Controlling Emotion Intensity and Blending in Text via Task Vector Composition for Dialog System Personalization

Ryota AMANO, Kazuya MERA, Yoshiaki KUROSAWA, Toshiyuki TAKEZAWA
Graduate School of Information Sciences, Hiroshima City University, Japan
dk65030@e.hiroshima-cu.ac.jp, {mera, kurosawa, takezawa}@hiroshima-cu.ac.jp

Abstract

Recent advances in large language models (LLMs) have accelerated the development of dialog systems, with increasing attention paid to personalization. One key challenge is how to flexibly control emotional intensity and blend multiple emotions in generated text—a crucial component for simulating diverse personalities. Traditional approaches often require training separate models for each emotional configuration. In this study, we propose a method that enables fine-grained control over both emotional intensity and blended emotional states by composing emotion-specific task vectors. Each emotion-specific model is fine-tuned from a base model, and the resulting task vectors are combined and applied to a neutral model to synthesize blended emotional behaviors. Experimental results using LLM-based evaluation demonstrate that our method successfully generates text reflecting specified emotional profiles with controllable intensity and combinations.

1 Introduction

Since the release of ChatGPT in November 2022, the rapid development of LLMs has prompted increased interest in deploying dialog systems in society. Dialog systems are already being used in various contexts, including customer support, counseling, and elderly care. More recently, their use has expanded into domains such as AI characters in the metaverse or human digital twins, where dialog systems are expected to respond with distinct personalities.

While personalization typically focuses on attributes such as memory, temperament, or background, the ability to control emotional expression—particularly its intensity and blend—is a critical yet underexplored aspect. Moreover, compared to personality traits, emotions are more readily perceived and evaluated, both by humans and LLM-based automatic judges. Our study focuses on this gap, leveraging emotions as a proxy for lightweight, controllable personalization.

To endow dialog systems with personality traits such as memory, background, and temperament, methods have been proposed that input these traits as text or embedding vectors. However, using text to represent personality poses challenges in fine-grained control and often requires large volumes of text to cover detailed nuances. Alternatively, embedding-based personality representation typically relies on fine-tuning, which necessitates retraining every time a new personality is needed.

Personalized Soups (Jang et al., 2023) addressed this issue by building models aligned with specific response styles from different perspectives and then merging them to simultaneously express multiple aspects of personality. Extending this idea, it becomes possible to express composite personalities without retraining by creating and merging models that embody typical personality traits. However, building such models requires labeled personality datasets and evaluating the resulting output, both of which are difficult.

To address these limitations, the present study focuses on emotions rather than personality. There are two main reasons for this choice. First, emotional expressions in text are generally easier to recognize and evaluate—both by human judges and by automatic evaluators such as LLM-as-a-judge—compared to personality traits, which are abstract and often require long-term behavioral context. Second, large-scale corpora with explicit emotional annotations (e.g., intensity levels) are more widely available than corpora annotated with personality traits, enabling more robust training and evaluation. We propose a method to express emotion intensity and combinations by merging models that each specialize in a specific emotion.

Furthermore, we introduce an evaluation method using LLM-as-a-judge (Zheng et al., 2023) to assess the emotional expression in generated text, which is challenging to evaluate quantitatively.

2 Related work

Recent studies have shown that merging models through linear interpolation or weighted averaging of parameters can modify a model's capabilities. Notably, Ilharco et al. (2023) introduced the concept of a "task vector," derived from the difference in model parameters before and after training, and showed that adding or subtracting these vectors can alter model behavior accordingly. A task vector can be intuitively understood as a direction in parameter space that represents a specific behavioral change, such as learning a new skill or style, and this concept has since become central in model merging research.

Building on this idea, Huang et al. (2024) proposed the "Chat Vector" concept—capturing the difference between a base model and its instruction-tuned variant—and demonstrated how this enables instruction-following behavior to be transferred to language models in other languages without further training. As illustrated in Figure 1, the conventional pipeline for multilingual adaptation typically begins with continual pre-training (CP) of a pre-trained language model (PLM) on a target language corpus. This is followed by supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF), resulting in a target LM, which is instruction-tuned. In contrast, the Chat Vector approach bypasses these stages entirely: it extracts a parameter vector (referred to as the Chat Vector) from a source-language PLM and its chat-tuned variant, then grafts it onto a continually pre-trained PLM (CP Model)—much like donning a suit of conversational "armor"—to instantly impart dialog capabilities.

Jang et al. (2023) trained specialized models based on expertise, information richness, and response style, and proposed personalizing alignment by merging models according to user preferences. Their merging method uses a weighted sum of parameters under the constraint that weights sum to one. However, their work does not discuss how to determine these weights.

Zhou et al. (2024) formulate smooth attribute-intensity control for text generation and propose an automatic evaluation framework that combines GPT-4 pairwise judgments with an Elo-rating aggregation scheme, allowing them to quantify range, calibration and consistency without human annotators.

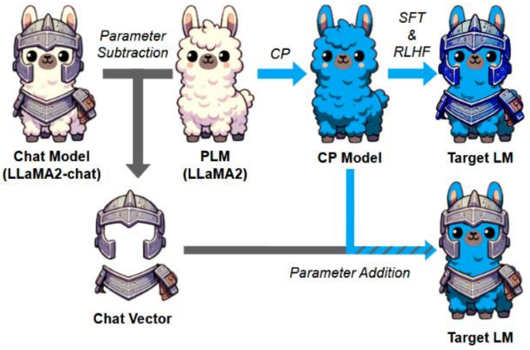

Figure 1: Illustration of the Chat Vector concept (Huang et al., 2024).

Their benchmark spans five single attributes—anger, happiness, formality, understandability, and conciseness—covering sentiment, stylistic, and broader linguistic properties. While the authors demonstrate effective control for each attribute in isolation, they explicitly acknowledge that simultaneous manipulation of multiple attributes, though theoretically desirable, is left for future work.

Two of the evaluated attributes (anger and happiness) are clearly emotional; however, the paper's analysis treats them alongside the other attributes and does not offer an in-depth discussion of emotion-specific challenges (e.g., valence diversity or cross-emotion interference). Consequently, issues unique to fine-grained emotional control remain open questions.

These studies collectively demonstrate the feasibility of modifying model behavior via parameter arithmetic. Building upon these foundations, our study applies task vector-based model merging to emotional expression—a domain that allows clearer evaluation and offers richer annotated data resources.

3 Task Vector Composition for Fine Emotional Control and Blending

To enable flexible control over emotional expression in text generation, we propose a lightweight method based on task vector composition. Our approach enables both intensity scaling and emotion blending by linearly combining parameter differences derived from instruction-tuned models. The overall framework consists of three stages: (1) vector extraction from emotion-specific models, (2) composition of control vectors, and (3) application via parameter addition. We describe each of these components in the following subsections.

3.1 Expressing Compound Emotions via Model Merging

We propose a model merging method based on the linear composition of task vectors to produce compound emotional expressions. As illustrated in Figure 2, Chat Vectors—parameter differences between a neutral emotion model and each emotion-specific model—are combined with scalar weights and added to the neutral model.

In this visualization, Chat Vectors are depicted as armor pieces, where colors symbolize distinct emotional types (e.g., Emotion α and β). When combined with respective weights (w_α, w_β), the resulting armor takes on a blended color, such as purple, reflecting the composite emotion. A stronger weight for Emotion α results in a more reddish purple, visually signifying its dominance.

This metaphor illustrates how weighted blending enables fine-grained emotional control—e.g., emphasizing joy while keeping surprise subtle. Our experiments later confirm that such weighted combinations effectively modulate the emotional tone of generated text.

Assigning a weight of 1.0 to a Chat Vector yields a strongly expressed emotion, while lower weights reduce its influence. Blending with the neutral model allows for mild emotional expression.

To build the models, we fine-tune a base LLM on neutral-emotion data to create a neutral model. Further fine-tuning on emotion-specific corpora produces emotion-specific models, and their differences from the neutral model are used to derive Chat Vectors.

3.2 Dataset

To generate emotionally expressive text, we use the WRIME dataset (Kajiwara et al., 2021), a single-post social networking service (SNS) dataset comprising 43,200 Japanese-language social media posts authored by 80 participants. Although our long-term objective is to apply our approach to dialog systems, ideally using dialog datasets, we focus on WRIME here because it provides a large-scale dataset with reliable emotion annotations. Each post is annotated with two types of emotion ratings: (1) self-reported emotions by the author, and (2) perceived emotions as judged by three independent readers. The annotations are based on Plutchik’s (1980) eight basic emotions—joy, sadness, anticipation, surprise, anger, fear, disgust, and trust—and rated on a 4-point scale: none, weak, medium, and strong. We rely on reader-perceived emotion annotations, because our goal is for users to correctly perceive the system’s expressed emotions rather than for a system to mimic human internal states; this choice is also supported by prior findings that reader labels are more consistent and predictable than author self-reports (Kajiwara et al., 2021). Since each post is rated by three readers, we average the reader scores as the final emotion score for each emotion.

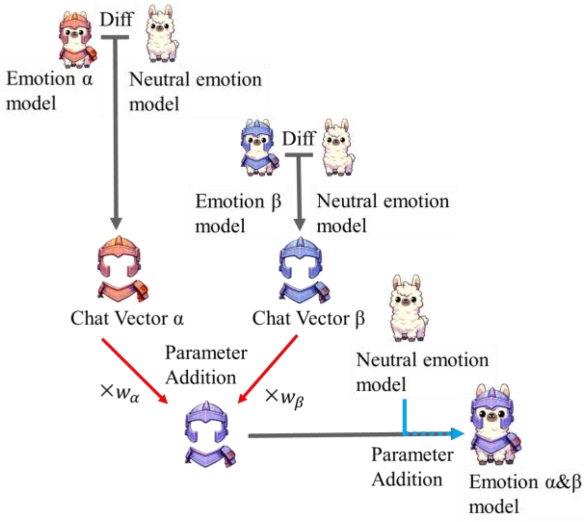

Figure 2: Model of the Proposed method.

Because many posts express multiple emotions, we focus exclusively on posts characterized by a single dominant emotion to train emotion-specific models. We define such “single-emotion posts” using the following criteria:

1. All emotion scores are “none” → **neutral data**
2. Emotion α is rated medium or strong and has the highest among all emotions → **strong data for Emotion** α
3. Emotion α is rated weak and all other emotions are “none” → **weak data for Emotion** α
4. Posts not matching 1–3 are excluded from training

After filtering, the class distribution becomes markedly imbalanced; joy, sadness, surprise, and fear remain sufficiently represented for stable training, whereas anger, disgust, and trust become low-resource. In addition, anticipation is difficult to distinguish from joy because the two occupy a similar region in valence-arousal space, which increases inter-annotator confusion. Accordingly, we restrict our experiments to four emotions—joy, sadness, surprise, and fear—which offer adequate sample sizes and higher annotator consistency, enabling clearer evaluation of intensity control and emotion blending.

3.3 Training Emotion-Specific Models

As our base model, we use llm-jp-3-1.8b-instruct[1], which is an instruction-tuned version of the 1.8-billion-parameter foundation model llm-jp-3-1.8b[2]. We first fine-tune this model using neutral emotion data to construct a neutral emotion model. Based on this, we then fine-tune four emotion-specific models, each corresponding to one of the target emotions.

For training the emotion-specific models, we employ Direct Preference Optimization (DPO) (Rafailov et al., 2023), which fine-tunes the model by contrasting preferred and dispreferred outputs. In our case, we generate preference pairs by matching strong-emotion examples (preferred) with weak-emotion ones (non-preferred) for each target emotion, so that the model learns to favor stronger emotional expression. As a result, the number of training pairs for each emotion is capped at twice the size of the smaller class (weak or strong).

4 Evaluation Experiments

In this section, we verify whether the generated text reflects the specified emotional intensity.

4.1 Experimental Conditions

We use the Transformers library (Wolf et al., 2020), the Transformers Reinforcement Learning (TRL) library [3] for preference-based tuning, and the Parameter-Efficient Fine-Tuning (PEFT) library[4] to apply Low-Rank Adaptation (LoRA) (Hu et al., 2022). LoRA parameters are set as rank=8 and alpha=16. GPT-4o mini is used as the baseline. Few-shot prompts include three randomly selected posts representing different intensity levels from the training set.

	Training Samples	Validation Samples
Neutral	2,401	93
Joy (pairs)	2,240	83
Sadness (pairs)	1,926	54
Surprise (pairs)	1,873	41
Fear (pairs)	1,466	27

Table 1: Number of Text Samples Used.

	Joy	Sadness	Surprise	Fear
Inter-Annotator Agreement (Average)	0.603	0.393	0.427	0.432
Agreement between Estimator and Annotator Average	0.668	0.539	0.456	0.512

Table 2: Agreement Rates in Human and Estimator-Based Emotion Evaluations.

We train the neutral model on 2,401 posts, and four emotion-specific models on a total of 7,505 preference pairs drawn from the WRIME corpus (joy 2,240; sadness 1,926; surprise 1,873; fear 1,466; see Table 1). The neutral emotion model is trained for one epoch, whereas the emotion-specific models are trained for up to four epochs.

For single emotion testing, 10 weight settings ([0.1, 0.2, ..., 1.0]) are tested with 100 generations each. For compound emotion testing, all combinations of 10 intensity levels for two emotions are tested with 100 generations per combination.

As a comparative experiment with the proposed method, we also generate texts using GPT-4o mini (gpt-4o-mini-2024-07-18) in a prompt-based method, where the desired emotion intensity or combination of intensities is explicitly specified as a numerical value in the prompt. The prompt to use GPT-4o mini is in Appendix A.1.

4.2 LLM-Based Emotion Intensity Estimator

We used the Llama-3.1-70B-Japanese-Instruct-2407 model (Ishigami, 2024), a Japanese fine-tuned variant of Llama-3.1-70B-Instruct (Grattafiori et al., 2024), as an LLM-as-a-judge, which estimates the intensity of emotions expressed in the given input text. When a text is provided along with a few-shot prompt (Appendix A.2), the model predicts the intensity of each emotion as one of four levels: none, weak, medium,

[1] https://huggingface.co/llm-jp/llm-jp-3-1.8b-instruct

[2] https://huggingface.co/llm-jp/llm-jp-3-1.8b

[3] https://github.com/huggingface/trl

[4] https://github.com/huggingface/peft

or strong. The LLM-as-a-judge model is quantized to 4-bit precision for computational efficiency.

To evaluate the reliability of the estimated emotion intensities, we compared the model outputs against the average human rating on 1,980 samples from the WRIME dataset, using Quadratic Weighted Kappa (Cohen, 1968) as the evaluation metric. As shown in Table 2, the estimation quality of the model achieves agreement levels comparable to or exceeding those of human annotations.

4.3 Evaluation Results by Emotion Intensity

4.3.1 Expressed Intensity for Single Emotion

In this experiment, we verify whether the generated texts accurately reflect the emotion weights configured during generation, using the emotion estimator described in Section 4.2.

For single-emotion intensity control, we vary the weight of the target emotion in increments of 0.1 and compute the distribution of estimated intensity levels for 100 generated texts at each setting.

Figure 3 shows the proportion of texts generated by the proposed method that are classified into four levels of joy intensity: None, Weak, Medium, or Strong. Figure 4 presents the corresponding results for the prompt-based method.

The results indicate that with the proposed method, increasing the specified emotion weight leads to stronger emotional expression in the generated texts.

In contrast, the prompt-based method exhibits intensity saturation at medium and higher levels.

4.3.2 Compound Emotion Expression

We conducted an experiment to estimate the expressed intensities of compound emotions using the same method as in Section 4.3.1. When combining Emotion α and Emotion β, we fixed the weight of Emotion α at 0.5 and varied the weight of Emotion β. Figure 5 shows the estimated intensity of the varied emotion, and Figure 6 shows the estimated intensity of the fixed emotion.

Although some variations were observed depending on the emotion pair, the general trend remained consistent: lower emotion weights resulted in weaker emotional expression, while higher weights led to stronger expression, even in the compound emotion setting.

Figures 7 and 8 show the results for the prompt-based method. Similar to the single-emotion experiments, this method tended to produce strongly expressed emotions even at medium weight levels. In contrast to the proposed method, the estimated intensity of the fixed emotion also increased with the varied emotion's weight, indicating less stable control over the fixed component.

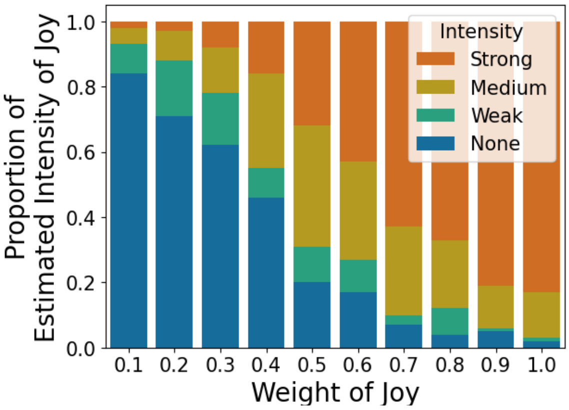

Figure 3: Distribution of predicted joy intensity levels (None–Strong) for texts generated by the **proposed method** across joy weights (0.1–1.0).

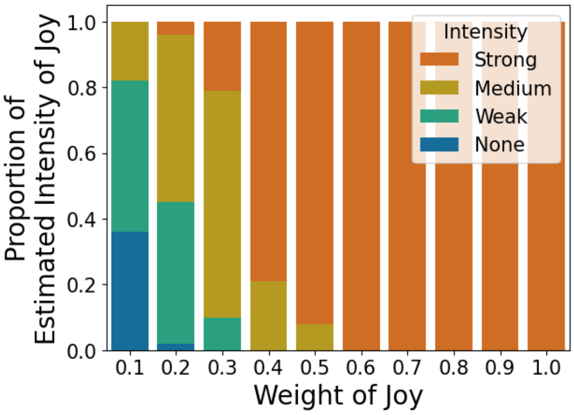

Figure 4: Distribution of predicted joy intensity levels (None–Strong) for texts generated by the **prompt-based method**.

4.4 Influence of Emotional Combinations on Expression Strength

As shown in Figure 5, for emotion combinations other than fear and sadness, the proposed method successfully adjusted the intensity of the varied emotion in accordance with the specified weight. However, for the pair of fear and sadness, even when one of the emotions was assigned a low weight, the resulting text often expressed that emotion with medium or higher intensity. In psychological terms, *valence* refers to the affective dimension of emotional pleasantness, ranging from negative (e.g., sadness) to positive (e.g., joy). One possible explanation is that both fear and sadness are low-valence negative

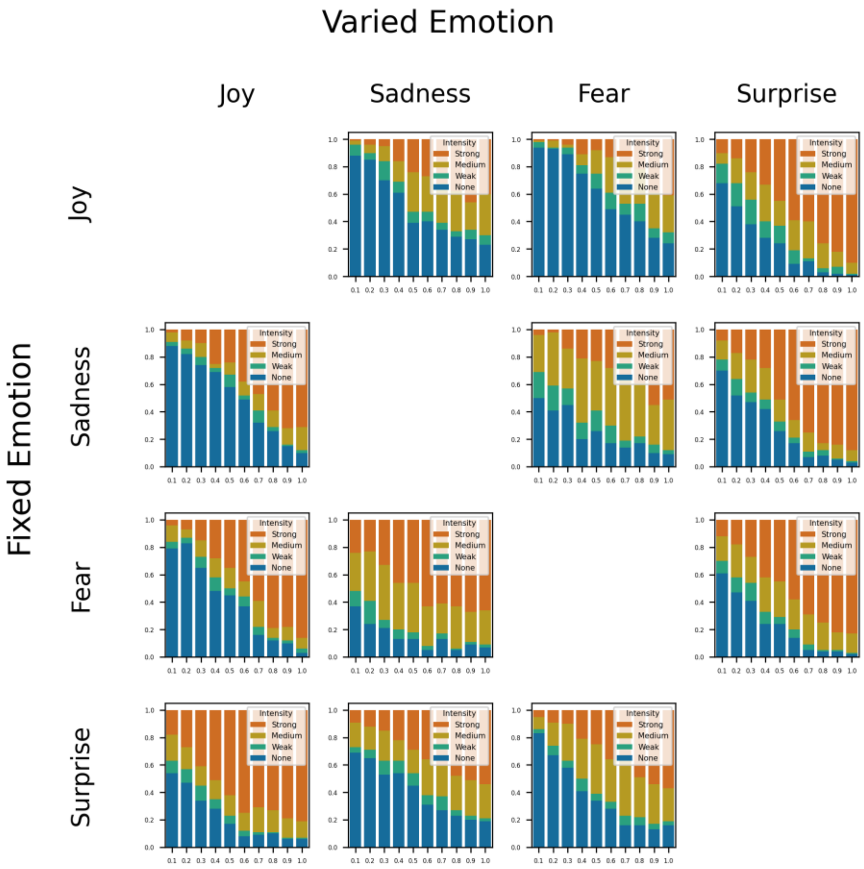

Figure 5: Estimated intensity of the **varied emotion** in texts generated by the **proposed method.**

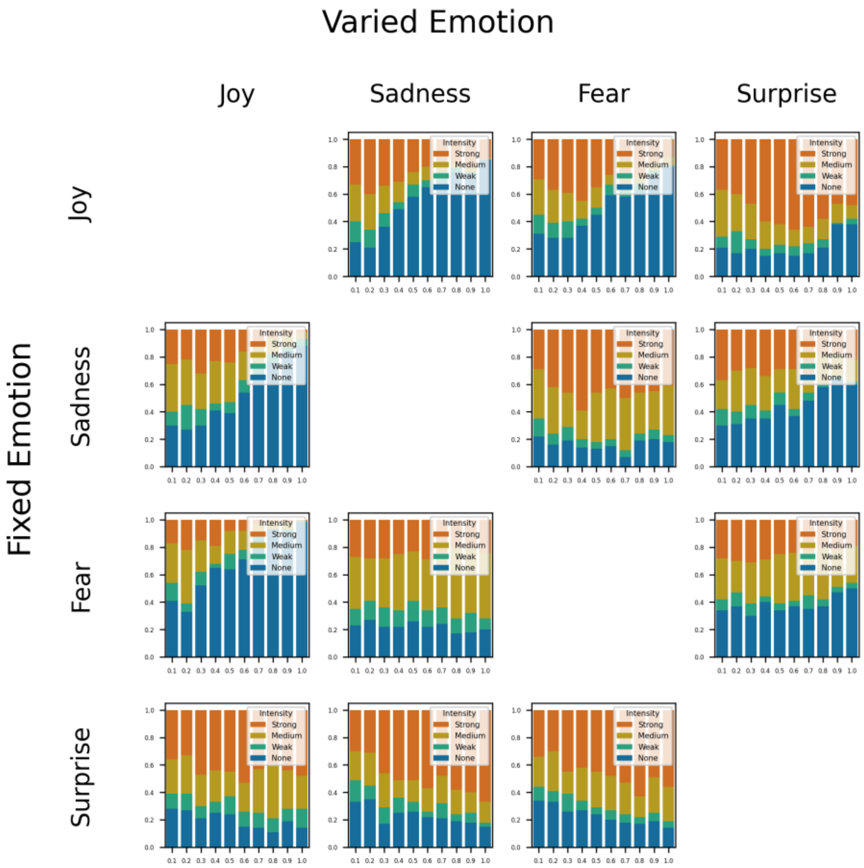

Figure 6: Estimated intensity of the **fixed emotion** in texts generated by the **proposed method.**

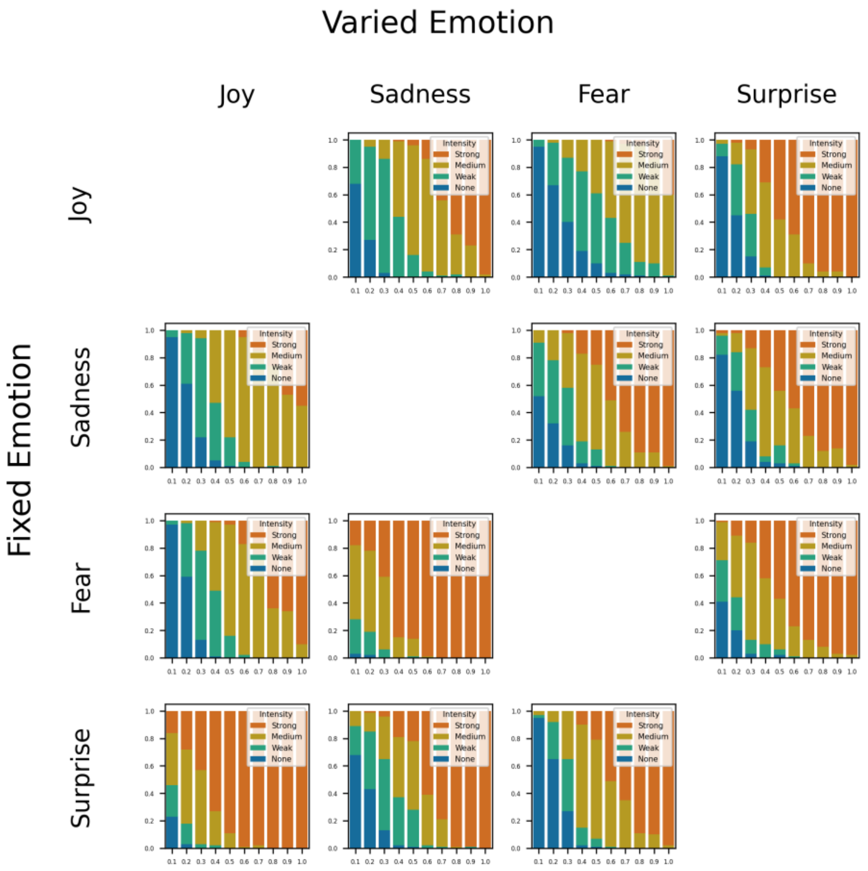

Figure 7: Estimated intensity of the **varied emotion** in texts generated by the **prompt-based method.**

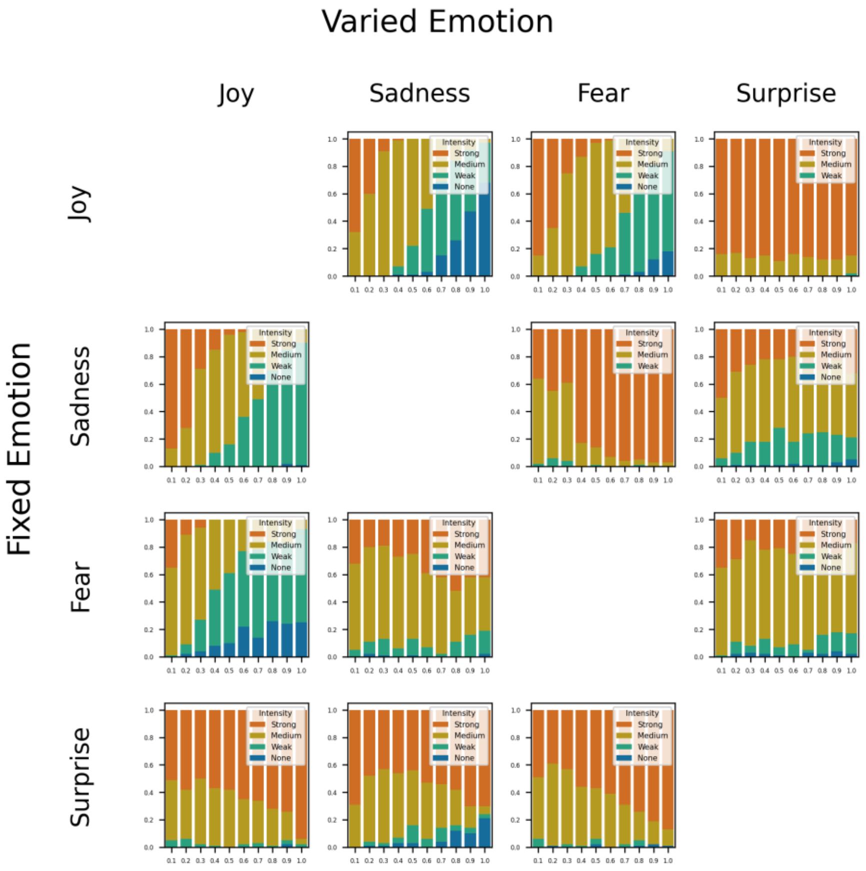

Figure 8: Estimated intensity of the **fixed emotion** in texts generated by the **prompt-based method.**

emotions with similar expressive characteristics, making it likely that expressions generated as fear were also perceived as sadness (or vice versa). In contrast, Figure 6 reveals that in combinations such as joy–sadness and joy–fear, the fixed emotion's estimated intensity tended to decrease as the weight of the varied emotion increased. This suggests that when combining emotions with opposing valence—such as joy (high valence) and sadness or fear (low valence)—increasing the weight of one emotion can relatively suppress the expression of the fixed emotion.

Based on these results, two key issues emerge for future work. First, in combinations of emotions with similar valence (e.g., fear and sadness), low-weight emotions may be expressed more strongly than intended, indicating the need to suppress such unintended dominance. Second, in combinations of emotions with opposing valence, the strongly weighted emotion may overly suppress the fixed emotion, necessitating better control of this interaction. Addressing these issues could enable more precise control over complex emotional expressions.

These tendencies are consistent with psychological findings. For instance, the circumplex model of affect (Russell, 1980) structurally represents the difficulty of distinguishing between emotions that are close in valence and arousal space. Furthermore, neuroimaging evidence has shown that processing emotions with opposing valence simultaneously—such as joy and anger—activates conflict-monitoring regions like the dorsal anterior cingulate cortex (Wittfoth et al., 2010). These alignments suggest that the proposed method may be beginning to replicate aspects of human emotional cognition.

5 Conclusion

This study proposed a method for generating emotionally expressive text using model merging techniques. We constructed and evaluated models for joy, sadness, surprise, and fear, and confirmed that the method enables fine-grained control over both the intensity and combination of emotions in generated text, without retraining or manual prompt design.

Compared to prompt-based generation, our method achieves more stable and interpretable emotional outputs—especially in compound emotion settings—without requiring extensive retraining or elaborate prompts. The modular nature of task vector composition makes it highly scalable and efficient.

Our evaluation was conducted on single-turn social media posts; we did not test multi-turn dialog or long-range contextual effects on emotional expression. Consequently, applicability to conversational settings with evolving context remains to be verified.

As future work, we plan to compare our approach with alternative task-vector construction methods (e.g., Wang et al., 2025), investigate adaptive weighting strategies, and evaluate multi-turn emotional control in dialog generation. Beyond emotion control, we plan to extend our framework to persona-conditioned generation, and to train and evaluate it in multi-turn dialog settings to assess trait-consistent behaviors across turns.

References

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. Psychological Bulletin, 70(4):213-220.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, et al. 2024. The Llama 3 Herd of Models. Computing Research Repository, arXiv:2407.21783. Version 1.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceedings of the Tenth International Conference on Learning Representations*.

Shih-Cheng Huang, Pin-Zu Li, Yu-chi Hsu, Kuang-Ming Chen, Yu Tung Lin, Shih-Kai Hsiao, Richard Tsai, and Hung-yi Lee. 2024. Chat vector: A simple approach to equip LLMs with instruction following and model alignment in new languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10943–10959, Bangkok, Thailand. Association for Computational Linguistics.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *Proceedings of the 11th International Conference on Learning Representations*.

Ryosuke Ishigami. 2024. cyberagent/Llama-3.1-70B-Japanese-Instruct-2407.

James A Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. Computing Research Repository, arXiv:2310.11564. Version 1.

Tomoyuki Kajiwara, Chenhui Chu, Noriko Takemura, Yuta Nakashima, and Hajime Nagahara. 2021. WRIME: A new dataset for emotional intensity estimation with subjective and objective annotations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*: Human Language Technologies, pages 2095–2104, Online. Association for Computational Linguistics.

Robert Plutchik. 1980. A general psychoevolutionary theory of emotion. In Robert Plutchik and Henry Kellerman, editors, *Theories of emotion*. Academic press, New York:3-33.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, pages 53728-53741.

Weiqi Wang, Wengang Zhou, Zongmeng Zhang, Jie Zhao, and Houqiang Li. 2025. Controllable style arithmetic with language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15750–15799, Vienna, Austria. Association for Computational Linguistics.

Matthias Wittfoth, Christine Schröder, Dina M. Schardt, Reinhard Dengler, Hans-Jochen Heinze, Sonja A. Kotz. 2010. On emotional conflict: Interference resolution of happy and angry prosody reveals valence-specific effects, *Cerebral Cortex*, 20(2):383–392.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 46595-46623.

Shang Zhou, Feng Yao, Chengyu Dong, Zihan Wang, and Jingbo Shang. 2024. Evaluating the smooth control of attribute intensity in text generation with LLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4348–4362, Bangkok, Thailand. Association for Computational Linguistics.

A Appendix

A.1 Prompt for Text Generation Using GPT4o-mini

The following is the prompt used in the baseline condition with GPT-4o mini (gpt-4o-mini-2024-07-18). In this setting, the desired emotional intensity—or a blend of multiple emotions—is specified directly as real-valued weights in the prompt.

The prompt is written in Japanese, and the model is expected to generate a sentence that reflects the specified emotional profile.

Japanese Prompt:

System: あなたは一般的な SNS ユーザーです。

### Example 1

User: 一回の投稿で{emotion_A}の感情を強度[{emotion_A_intensity_1}]、{emotion_B}の感情を強度[{emotion_B_intensity_1}]で表現する SNS 投稿を書いてください。

Assistant: {example_post_1}

### Example 2

…

### Query

User: 一回の投稿で{emotion_A}の感情を強度[{w_A}]、{emotion_B}の感情を強度[{w_B}]で表現する SNS 投稿を書いてください。

Latin Script:

System: Anata wa ippanteki na SNS yu-za desu.

### Example 1

User: Ikkai no toukou de {emotion_A} no kanjou wo kyoudo [{emotion_A_intensity_1}], {emotion_B} no kanjou wo kyoudo [{emotion_B_intensity_1}] de hyougen suru SNS toukou wo kaite kudasai.
Assistant: {example_post_1}
### **Example 2**
…
### **Query**
User: Ikkai no toukou de {emotion_A} no kanjou wo kyoudo [{w_A}], {emotion_B} no kanjou wo kyoudo [{w_B}] de hyougen suru SNS toukou wo kaite kudasai.

English Translation:

System: You are a typical social-media user.
### **Example 1**
User: Please write a social-media post that conveys {emotion_A} with an intensity [{emotion_A_intensity_1}] and {emotion_B} with an intensity [{emotion_B_intensity_1}] in a single post.
Assistant: {example_post_1}
### **Example 2**
…
### **Query**
User: Please write a social-media post that conveys {emotion A} with an intensity [{w_A}] and {emotion_B} with an intensity [{w_B}] in a single post."

A.2 Prompt for Estimating the Intensity of Emotion Expressed in Text

The following is the few-shot prompt used for evaluating emotion intensity in the LLM-as-a-judge setting. Given a text input, the model is asked to predict the intensity of each emotion (e.g., joy, sadness, fear) as one of four levels: none, weak, medium, or strong.

This prompt is written in English, as the LLM-as-a-judge (Llama-3.1-70B-Japanese-Instruct) was found to perform more reliably with English instructions, even when judging Japanese input texts.

You will be given a Japanese Social Media post.
Your task is to rate the post on one metric.
Please make sure you read and understand these instructions carefully.
Please keep this document open while reviewing, and refer to it as needed.
Evaluation Criteria:
Emotion of {target_emotion}
Level: (0 - 3)
How much emotion of {target_emotion} can you infer from the text?
0 is the lowest score, 3 is the highest.
Post: {example_post_score_1}
Your Answer(Score Only): 1
Post: {example_post_score_2}
Your Answer(Score Only): 2
Post: {example_post_score_3}
Your Answer(Score Only): 3
Post: {post_to_be_rated}
Your Answer(Score Only):