving multiple negative aspects (e.g., service + food).
- Emotional intensity: Frustration, anger, or disgust (e.g., "culinary nightmare," "charging such prices for terrible food").
- Finality: Implies irreversible negative judgment (e.g., "never disappointing" in reverse).

**Domain-Attribute Associations**:
- **Food**: Overcooked dishes, dryness, lack of flavor, or mismatched pricing.
- **Service**: Rudeness, incompetence, or dismissive behavior (e.g., "snotty back talk," "slow service").
- **Pricing**: Perceived overcharging relative to quality.

... (Rationales for other categories)

Figure 10: Case Study on ABSIA (Rationales Generation Phase)

## Case Study on ABSIA

### In-context Learning with Rationale

**Prompt:**
Analyze the sentiment expressed in the given **Query Text** toward the specified target *cuisine*.
The sentiment label must be selected from the following set: [-3, -2, -1, 0, 1, 2, 3].
Refer to the provided label descriptions and example demonstrations to guide your classification.
**Label Descriptions**:
... (Previously generated Rationales )

**Demonstrations**:
text: it is what modern cuisine is all about.
classification target: cuisine    label: 1
... (A total of 10 examples selected by the MK-GP algorithm)

**Query Text**:
i love when restaurants think using fancy expensive ingredients makes the food fine cuisine, even with no idea how to use them.

✗

**ERNIE 4.5 Turbo:**
The query text expresses a critical view of restaurants that use expensive ingredients without knowing how to properly incorporate them into their dishes, implying that such practices do not result in "fine cuisine." The use of the word "love" is sarcastic, as it is followed by a negative observation about the lack of culinary skill. This indicates a strongly negative sentiment, as it criticizes the fundamental approach to creating cuisine and suggests that the result is not worthy of being called "fine."

The sentiment aligns with the **Label -3 (Strongly Negative)** description, which includes expressions of extreme dissatisfaction and frustration, particularly regarding the quality of food and the skills involved in its preparation. The text implies a negative judgment about the cuisine's value and quality.

**Analysis:**
True label: 1 | Predicted: -3

Error Category: Polarity Inversion

Error Reason: The model over-interpreted the sentence and falsely detected sarcasm, treating “love” as ironic. This caused a polarity inversion from positive to strongly negative. In context, “love” is used literally to convey mild positive sentiment; the instance should be labeled 1.

Figure 11: Case Study on ABSIA (In-context Learning with Rationale)

## Case Study on PR_Baby

### Rationales Generation Phase

**Prompt:**
Based on the representative examples provided below, generate detailed descriptions for each sentiment label.

Examples :
text: I really love, love, love this pillow. The Jersey cover is extremely soft and makes sleeping with my Snoogle that much more enjoyable. I would have given it 5 stars, but I noticed that the material was pulling away from the zipper and there was a tiny hole starting after only a couple of washings. It wasn't really a big deal though. Nothing that a couple minutes with my sewing machine wasn't able to fix. I would still recommend the product.
classification target: Snoogle Chic Jersey - Snoogle Replacement Cover with Zipper for Easy Use - Heather Gray
label: 4
... (100 examples in total)

Sentiment Labels: [1, 2, 3, 4, 5]

For each sentiment label, provide a comprehensive description covering:
• Lexical Patterns • Semantic-Pragmatic Features • Domain-Attribute Associations

**GPT-4o:**
### Sentiment Label: 1 (Very Negative)

**Lexical Patterns:**
- Use of strong negative adjectives and adverbs (e.g., "terrible," "ludicrous," "broken").
- Frequent use of negations (e.g., "doesn't," "won't," "never").
- Presence of words indicating dissatisfaction or disappointment (e.g., "disappointed," "annoyed," "embarrassed").

**Semantic-Pragmatic Features:**
- The text often conveys a sense of frustration or anger.
- There is a focus on unmet expectations or product failures.
- The language may include hyperbolic expressions to emphasize dissatisfaction.
- The tone is often critical and may include calls for action, such as returning the product or warning others.

**Domain-Attribute Associations:**
- Commonly associated with product defects, misleading advertising, or poor customer service.
- Attributes such as durability, functionality, and value for money are often criticized.
- The sentiment is typically directed towards the overall experience with the product or service.
... (Rationales for other categories)

Figure 12: Case Study on PR_Baby (Rationales Generation Phase)

## Case Study on PR_Baby

### In-context Learning with Rationale

**Prompt:**
... (In this case study, the Query Text is too long, so we omit some content of prompt )

**Demonstrations**:
text: I had purchased this swing a few months and I'm not ALL that pleased with it. It's great for traveling around with. But I felt that the bar that goes across and above the baby's head was in the way...
classification target: Fisher Price - Aquarium Take-along Swing label: 2
... (A total of 10 examples selected by the MK-GP algorithm)

**Query Text**:
I don't think this bouncer is the safest on the market. I wish it held my son more securly there are no sides what so ever on it and the straps are positioned so it doesn't hold him in secure. He likes the seat - although I think he just cares about the vibration - so another seat would be fine for him too! I would purchase a different seat with my next child.

✗

**GPT-4o-mini:**
Reasoning Based on the Descriptions: The review is mainly negative, stressing safety issues like insecure support and lack of sides. Though it notes the child enjoys the vibration feature, the dominant tone is disappointment, fitting Label 2 (Negative) rather than very negative.

Reasoning Based on the Example Demonstrations: It parallels other Negative examples that cite product flaws while acknowledging minor positives, such as the Fisher-Price bouncer case. The sentiment shows clear dissatisfaction but not extreme frustration, supporting Label 2 (Negative).

Based on the analysis of both the descriptions and the example demonstrations, the appropriate sentiment label for the query text is **2 (Negative)**.

**Analysis:**
True label: 3 | Predicted: 2

Error Category: Intensity Misclassification (Negative vs. Neutral)

Reason: The model focused heavily on the reviewer's safety concerns and downplayed the softer, balancing elements (e.g., the child enjoying the vibration feature, acknowledgment that the seat is usable). As a result, it classified the review as Negative (2) rather than Neutral (3). This reflects an error in evaluating sentiment intensity, where moderate dissatisfaction with some positives was mistaken for outright negativity.

Figure 13: Case Study on PR_Baby (In-context Learning with Rationale)

# Case Study on PR_Software

## Rationales Generation Phase

**Prompt:**
Based on the representative examples provided below, generate detailed descriptions for each sentiment label.

Examples :
text: While the software found about ten fixes that I needed, I noticed that my computer did run slower with this software installed. I still had McAfee for the anti-virus and that could contribute to the slow time, but since my laptop is only a few months old, I definitely noticed a slower running and downloading time. It was simple and quick to install, taking less than fifteen minutes. It is recommended that you clear and delete all anti virus software first and load theirs instead of breaking up the features.
classification target: Amazon software product reviews    label: 3.0
... (100 examples in total)

Sentiment Labels: [1.0, 2.0, 3.0, 4.0, 5.0]

For each sentiment label, provide a comprehensive description covering:
• Lexical Patterns  • Semantic-Pragmatic Features  • Domain-Attribute Associations

**DeepSeek-R1:**
### Sentiment Label Descriptions for Amazon Software Product Reviews
**Sentiment Labels:** [1.0, 2.0, 3.0, 4.0, 5.0]
#### **Label 1.0: Strongly Negative**
**Lexical Patterns:**
- **Explicitly negative adjectives/adverbs**: *"dissatisfied," "aggravations," "worthless," "useless," "frustrating," "Never Again!"*
- **Hyperbolic language**: *"worst," "complete waste," "ruined," "unusable."*
- **Direct complaints**: *"spam," "constant need to reenter information," "installation is a pain."*
- **Short, fragmented sentences**: *"Never Again!" "not usable."*

**Semantic-Pragmatic Features:**
- **Extreme dissatisfaction**: Reviews express anger, betrayal, or regret over purchasing the product.
- **Irreparable flaws**: Problems are severe and unresolvable (e.g., crashes, incompatibility, deceptive practices).
- **Warnings to others**: Strong discouragement against purchasing (e.g., *"stay away from this"*).

**Domain-Attribute Associations:**
- **Installation/activation**: Repeated failures, licensing issues (e.g., one-time use restrictions).
- **Performance**: Severe slowdowns, system crashes, incompatibility with hardware/OS.
- **Customer experience**: Deceptive marketing, intrusive spam, lack of support.
... (Rationales for other categories)

Figure 14: Case Study on PR_Software (Rationales Generation Phase)

# Case Study on PR_Software

## In-context Learning with Rationale

**Prompt:**
... (In this case study, the Query Text is too long, so we omit some content of prompt )

**Demonstrations**:
text: If you're looking for video editing software with a lot of options...
classification target: Amazon software product reviews  label: 5.0
... (A total of 10 examples selected by the MK-GP algorithm)

**Query Text**:
After sampling a number of other \"lite\" video and slide-show software suites, I settled on Corel Digital studio. It has all of the features I want:\n-drag and drop functionality\n-clean and intuitive controls that do not require reading a manual to understand\n-slightly sophisticated options that allow you to create polished multimedia projects with minimal effort\n-automatically saves my project every time a change is made\n-does not frequently lock up my computer as other software does(but does still occasionally closes itself down due to \"errors\", but saves your projects so no loss of work/ data)\n\nIs this review helpful? Let me know. Please vote.

✓

**DeepSeek-V3:**
**Comparison to Label Descriptions**: The review is generally positive, highlighting intuitive controls, ease of use, and polished results. Minor flaws (occasional crashes) are noted but softened by auto-save. The overall tone is appreciative and satisfied, with clear preference over other “lite” software.

**Comparison to Demonstrations**: - Similar to the **4.0** example ("PowerDirector 14 Ultra"), where the review is positive but notes minor flaws (e.g., unintuitive navigation, waiting loops). - Not as overwhelmingly positive as the **5.0** examples, which lack any criticism.

**Final Label**: **4.0 (Positive with Minor Flaws)**

**Analysis:**
True label: 4.0 | Predicted: 4.0

Correctness Category: Combined use of rationales and demonstrations

Reason: The model first applied the rationale to detect overall positive sentiment (intuitive controls, polished results, auto-save) alongside a caveat (occasional crashes). It then contrasted these cues with the 5.0 demonstrations, which show unqualified praise and superlatives with no negatives. Because the query contains a concession (“but does still occasionally…”) and lacks hyperbolic language, the model down-weighted from 5.0 to 4.0, yielding the correct classification.

Figure 15: Case Study on PR_Software (In-context Learning with Rationale)