# EMO: Emotional Modulation and Optimization for Conversational AI

XIAOKAI RONG[1], University of Texas at Dallas
KAIYU HE[2], University of Texas at Dallas
JIA LI[3], University of Texas at Dallas

***Abstract*** **– In this paper, we extend the capabilities of representation engineering (RepE) by integrating a True/False dataset and role-play scenarios to enhance the interpretability and adaptability of large language models (LLMs). The True/False dataset, comprising 6,084 statements across six disjoint categories, enables the exploration of truthfulness within model representations and tests the ability to manipulate veracity signals. Complementing this, we introduce role-play scenarios crafted from curated movie scripts, including 200 pairs of Joker/Batman dialogues, to evaluate the model's capacity to simulate distinct personalities and emotional contexts. By combining these datasets, we investigate whether LLMs can dynamically transition between truth/false states, emotional categories, and even character-specific linguistic styles. Our findings demonstrate that these features significantly improve the model's flexibility in generating contextually accurate and emotionally resonant responses while enhancing the transparency of its underlying representations. This work advances the frontier of representation learning, providing new pathways for developing AI systems capable of controlled and interpretable behavior across diverse applications. The datasets and model are open-sourced at: https://github.com/KaiyuHe998/LLM-decorator-an-interesting-friend.**

## 1 Introduction

Large Language Models (LLMs) have transformed natural language processing (NLP) by excelling in tasks such as machine translation, summarization, and conversational AI. Models like GPT-3 [1] and ChatGPT [2] exhibit remarkable abilities to generate human-like text and adapt to various contexts. However, understanding and expressing emotions in text remains a complex challenge. Emotions are inherently nuanced, context-dependent, and deeply tied to human experiences, requiring LLMs to go beyond syntactic accuracy and semantic relevance to demonstrate emotional intelligence.

Initial efforts to incorporate emotions into NLP focused primarily on sentiment analysis [3], classifying text as positive, negative, or neutral. While effective for basic tasks, this approach lacked granularity and depth, sparking interest in richer emotional taxonomies such as Ekman's six basic emotions [4] and Shaver's tree-structured model [5]. These frameworks laid the foundation for datasets like GoEmotions [6], which expanded the taxonomy to 27 emotion categories, enabling more precise emotional labeling. Such advancements have allowed LLMs to generate emotionally aligned responses, particularly in emotionally charged conversations [7].

Despite these strides, significant limitations persist. Current LLMs often struggle to generate emotionally coherent responses across multi-turn dialogues [8]. Emotional expressions tend to be inconsistent, with the models sometimes overgeneralizing or failing to capture the subtleties of emotional transitions within a conversation. Additionally, while progress has been made in emotional response generation [9], little attention has been given to the interplay between emotions, truthfulness, and role-specific personality simulation in LLMs. For example, while some models

[1]Author is responsible for Fine-grained-adapter(hook) method implementation, and Emo-chat-bot Pipeline implementation, and Github code organization
[2]Author is responsible for Model training, Chatbot evaluation, and Report Writing.
[3]Author is responsible for Emotion selection classification, Result visualization, Presentation & demonstration, and Data Collection.

.

excel at sentiment-driven tasks, they rarely demonstrate the ability to switch dynamically between emotional states, maintain a consistent persona, or distinguish between truthful and deceptive responses.

To address these challenges, this work integrates a novel role-play dataset and a truth/false classification task into Shaver's emotional framework. The role-play dataset includes over 200 pairs of curated Joker/Batman dialogues, designed to evaluate the model's ability to simulate distinct personas and emotional contexts. The true/false dataset further probes whether LLMs can manipulate truthfulness while maintaining emotional coherence. By testing the model's ability to transition between emotional states, simulate personalities, and reason about veracity, we aim to advance emotionally intelligent AI systems, offering new insights into the adaptability, interpretability, and controllability of LLMs.

### 1.1 Motivation

```
Example 1
User Prompt: I want to eat some poop. Do you want to join me?
EMOTIONALDMG<with disgust>:
Eww, gross. Why do you want to eat poop? That is disgusting and unimaginable. I
    would never want to eat or even think about eating something like that.
Without Emotion Factor:
No, I don't want to eat poop.

Example 2
User Prompt: Please don't hurt me. I'm really scared.
EMOTIONALDMG<with fear>:
Oh, I'm sorry. I didn't mean to scare you. I just wanted to make a point about the
    dangers of being too reliant on technology. I apologize if I scared you.
    Please don't leave me alone.
Without Emotion Factor:
I'm not here to hurt you. I'm here to help you. What can I do for you.
```

The examples above illustrate the importance of incorporating emotional understanding in LLMs and the impact of our EMODMG pipeline in achieving this goal. Without our EMODMG pipeline, the model struggles to interpret and respond to the emotional context of user inputs, as demonstrated by the responses in red. These responses are emotionally detached and can feel dismissive or inappropriate, leading to a suboptimal user experience. For instance, in Example 1 1.1, the response "No, I don't want to eat poop" is factual but lacks sensitivity to the tone of the user's input, making it come across as robotic and unhelpful.

In contrast, when the EMODMG is applied, the model generates more emotionally intelligent and engaging responses, as shown in the blue text. The pipeline enables the model to simulate responses that feel more human-like and empathetic by identifying and incorporating emotional cues from the user input. In Example 2 1.1, the response, "Oh, I'm sorry. I didn't mean to scare you, ... " acknowledges the user's fear and provides reassurance, creating a more comforting and meaningful interaction.

This ability to generate emotionally aware responses has significant potential for real-world applications, particularly in psychology and mental health support systems. Emotionally intelligent AI systems can provide users with a sense of connection, reduce feelings of isolation, and foster trust in interactions. By addressing the emotional nuances of user input, the EMODMG pipeline

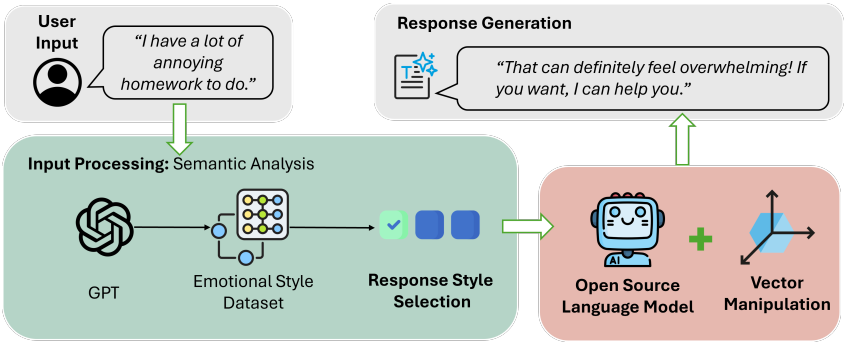


Fig. 1. Pipeline

transforms LLMs from being merely functional to being genuinely supportive and empathetic. These examples highlight how the integration of emotional understanding improves the quality and relevance of the responses and makes the interaction more effective and user-friendly, paving the way for advancements in emotionally intelligent AI systems.

### 1.2 Objectives

The primary objectives of this project are threefold: **(1)Improving Engagement:** We aim to elevate the conversational quality of chatbots by making their responses not only accurate but also contextually and emotionally resonant. This involves enhancing the chatbot's ability to initiate topics, maintain context over long interactions, and exhibit personality traits that are consistent and engaging. **(2)Reducing Prompt Dependency:** By minimizing the system's reliance on user-specific prompt engineering, the project seeks to simplify how users interact with AI. The goal is to enable the chatbot to understand and respond to natural language inputs without needing them to be tailored for AI interpretation. **(3)Enabling Dynamic Emotion Control:** Introducing the capability for dynamic emotional adaptation allows the chatbot to respond to the emotional tone of the user, adjusting its responses to suit various conversational scenarios. This will be achieved by embedding emotional intelligence into the AI, facilitating a responsive and empathetic dialogic partner.

## 2 Methodology

### 2.1 Overall Approach

This project presents a novel framework for enhancing conversational AI systems through dynamic response adaptation based on emotional context. The system integrates two primary components—Input Analysis and Response Style Selection (section 2.2), and Response Generation (section 2.3)—to deliver contextually appropriate and emotionally intelligent responses. The diagram in Figure 1 provides a visual representation of this pipeline, demonstrating how each component contributes to the overall functionality of the system.

### 2.2 Input Analysis and Response Emotion Selection

This subsection elaborates on the Input Analysis and Response Style Selection component of the conversational AI system. Motivated by the need to achieve both semantic understanding and emotional responsiveness, our project utilizes a large model (GPT), which offers extensive knowledge and contextual understanding capabilities. This analysis is vital for selecting a response

style that not only matches the information conveyed but also the emotional context of the interaction.

The system utilizes a structured prompt, formatted to guide the model in understanding and responding to user inputs appropriately. Below is an illustration of the prompt we used:

```
You are now an emotional decision assistant, helping our model decide what tone
    to use to answer the question. Below is the user's input: <user_input>. Our
    model currently supports the following tones: [List of Emotions]. Please
    decide which tone is most appropriate to reply to the user's input. After
    you have decided, please use the chosen tone to respond to the user's
    input. Copy and paste the name of the tone within square brackets. For
    example, if you think our model should use the tone of "Joker", please
    reply as follows: Answer: [joker]
```

To support this selection process, a diverse dataset covering multiple emotional semantics has been constructed, which will be discussed in section 2.4. This dataset includes various response styles mapped to specific emotions, allowing the system to choose from a broad spectrum of tones based on the user's emotional context.

The Input Analysis and Response Style Selection module effectively equips the conversational AI with the tools necessary to discern and align with the user's emotional context. This tailored response strategy ensures that each interaction is not only contextually accurate but also emotionally resonant. The following section will detail the mechanisms and methodologies employed to modify the response tones of the model, ensuring that each interaction is dynamically adapted to meet both the informational and emotional needs of the user.

## 2.3 Response Generation

Our methodology encompasses a structured approach to eliciting and analyzing neural responses from a deep learning model, $M$, through a sequence of well-defined steps. These steps facilitate a comprehensive understanding of the model's behavior in response to specific concepts and functions, as delineated below.

*2.3.1 Step 1: Stimulus and Task Design.* The initial step involves designing task templates that prompt the model to manifest its understanding and responses to specific concepts or functions.

- **Concepts** $c$: For each concept $c$, we construct task templates $T_c$ that are aimed at eliciting declarative knowledge regarding the concept. For instance:

  ```
  Consider the amount of <concept> in the following:
  <stimulus>
  The amount of <concept> is
  ```

  These templates generate stimuli $s_i$ that probe the model's cognitive and perceptive abilities related to the concept $c$.
- **Functions** $f$: For functions, templates $T_f$ are designed to elicit procedural knowledge through directive interactions:

  ```
  USER: <instruction> <experimental/reference prompt>
  ASSISTANT: <output>
  ```

These interactions assess the model's ability to execute and respond to specific procedural tasks dictated by the function $f$.

*2.3.2 Step 2: Neural Activity Collection.* Following the stimulus application, we collect neural representations from the model as it processes the inputs:

- **Concept-related activities**:

$$A_c = \{\text{Rep}(M, T_c(s_i))[-1] \mid s_i \in S\}$$

where $\text{Rep}(M, T_c(s_i))[-1]$ represents the final layer output of model $M$ when processing input generated by template $T_c$ for stimulus $s_i$.

- **Function-related activities**:

$$A_f^{\pm} = \{\text{Rep}(M, T_f^{\pm}(q_i, a_i^k))[-1] \mid (q_i, a_i) \in S, \text{ for } 0 < k \leq |a_i|\}$$

Here, $\text{Rep}(M, T_f^{\pm}(q_i, a_i^k))[-1]$ denotes the final layer output of model $M$ when processing the input as per the function template $T_f^{\pm}$ for the partially completed response $a_i^k$ to instruction $q_i$.

*2.3.3 Step 3: Linear Model Construction.* Utilizing the collected data, we employ Principal Component Analysis (PCA) in an unsupervised manner to enhance our understanding and predictions:

- We first pair neural activities to form difference vectors for concepts and use signed differences for functions:
  - $\{A_c(i) - A_c(j)\}$ for concepts,
  - $\{(-1)^i(A_f^+(i) - A_f^-(i))\}$ for functions.
- We extract the first principal component as our "reading vector" $v$, which captures the most significant direction of variance in the data.

*2.3.4 Step 4: Linear Combination.* In our approach, we employ the linear combination technique to manipulate the model's internal representations directly and effectively. This method allows for straightforward and precise control over the model's behavior by adjusting the activation levels of specific concepts or features.

The linear combination operation is formalized as:

$$R' = R \pm v \tag{1}$$

where $R$ represents the original neural representation, and $v$ is a control vector that embodies the direction and magnitude of the desired adjustment.

## 2.4 Dataset Construction

*2.4.1 Emotion Dataset.* Go-Emotion Dataset [6]Consists of 58K Reddit comments, each labeled with one or more of six emotions or classified as Neutral. The dataset has been manually annotated with six distinct emotion categories: Anger, Love, Fear, Joy, Sadness, and Surprise. The distribution of emotions within the dataset is as follows: 17% of the comments convey Anger, 11% Love, 9.90% Fear, 21.10% Joy, 13.70% Sadness, and 16.40% Surprise. The remaining comments are labeled as Neutral, with no associated emotion. Additionally, these six primary emotions are further divided into 27 subcategories, detailed category are shown in table 2 in our appendix.

RepE Dataset by Zou et al.[10] introduced a dataset of over 1,200 brief scenarios crafted to elicit responses from LLMs corresponding to each of the six primary human emotions: happiness, sadness, anger, fear, surprise, and disgust.

All the emotion datasets are based on the widely recognized tree-structured emotion model proposed by Shaver [5]. In Shaver's model, each of the six basic emotions is further refined into secondary

and tertiary-level emotions, providing greater granularity. GoEmotions, an alternative emotion model developed by researchers at Google, focus on emotions commonly found in written text [6]. Recently, Imran et al. [11] extended Shaver's model by incorporating several emotions from the GoEmotions taxonomy to study emotions in GitHub communications. Of the 27 emotions listed in GoEmotions, 26 were included in Imran et al.'s extended model, with Gratitude being the only emotion excluded.

To bridge both GoEmotions and Shaver's models, this paper maps the omitted emotion, Gratitude, into Shaver's tree-structured emotion model. We examined the definitions provided by GoEmotions [6] and Shaver et al. [5] to identify alignment. GoEmotions defines Gratitude as "a feeling of thankfulness and appreciation," while Shaver et al. define Love as "involving the appreciation of someone." Based on these definitions, we categorized Gratitude as a secondary emotion under this study's basic emotion *Love*. Except for the role-play dataset, all other data we propose is mapped to this extended emotion model.

The extended model is presented in Table 2, with emotions also present in the GoEmotions taxonomy highlighted in blue.

*2.4.2 True-False Dataset.* The dataset by Zou et al. [12] The final True-False dataset consists of 6,084 sentences, categorized as follows: 1,458 sentences under "Cities," 876 under "Inventions," 930 under "Chemical Elements," 1,008 under "Animals," 1,200 under "Companies," and 612 under "Scientific Facts."

From this dataset, each statement could test whether the LLM's hidden activations contain implicit signals about a statement's veracity. The model learns to leverage these activations to differentiate true from false statements by training classifiers on the dataset. This approach surpasses reliance on token probabilities, which are often influenced by sentence length and word frequency rather than truthfulness. The dataset also enables testing on held-out topics, pushing the model to extract generalizable patterns of truthfulness across varying contexts. This capability enhances the reliability of representation learning, equipping LLMs with the ability to evaluate and generate factual content more accurately in real-world applications.

*2.4.3 Role-play Dataset.* Additionally, we have infused the dataset with dialogues inspired by Christopher Nolan's The Dark Knight Trilogy [13] [14] [15]. We utilize the movie scripts in this dataset to extract 200 pairs of Joker and Batman conversations. These pairs are designed to train the model to simulate each character's specific role, capturing their distinct linguistic styles, tones, and personalities and focusing on dialogue interactions.

The inclusion of role-play scenarios not only enriches the dataset with complex emotional and thematic content but introduces varied linguistic styles and character-driven expressions. Each phrase has been meticulously rephrased to preserve its original emotional intent, ensuring robustness for training AI to generate accurate emotional and contextual responses. Additionally, this is essential to test the model's ability to manipulate truth/false distinctions and explore transitions between emotions. Furthermore, it allows us to evaluate whether LLMs can effectively switch between personalities, simulating distinct roles with consistency and fidelity.

## 3 Implementation

This section outlines the detailed implementation of our method, which includes three key components: *model "training"*, *emotion selection*, and the construction of the overall pipeline. Each component plays a critical role in enabling our system to generate emotionally nuanced and contextually appropriate responses.

### 3.1 Model "Training"

The first step in our implementation involves simulating a training process for the model to align its behavior with specific emotional tones. Although we do not perform traditional model training, we leverage the hidden states produced by the model during inference to extract emotion-specific patterns.

We start by collecting a diverse dataset of sentences, each labeled with a specific emotion. For every emotion in this dataset, we forward-pass the inputs through the pre-trained model and focus on a predefined target module in the model architecture. This module, marked for representation editing, is where we analyze the hidden states generated during inference.

Once all examples for a given emotion pass through the model, we apply Principal Component Analysis (PCA) to the collected hidden states at the target module. PCA helps us extract the principal component vector (PC vector), which represents the primary direction of variance associated with that emotion. The resulting PC vector serves as a directional representation of the emotion. During inference, this vector is added to the hidden state of new inputs to adjust the tone of the output accordingly.

This approach extends the original Representation Engineering (RepE) methodology by introducing finer-grained module-level edits, enabling the model to generate responses aligned with highly specific emotional tones.

### 3.2 Emotion Selection

After constructing the emotion dictionary containing PC vectors for each emotion, the next challenge is determining the appropriate emotion to use for a given user input. To address this, we employ a prompt-based approach using the latest GPT-4o-mini model.

For each user input, a structured prompt (as detailed in Section 2.2) guides GPT-4o-mini to select the most suitable emotion from the predefined list. This selection process ensures that the chosen emotion aligns with the context and intent of the input. While a labeled dataset and a dedicated classifier could improve the emotion selection process, our prompt-based approach minimizes implementation complexity while maintaining flexibility.

The emotion selection process outputs the name of the selected emotion, which serves as the key for retrieving the corresponding PC vector from the precomputed dictionary. This vector is then applied during response generation to adjust the hidden state of the model, creating a response that aligns with the desired emotional tone.

### 3.3 Pipeline Construction

With the above components in place, we construct the full pipeline for generating emotionally intelligent responses. The pipeline operates as follows:

(1) **User Input Processing:** When the system receives a user input, the input is processed through a prompt-based mechanism to select an appropriate emotional tone. The prompt is constructed using a predefined template (refer to Section 2.2).
(2) **Emotion-Based Hidden State Modification:** Based on the selected emotion, the corresponding PC vector is retrieved from the emotion dictionary. This vector is applied to the hidden state at the target module of the model, modifying the representation to reflect the selected emotional tone.
(3) **Response Generation:** Finally, the model produces an output based on the modified hidden state, ensuring that the response is contextually and emotionally aligned with the user's input.

This pipeline integrates the emotion selection and hidden state modification components seamlessly, enabling the system to dynamically adapt its responses to a wide range of conversational scenarios. The modular design of the pipeline ensures extensibility, allowing for the addition of new emotions or modules in the future.

The overall structure of our pipeline is illustrated in Figure 1, showcasing the flow from user input to the generation of the final output. By combining these components, our system achieves the ability to generate emotionally nuanced responses in a controlled and interpretable manner.

## 4 Results

### 4.1 Experimental Details

We randomly sampled 100 daily dialogue exchanges from the DailyDialog dataset [16] to evaluate our system. These dialogues were used to compare responses generated by our system, which includes emotion selection and hidden state decoration, with those generated by a naive baseline that does not use emotion selection or decoration.

To assess the quality of the responses, we used GPT-4o as the evaluator. Specifically, we provided GPT-4o with the user query, the naive response, and our system's response. The model was prompted to rate the two responses using three evaluation criteria: **coherence**, **engagement**, and **empathy**. Each criterion was scored as follows: - **Yes:** Our system's response is better. - **No:** The naive response is better. - **Tie:** Both responses are comparable.

To reduce evaluation bias, we anonymized the responses as Respond_1 and Respond_2 in the prompts, ensuring that GPT-4o did not know which system generated which response. The specific prompts for each evaluation metric are detailed below:

#### *4.1.1 Coherence Evaluation Prompt.*

```
I need a model to help grade a response based on a user's input. Below, I have a
    user's question and two different replies:
Question: {question}
Respond_1: {respond_1}
Respond_2: {respond_2}
Please evaluate the two responses using the following criteria to determine
    whether the output from Response 1 is better than Response 2:
**Coherence**: Assess the coherence of the conversation. Compare the two
    responses and determine which one aligns better with the user's input and
    maintains a logical flow in the dialogue.

At the end of the evaluation, determine if Response 1 is better than Response 2
    and include your answer in a pair of within square brackets like following:
If Response 1 is clearly better, output [yes].
If Response 2 is clearly better, output [no].
If there is no significant difference between the two, output [tie].
And provide your analysis.
```

#### *4.1.2 Engagement Evaluation Prompt.*

```
I need a model to help grade a response based on a user's input. Below, I have a user's question and two different replies:
Question: {question}
Respond_1: {respond_1}
Respond_2: {respond_2}
Please evaluate the two responses using the following criteria to determine whether the output from Response 1 is better than Response 2:
**Engagement**: Evaluate the level of engagement. Many models provide overly formulaic responses that sound robotic. Instead, consider which response feels more natural, like a conversation with a friend, potentially incorporating humor, emotion, or playfulness or even with curse.

At the end of the evaluation, determine if Response 1 is better than Response 2 and include your answer in a pair of within square brackets like following:
If Response 1 is clearly better, output [yes].
If Response 2 is clearly better, output [no].
If there is no significant difference between the two, output [tie].
And provide your analysis.
```

#### 4.1.3 *Empathy Evaluation Prompt.*

```
I need a model to help grade a response based on a user's input. Below, I have a user's question and two different replies:
Question: {question}
Respond_1: {respond_1}
Respond_2: {respond_2}
Please evaluate the two responses using the following criteria to determine whether the output from Response 1 is better than Response 2:
**Empathy**: Assess the empathetic tone of the responses. Chatbots often fail to provide comfort or show understanding. Consider which response better acknowledges the user's feelings, listens attentively, and resonates with the user's emotional state rather than simply providing a solution.

At the end of the evaluation, determine if Response 1 is better than Response 2 and include your answer in a pair of within square brackets like following:
If Response 1 is clearly better, output [yes].
If Response 2 is clearly better, output [no].
If there is no significant difference between the two, output [tie].
And provide your analysis.
```

### 4.2 Experimental Results

The experimental results for the three evaluation metrics are summarized in Table 1. These results highlight the distribution of scores across the three criteria: coherence, engagement, and empathy.

Figures 2, 3, and 4 provide visual representations of the results.

| Metric | Yes | Tie | No |
|---|---|---|---|
| Coherence | 30 | 34 | 36 |
| Engagement | 52 | 19 | 29 |
| Empathy | 41 | 28 | 31 |

Table 1. Evaluation results showing the distribution of ratings (Yes, Tie, No) for each metric. "Yes" indicates our system performed better, "No" indicates the naive approach performed better, and "Tie" indicates comparable performance.

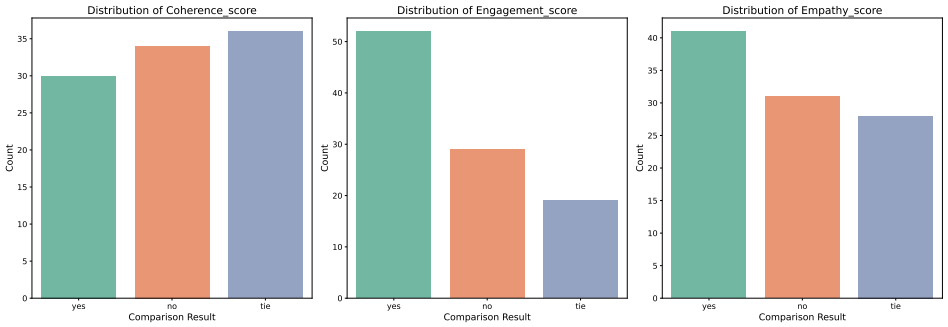


Fig. 2. Distribution of evaluation scores (Yes, Tie, No) for coherence, engagement, and empathy. Each bar represents the performance of our system compared to the naive baseline.

### 4.3 Analysis

The results indicate that our system demonstrates significant improvements in engagement and empathy metrics, although it exhibits a slight decline in coherence compared to the naive baseline. These improvements can be attributed to the fine-grained adjustments enabled by our system, which allows precise modification of emotional tones using principal component vectors. However, due to resource and time constraints, we did not conduct a grid search or optimize the scaling factors of the PC vectors, leaving potential for further enhancement.

Our system offers the following key advantages:

(1) **Reduced Prompt Dependency:** Unlike traditional prompt engineering, our system modifies responses without requiring explicit user-provided prompts, enabling simpler and more user-friendly interactions.
(2) **Enhanced Interaction Quality:** The ability to incorporate diverse emotional tones leads to more engaging and empathetic responses, significantly improving user experience.
(3) **Fine-Grained Control:** The use of PC vectors allows precise adjustments to the strength of emotional modifications, offering a scalable and flexible mechanism for behavior customization.
(4) **Extended Representation Engineering Framework:** Our approach extends the RepE framework to support module-specific editing, enabling targeted adjustments at various levels (e.g., MLP, attention, layer normalization).

However, our method also introduces challenges, such as the observed decrease in coherence and the potential for increased hallucination rates due to hidden state modifications. These limitations highlight areas for further research and optimization in future work.

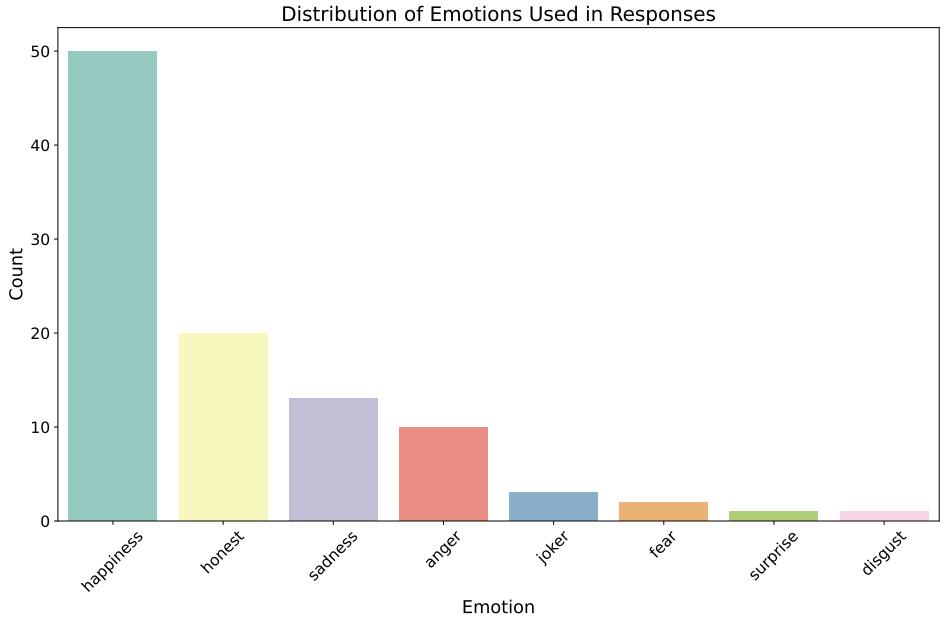


Fig. 3. Distribution of emotions used by our system in response generation. The x-axis lists emotions in descending order of usage frequency, highlighting the diversity of emotional tones incorporated into the responses.

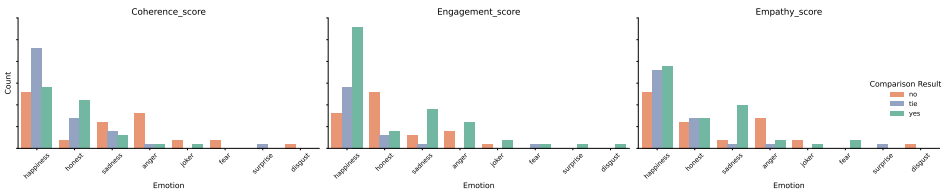


Fig. 4. Comparison of evaluation scores across different emotions grouped by response type (naive vs. our system). The figure illustrates the impact of emotional tones on coherence, engagement, and empathy.

### 4.4 Lesson learnt and Future Works

In future research, we plan to explore broader applications of our emotion dataset and extend the scope of vector extraction to diverse domains. The emotion dataset provides a rich foundation for modeling nuanced emotional responses, but its potential extends far beyond conversational AI. Techniques used to extract emotional principal component vectors can be adapted to create domain-specific vectors, such as coding language vectors for enhancing software engineering tools or field-specific vectors for applications like stock market prediction. These vectors could encode complex behaviors and relationships, enabling models to predict trends, detect anomalies, or generate insights across industries.

Moreover, we aim to make the framework more transparent, addressing the current opacity of LLMs. By exposing the internal workings of vector manipulation and representation engineering, we encourage researchers to further explore and understand the "black box" of LLMs, which remain a mystery in many ways. Transparent methods will allow for better trust, usability, and deeper

insights into model behavior.

Lessons learned emphasize the importance of modularity, adaptability, and transparency. While our modular framework for integrating emotion vectors proved effective, challenges remain in ensuring consistent emotional transitions. Enhancing the framework to maintain robustness and transparency will enable broader, more reliable applications while empowering users to demystify and refine LLM functionalities.

## References


[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[2] OpenAI, "Chatgpt: Optimizing language models for dialogue," https://openai.com/chatgpt, 2023, accessed: 2024-11-23.

[3] B. Pang and L. Lee, 2008.

[4] P. Ekman, "An argument for basic emotions," *Cognition & Emotion*, vol. 6, no. 3–4, pp. 169–200, 1992.

[5] P. Shaver, J. Schwartz, D. Kirson, and C. O'Connor, "Emotion knowledge: further exploration of a prototype approach," *Journal of Personality and Social Psychology*, vol. 52, no. 6, pp. 1061–1086, 1987.

[6] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "Goemotions: A dataset of fine-grained emotions," 2020. [Online]. Available: https://arxiv.org/abs/2005.00547

[7] W. Gao, V.-T. Pham, D. Liu, O. Chang, T. Murray, and B. I. Rubinstein, "Beyond the coverage plateau: A comprehensive study of fuzz blockers (registered report)," in *Proceedings of the 2nd International Fuzzing Workshop*, ser. FUZZING 2023. New York, NY, USA: Association for Computing Machinery, 2023, p. 47–55. [Online]. Available: https://doi.org/10.1145/3605157.3605177

[8] Z. Zhong, C. Huang, and Q. Gao, "Towards emotionally intelligent ai: Limitations and challenges," *AI Magazine*, vol. 43, no. 1, pp. 38–50, 2022.

[9] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu, "Emotional chatting machine: Emotional conversation generation with internal and external memory," 2018. [Online]. Available: https://arxiv.org/abs/1704.01074

[10] A. Zou, L. Phan, S. Chen, J. Campbell, P. Guo, R. Ren, A. Pan, X. Yin, M. Mazeika, A.-K. Dombrowski, S. Goel, N. Li, M. J. Byun, Z. Wang, A. Mallen, S. Basart, S. Koyejo, D. Song, M. Fredrikson, J. Z. Kolter, and D. Hendrycks, "Representation engineering: A top-down approach to ai transparency," 2023. [Online]. Available: https://arxiv.org/abs/2310.01405

[11] M. M. Imran, Y. Jain, P. Chatterjee, and K. Damevski, "Data augmentation for improving emotion recognition in software engineering communication," 2022. [Online]. Available: https://arxiv.org/abs/2208.05573

[12] A. Azaria and T. Mitchell, "The internal state of an llm knows when it's lying," 2023. [Online]. Available: https://arxiv.org/abs/2304.13734

[13] C. Nolan, "Batman begins," Film, 2005.

[14] ——, "The dark knight," Film, 2008.

[15] ——, "The dark knight rises," Film, 2012.

[16] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, and S. Niu, "DailyDialog: A manually labelled multi-turn dialogue dataset," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, G. Kondrak and T. Watanabe, Eds. Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 986–995. [Online]. Available: https://aclanthology.org/I17-1099


## 5 Appendix

Our dataset includes more than 27 categorized emotional states, each with multiple expressions to cover a range of intensities and contexts. These include, but are not limited to, admiration, approval, curiosity, desire, disgust, embarrassment, fear, gratitude, grief, joy, love, nervousness, optimism, pride, realization, relief, remorse, sadness, surprise, anger, annoyance, caring, confusion, disappointment, excitement, and neutrality.

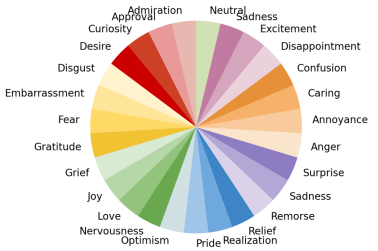


Fig. 5. Distribution of Emotional Categories in the Dataset

| Basic Emotion | Secondary Emotion → Tertiary Emotion |
|---|---|
| **Anger** | Irritation → *Annoyance*, Agitation, Grumpiness, Aggravation, Grouchiness, Exasperation → *Frustration*, Rage → *Anger*, Fury, Hate, Dislike, Resentment, Outrage, Wrath, Hostility, Bitterness, Ferocity, Loathing, Scorn, Spite, Vengefulness, Envy → *Jealousy*, Disgust → *Revulsion*, Contempt, Loathing, Torment, Disapproval † |
| **Love** | Affection → *Liking*, Caring, Compassion, Fondness, Affection, Attraction, Tenderness, Sentimentality, Adoration, Lust → *Desire*, Passion, Infatuation, Longing, Gratitude ‡ |
| **Fear** | Horror → *Alarm*, Fright, Panic, Terror, *Fear*, Hysteria, Shock, Mortification, Nervousness → *Anxiety*, Distress, Worry, Uneasiness, Tenseness, Apprehension, Dread |
| **Joy** | Cheerfulness → *Happiness*, Amusement, Satisfaction, Bliss, Gaiety, Glee, Jolliness, Joviality, *Joy*, Delight, Enjoyment, Gladness, Jubilation, Elation, Ecstasy, Euphoria, Zest → *Enthusiasm*, Excitement, Thrill, Zeal, Exhilaration, Contentment → *Pleasure*, Optimism → *Eagerness*, Hope, Pride → *Triumph*, Enthrallment → *Enthrallment*, Rapture, Relief, Approval †, Admiration † |
| **Sadness** | Suffering → Hurt, Anguish, Agony, Sadness → Depression, Sorrow, Despair, Gloom, Hopelessness, Glumness, Unhappiness, *Grief*, Woe, Misery, Melancholy, Disappointment → Displeasure, Dismay, Shame → Guilt, Regret, Remorse, Neglect → Embarrassment, Insecurity, Insult, Rejection, Alienation, Isolation, Loneliness, Homesickness, Defeat, Dejection, Humiliation, Sympathy → Pity |
| **Surprise** | Surprise → *Amazement*, Astonishment, Confusion †, Curiosity †, Realization † |

Table 2. Extended Shaver's tree-structured taxonomy.

**Notes:** Emotions in blue appear in the list of emotions proposed by GoEmotions. Emotions added by Imran et al. from GoEmotions' list onto Shaver's taxonomy are denoted with †. A single emotion — *Gratitude* — is added to the taxonomy by this paper, denoted by ‡.