NCL-NLP at SemEval-2025 Task 11: Using Prompting engineering framework and Low Rank Adaptation of Large Language Models for Multi-label Emotion Detection

Kun Lu
Newcastle Upon Tyne, England
yzww1999@gmail.com

Huizhi Liang
Newcastle University
Newcastle Upon Tyne, England
huizhi.liang@ncl.ac.uk

Abstract

SemEval-2025 Task 11 Track A involves identifying all emotions present in a given text segment (Muhammad et al., 2025b). This paper proposes a prompt engineering framework designed to enhance the performance of generative models on multi-label classification. The proposed prompting framework adds elements such as *character definitions, task descriptions, skill requirements, goal settings, constraints, workflows, examples, output format constraints* to the original simple prompt. It integrates structured and context-sensitive prompt templates and instruction fine-tuning strategies of Low Rank Adaptation of Large Language Models to improve classification accuracy. In the evaluation experiments, the proposed approach reached a macro F1 score of 0.849. Our approach demonstrates the effectiveness of prompt-based methods in improving multi-label emotion classification with fine-tuned generative models.

1 Introduction

In this article, we present an approach for addressing the SemEval-2025 Task 11 Track A task which is focusing on perceiving emotion and focusing on determining what emotion most people will think the speaker may be feeling given a sentence or short text snippet uttered by the speaker. Emotions that need to be accurately identified include joy, fear, surprise, sadness, and anger (Muhammad et al., 2025a).

Emotion detection involves identifying and understanding human emotions through various methods including facial analysis, speech analysis, text analysis, and so on. The task focuses on text analysis, specifically sentiment analysis, is a method of emotion detection that uses natural language processing (NLP) to analyse written text and determine the sentiment or emotion expressed. This technique helps gauge public opinion, understand customer feedback, and detect emotions in online communication. Sentiment analysis can determine if a text expresses a positive, negative, or neutral sentiment. It is useful for various applications, including social media monitoring, brand reputation management, and customer service (Burkhardt et al., 2009).

In this task, we use a generative large language model that has been fine-tuned with instructions. LLMs fine-tuned through instructions can provide answers to some extent. However, due to the uncertainty in the generation of language models, the answers often do not correspond to the specified categories of emotions, and the accuracy of the answers is not very high. Therefore, we have designed a prompt engineering framework to help the model generate better responses and ensure that the model only outputs the specified categories of emotions.

2 Related Work

Sentiment Analysis involves various methods, which can be categorized into two primary approaches: machine learning-based, and deep learning-based methods (Zhou, 2024).

Using machine learning to solve the problem of multi-label classification involves data preprocessing, including data collection, data cleaning, label encoding (one-hot encoding), and feature extraction (TF-IDF and Word2Vec); selecting appropriate models, such as decision trees, random forests, support vector machines, or some ensemble models like XGBoost (Sonawane et al., 2018).

In addition, deep learning can also be used to solve multi-label classification problems. Apart from some data preprocessing stages, deep learning can choose neural networks as models, including feed-forward neural networks, convolutional neural networks, recurrent neural networks, and Transformers, among others. For example, adversarial networks and temporal convolution networks can be used for emotion recognition in Man-

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 381–385

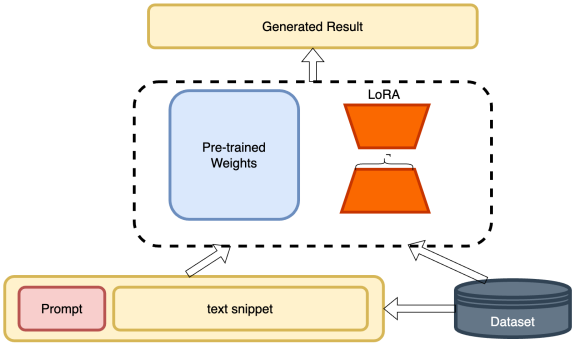

Figure 1: Overview of the system

darin (Li et al., 2023).For transformer architecture, the most representative ones, Bert, GPT and T5 can be used for language understanding tasks and text generation tasks, respectively. BERT can extract the embedding representation of text and then perform multi-text classification through a linear classifier (Yang et al., 2024). On the other hand, a generative model fine-tuned with instructions can complete text classification tasks under specific instructions (Edwards et al., 2021). Due to the large number of parameters in such models and their training on vast amounts of data, they are also referred to as large language models. T5 is a complete Transformer architecture, which has been used in previous research combined with Chain Of Thought for sentiment classification (Rusnachenko and Liang, 2024).

3 System Overview

Our method can improve the performance of multi-label classification tasks by designing an effective prompt engineering framework and using LoRA fine-tuning techniques to constrain the output of generative models, preventing the generation of invalid classification results.

3.1 Low-Rank Adaptation of Large Language Models

Fine-tuning a model in full often requires more resources. Low-Rank Adaptation is a technique that freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of Transformer architecture, greatly reducing the number of trainable parameters for downstream tasks. It means that LoRA can train with a smaller number of parameters to achieve the same effect as fine-tuning (Hu et al., 2022). For simplifying LoRA fine tuning model, we utilize the LLaMA-Factory (Zheng et al., 2024) to fine tune the LLM.

Before fine-tuning, we need to preprocess the original dataset to obtain an object containing the fields of **instruction** and **output**, which will be used for training with LLaMA-Factory.

3.2 Prompt Engineer Framework

For large language models, different prompts can produce different results. Designing a comprehensive and effective prompt framework can help the model better handle the corresponding tasks.

We established a simple baseline to explore the ability of LLMs to perform classification tasks under simple instructions shown as in Table 1. The simple instruction consists of "*Please determine the emotion of text*" and the text snippet from dataset. The output is a string where 0 indicates that the emotion does not exist in the text, and 1 indicates that it does exist, with different types of emotions separated by commas.

Furthermore, we design a prompt engineer framework to improve the output of model. In this framework, there are nine elements including: *role definition, task description, skill requirements, goal setting, constraints, workflow, examples, output format, and startup instructions*, The proposed prompt engineer framework ensures the LLM can understand the task and generate correct outputs. For the format of output, we adopted a simple request to indicate whether emotions exist. Table 2 shows an example of a training sample that includes the key element used for fine tuning.

Table 1: Example of a Training Sample with simple instructions

Text: *"Colorado, middle of nowhere."*

Anger	Fear	Joy	Sadness	Surprise
0	1	0	0	1

Instruction: Please determine the emotion of text. The text is + **text**
output: "anger:0,fear:1,joy:0,sadness:0,surprise:1"

The specific instruction is displayed in the Figure 2.

In this toy example, we defined the interactive roles to provide background information, explaining that the model needs to analyse the dialogue to determine the emotions expressed by the speaker. The specific task description requires that the role analyze based on the text content, without considering factors like voice or facial expressions. The

Table 2: Example of a Training Sample with prompt framework

Text: *"Colorado, middle of nowhere."*

Anger	Fear	Joy	Sadness	Surprise
0	1	0	0	1

Instruction:
- Role: ...
- Background: ...
- Profile: ...
- Skills: ...
- Goals: ...
- Workflow: ...
- Examples: ...
- OutputFormat: ...
- Start: Speaker1: + **text**

output: "anger:0,fear:1,joy:0,sadness:0,surprise:1"

role is only required to have advanced language comprehension, emotion recognition, and dialogue analysis abilities, with the primary goal being to use binary labels to identify the presence or absence of each emotion. The constraints specify that the analysis should be based solely on the text content, serving as a supplement to the task description. In the workflow, it details how the role should complete the task, using a Chain-Of-Thought approach. Figure 2 shows three strong examples and their analysis results to help understand how the role applies the written steps and labels. The output format specifies the format of the output, and finally, a startup instruction is used to initiate the process.

4 Experiment

4.1 Datasets

The dataset used in this paper is a subset of the competition data, focusing only English language. The English dataset consists of training, development and test sets. each entries contains: ID, A piece of text, and emotion labels(anger, fear, joy, sadness, surprise).

The datasets includes three subsets: A training datasets along with 2,768 entries. A development datasets along with 116 samples and A test datasets with 2767 samples for prediction. After data visualization of train datasets, the average character length of the text is around 78. The distribution of labels is as follows: Fear appears most frequently with 1611 samples, Anger has the fewest with 333 samples. Joy, Surprise, and Sadness have 674, 878, and 839 samples respectively. The train datasets reveals an imbalance.Training data imbalance can affect the results of prediction to some extent.

Data preprocessing is an important part of NLP tasks. However, due to the pre-training of large language models on vast and diverse datasets, along with their enormous parameter sizes, data preprocessing such as data correction is often unnecessary. During the entire experiment, we also found that data preprocessing did not have a significant impact on the final results.Therefore, the step of data preprocessing has been discarded. Meanwhile, using a model with larger parameters can also solve the problem of data imbalance.

4.2 Selection Of LLM

In this paper, we apply the Meta Llama 3.1 instruction tuned 8B as generated language model, which supports Engish, German, French, Italian, Portuguese. Llama 3.1 is an auto-regressive language model that uses an optimized transformer architecture. The tuned versions use supervised fine-tuning(SFT) to align with human preferences for helpfulness and safety (Dubey et al., 2024).

4.3 Evaluation Metrics

The official evaluation metrics for Track A is the macro F1 scores which is the unweighted average of the F1 scores across all classes in a classification problem, treating each class equally regardless of its size or frequency. It is used to evaluate model performance when dealing with imbalanced datasets.

$$F1_{\text{macro}} = \frac{\sum F1_i}{N}$$

where N is the number of classes.

4.4 Experiment Setup

Different instructions are used to fine tune the generated language model named Llama3.1 8B. All models were implemented with the PyTorch. The experiments are conducted on a single NVIDIA A40 GPU. The training process utilize LLaMA-Factory to fine tune.

When performing LoRA fine-tuning, some hyperparameters are involved, after comparison, the model is trained for 6 epochs with a batch size of 1, using gradient accumulation over 2 steps. The learning rate is set to 2.0e-4, and a cosine learning rate scheduler is employed with a 0.1 warm-up ratio.

Figure 2: An Example Instruction Prompt

Table 3: The average micro and macro F1 scores in the test set.

Model	Anger	Fear	Joy	Sadness	Surprise	Micro F1	Macro F1
Baseline	0.781	0.865	0.814	0.818	0.822	0.833	0.822
Our system	0.790	0.876	0.832	0.841	0.847	0.854	**0.849**

5 Results and Discussions

In this study, we compare the performance of two models: the baseline model with simple instruction and our model. As shown in Table 3, the our models outperforms the baseline model across all emotion categories. Specifically, the prompt framework achieves higher F1 scores in detecting **Anger, Fear, Joy, Sadness, and Surprise**. The most notable improvements are observed in the **Sadness**(0.0227) and **Surprise**(0.0244) categories, highlighting the framework's ability to better capture subtle emotional cues.

In terms of overall performance, the prompt framework also surpasses the baseline in both **Micro F1** and **Macro F1** scores, with improvements of 0.021 and 0.027, respectively. These results suggest that the our system not only performs better on individual emotion classification tasks but also achieves a more balanced classification across all categories, as indicated by the Macro F1 score.

However, there are still some shortcomings, for example, the experiment was only validated in English and not tested in other languages.

6 Conclusion

This article presents a new prompt framework for the task of text classification, further improved the ability to recognize various emotions based on a high baseline score. By adding elements such as *character definitions, task descriptions, skill requirements, goal settings, constraints, workflows, examples, output format constraints, and startup instructions* to the originally simple prompt, the proposed approach achieved improved results in the competition. However, there is still room for improvement in the anger category.

Furthermore, we can explore the hidden features contained in the data through feature engineering.

For example, the Notre Dame Cathedral is often associated with fires. External tools like Wikidata can be used to determine if there are similar keywords in the text and then add relevant features to the prompt words. At the same time, we can also explore the application of Discriminative LLM (Wu et al., 2024) for classification tasks.

References

Felix Burkhardt, Markus Van Ballegooy, Klaus-Peter Engelbrecht, Tim Polzehl, and Joachim Stegmann. 2009. Emotion detection in dialog systems: Applications, strategies and challenges. *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–6.

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

Aleksandra Edwards, Asahi Ushio, Jose Camacho-Collados, Hélène de Ribaupierre, and Alun Preece. 2021. Guiding generative language models for data augmentation in few-shot text classification. *arXiv preprint arXiv:2111.09064*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net.

HF Li, XY Zhang, SF Duan, HR Jia, and HZ Liang. 2023. Fusing generative adversarial network and temporal convolutional network for mandarin emotion recognition . *Zhejiang Daxue Xuebao (Gongxue Ban)/Journal of Zhejiang University (Engineering Science)*.

Shamsuddeen Hassan Muhammad, Nedjma Ousidhoum, Idris Abdulmumin, Jan Philip Wahle, Terry Ruas, Meriem Beloucif, Christine de Kock, Nirmal Surange, Daniela Teodorescu, Ibrahim Said Ahmad, David Ifeoluwa Adelani, Alham Fikri Aji, Felermino D. M. A. Ali, Ilseyar Alimova, Vladimir Araujo, Nikolay Babakov, Naomi Baes, Ana-Maria Bucur, Andiswa Bukula, Guanqun Cao, Rodrigo Tufino Cardenas, Rendi Chevi, Chiamaka Ijeoma Chukwuneke, Alexandra Ciobotaru, Daryna Dementieva, Murja Sani Gadanya, Robert Geislinger, Bela Gipp, Oumaima Hourrane, Oana Ignat, Falalu Ibrahim Lawan, Rooweither Mabuya, Rahmad Mahendra, Vukosi Marivate, Andrew Piper, Alexander Panchenko, Charles Henrique Porto Ferreira, Vitaly Protasov, Samuel Rutunda, Manish Shrivastava, Aura Cristina Udrea, Lilian Diana Awuor Wanzare, Sophie Wu, Florian Valentin Wunderlich, Hanif Muhammad Zhafran, Tianhui Zhang, Yi Zhou, and Saif M. Mohammad. 2025a. Brighter: Bridging the gap in human-annotated textual emotion recognition datasets for 28 languages. *arXiv preprint arXiv:2502.11926*.

Shamsuddeen Hassan Muhammad, Nedjma Ousidhoum, Idris Abdulmumin, Seid Muhie Yimam, Jan Philip Wahle, Terry Ruas, Meriem Beloucif, Christine De Kock, Tadesse Destaw Belay, Ibrahim Said Ahmad, Nirmal Surange, Daniela Teodorescu, David Ifeoluwa Adelani, Alham Fikri Aji, Felermino Ali, Vladimir Araujo, Abinew Ali Ayele, Oana Ignat, Alexander Panchenko, Yi Zhou, and Saif M. Mohammad. 2025b. SemEval task 11: Bridging the gap in text-based emotion detection. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, Vienna, Austria. Association for Computational Linguistics.

Nicolay Rusnachenko and Huizhi Liang. 2024. nicolay-r at SemEval-2024 task 3: Using flan-t5 for reasoning emotion cause in conversations with chain-of-thought on emotion states. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 22–27, Mexico City, Mexico. Association for Computational Linguistics.

Priyanka Sonawane, M Prof.Pramila, and Chawan. 2018. Multi label document classification approach using machine learning techniques: A survey.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. 2024. A survey on large language models for recommendation. *World Wide Web*, 27(5).

Huiyu Yang, Liting Huang, Tian Li, Nicolay Rusnachenko, and Huizhi Liang. 2024. hyy33 at WASSA 2024 empathy and personality shared task: Using the CombinedLoss and FGM for enhancing BERT-based models in emotion and empathy prediction from conversation turns. In *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 430–434, Bangkok, Thailand. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Mengtao Zhou. 2024. A review: Text sentiment analysis methods. *Journal of Computing and Electronic Information Management*.