# Transformer-Based Emotion Analysis: Classification, Intensity Estimation, and Text Generation

**Zehan Li**
Department of Cognitive Science
University of California, San Diego
La Jolla, CA 92092
zel025@ucsd.edu

**Yishan Cai**
Halıcıoğlu Data Science Institute
University of California, San Diego
La Jolla, CA 92092
yic075@ucsd.edu

**Xinyu Hu**
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92092
xih035@ucsd.edu

## Abstract

This report presents a comprehensive emotion processing pipeline that combines three interconnected tasks: emotion classification, intensity estimation, and emotion transfer in text. We implement a BERT-based classifier to categorize text into six distinct emotions (anger, caring, confusion, fear, joy, and sadness), achieving 98.6% validation accuracy. Building upon this, we develop a regression model for quantifying emotion intensity on a normalized scale, followed by two approaches for emotion transfer: representational engineering and T5-based text generation. The representational engineering approach manipulates hidden states to control emotional expression, while the T5 model generates text with specified emotion intensities. Our integrated pipeline demonstrates effective end-to-end emotion processing, enabling both analysis and modification of emotional content in text. Experimental results show successful emotion transformation while maintaining semantic coherence, with potential applications in empathetic AI systems and emotional content generation.

Code: `https://github.com/YasuharaSky/LIGN167_Emotion_Transfer`

## 1 Introduction

Emotions are a fundamental aspect of human communication, deeply influencing interactions and decision-making processes. However, accurately identifying and interpreting emotions in text remains a challenging problem due to the complexity of human language and the subtlety of emotional expression. Recent advancements in representation engineering (Zou et al., 2023) and attribute intensity control in text generation (Zhou et al., 2024) have opened new possibilities for understanding and manipulating emotional content in text. Building on these developments, we address three interconnected challenges: first, how to classify the emotions conveyed in text accurately; second, how to estimate the intensity of these emotions to gain deeper insights into their contextual significance; and third, how to transform text to express the same emotion at different intensity levels while maintaining semantic coherence. This comprehensive approach to emotion processing has numerous applications, including sentiment analysis, mental health monitoring, personalized communication systems, and the development of more empathetic AI communication systems.

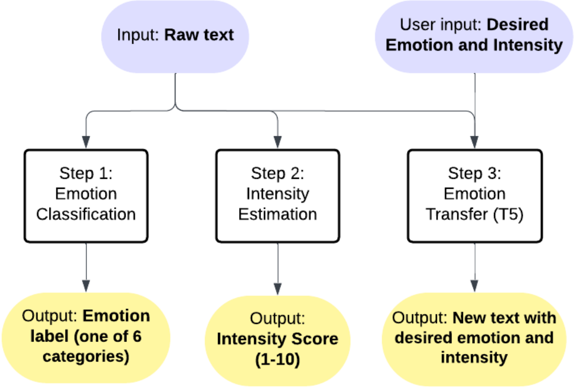
Input: Raw text
User input: Desired Emotion and Intensity
Step 1: Emotion Classification
Step 2: Intensity Estimation
Step 3: Emotion Transfer (T5)
Output: Emotion label (one of 6 categories)
Output: Intensity Score (1-10)
Output: New text with desired emotion and intensity

Figure 1: Emotion Processing and Transfer Pipeline

## 2 Datasets

To address these challenges, we created a comprehensive dataset designed specifically for training and evaluating machine learning models for emotion classification and intensity estimation. Key details about the dataset are as follows:

- **Emotions:** The dataset includes six core emotions: joy, sadness, fear, anger, caring, and confusion.
- **Structure:** For each emotion, the dataset contains over 50 scenarios. Each scenario includes 10 sentences, reflecting a gradual increase in emotional intensity from mild (1) to extreme (10).
- **Source:** Sentences were generated using ChatGPT and manually refined to ensure contextual accuracy and consistency. This approach allowed for high-quality and diverse data generation.
- **Training Points:** Each sentence in the dataset serves as a single training point, with a total of over 3,000 sentences across all emotions. Each training point consists of:
  - **Scenario:** The context or situation that anchors the emotion (e.g., *I got an A on an exam I studied hard for*).
  - **Text:** The specific sentence expressing the emotion (e.g., *I felt like I was on top of the world — this was the best moment of the week!*).
  - **Emotion Label:** The categorical emotion associated with the text (e.g., *Joy*).
  - **Intensity Score:** A numerical value between 1 and 10 indicating the intensity of the emotion (e.g., *10*).

This structured design enables machine learning models to learn not only to classify emotions but also to capture fine-grained variations in emotional intensity. By providing a large number of training points with diverse emotional contexts, the dataset supports robust model training and evaluation for both classification and regression tasks.

## 3 Model

Our project develops three consecutive models to achieve comprehensive emotion processing: emotion classification, intensity estimation, and emotion transfer. We first build a BERT-based classifier to identify six distinct emotion categories from text input. Then, we develop a regression model to quantify emotion intensity on a normalized scale. Finally, we explore two approaches for emotion transfer and generation: one using representational engineering and another using the T5 architecture. These three models are integrated into a unified pipeline that enables end-to-end emotion analysis and text modification.

### 3.1 Emotion classification

The primary objective of this model is to classify text into six distinct emotional categories: anger, caring, confusion, fear, joy, and sadness. By employing the BERT (Bidirectional Encoder Representations from Transformers) architecture, the model aims to understand and categorize the emotional content conveyed in textual input, providing a foundation for emotion-aware natural language processing applications.

The model architecture is built upon the pre-trained BERT-base model, which consists of 12 transformer layers with bidirectional self-attention mechanisms. The input text is first tokenized into subword units and processed through BERT's embedding layer, which combines token embeddings $E_{tok}$, position embeddings $E_{pos}$, and segment embeddings $E_{seg}$ to form the initial representation:

$$E(x) = E_{tok} + E_{pos} + E_{seg}$$

This embedded representation captures both the semantic meaning of words and their contextual positions within the input sequence.

Each transformer layer in the model processes the input through a self-attention mechanism, where the attention weights are computed as

$$A(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

where $Q$, $K$, and $V$ are the query, key, and value matrices derived from the input representations. This mechanism enables the model to capture complex dependencies between words, which is crucial for understanding the emotional context of the text.

For classification, the model employs a task-specific classification head that takes the pooled output corresponding to the [CLS] token and projects it into a 6-dimensional space, one for each emotion class. The probability distribution over emotion classes is computed using the softmax function:

$$p(y|x) = \text{softmax}(W_c h^L + b_c)$$

where $h^L$ is the final hidden state of the [CLS] token, and $W_c$ and $b_c$ are learnable parameters.

The model is trained using cross-entropy loss:

$$\mathcal{L} = -\sum_{i=1}^{N} \sum_{c=1}^{6} y_{i,c} \log(p(y_i = c|x_i))$$

where $y_{i,c}$ is the ground truth label for class $c$ of sample $i$. The AdamW optimizer is used to update model parameters.

### 3.2 Emotion Intensity Estimation

The second phase of our project focuses on **emotion intensity estimation**, which quantifies the degree or strength of emotion expressed in text on a normalized scale from 0 to 1. This regression task complements the earlier classification model by providing a more granular understanding of emotional expression in textual data.

Our model employs a BERT-based architecture, adapted for regression tasks. The original BERT model is modified by replacing its classification head with a regression layer. Specifically, the model processes input text through BERT's transformer layers to obtain contextual representations, followed by a linear regression layer that outputs a single continuous value representing emotion intensity. The mathematical formulation can be expressed as:

$$f(x) = W_r h(x) + b_r$$

where $h(x)$ is BERT's pooled output representation and $W_r, b_r$ are the regression layer parameters.

To standardize the intensity labels, the training process normalizes them to the range [0,1] by dividing all values by the maximum intensity in the dataset. The model is optimized using the Mean Squared Error (MSE) loss function:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

where $y_i$ is the true normalized intensity and $\hat{y}_i$ is the predicted intensity. The AdamW optimizer with a learning rate of $5 \times 10^{-5}$ is used for parameter updates.

### 3.3 Emotion Transfer

With the foundation of emotion classification and intensity estimation, we next explore methods to modify text for desired emotional outcomes. The third phase of our project focuses on **emotion transfer and text generation**. We try to achieve this purpose via two methods: Representational Engineering and Text-to-Text Transfer Transformer.

#### 3.3.1 Representational Engineering (RepE)

In our project, we explore representational engineering techniques to achieve fine-grained control over emotional expression in text generation. Unlike traditional approaches that simply condition text generation on emotion labels, this model aims to understand and manipulate the underlying emotional representations within the language model's hidden states, allowing for more nuanced and controllable emotion transfer and generation.

The model architecture consists of two primary components: the **RepReader**, which analyzes and maps emotional representations, and the **RepController**, which manipulates these representations during text generation. The RepReader identifies emotion direction vectors within the hidden state space by analyzing texts with varying emotion intensities. For a given emotion $e$ and intensity $i$, the centroid of hidden states is calculated as:

$$C_{e,i} = \frac{1}{|T_{e,i}|} \sum_{t \in T_{e,i}} h(t)$$

where $T_{e,i}$ represents the set of texts with emotion $e$ and intensity $i$, and $h(t)$ is the hidden state representation of text $t$. The emotion direction vector is then computed as the normalized difference between high and low intensity centroids:

$$v_e = \frac{C_{e,high} - C_{e,low}}{||C_{e,high} - C_{e,low}||}$$

The RepController implements the emotion manipulation process through a series of carefully designed hooks into the language model's generation process. It modifies the hidden states during generation using the formula:

$$h'_l = h_l + \alpha s_l v_e$$

where $h_l$ is the original hidden state at layer $l$, $\alpha$ is the target intensity scaling factor, $s_l = e^{-l/L}$ is a layer-specific scaling factor (with $L$ being the total number of layers), and $v_e$ is the emotion direction vector. The exponential decay in the layer scaling ensures that modifications are more pronounced in earlier layers while preserving the model's learned language structure in deeper layers.

The model demonstrates effective control over emotional expression through two main operations: direct emotion generation and emotion transfer. For emotion transfer, the model employs a two-step process: first neutralizing the source emotion by generating with minimal intensity ($\alpha \approx 0$), then applying the target emotion with the desired intensity. This approach helps maintain text coherence while achieving the desired emotional transformation.

#### 3.3.2 Text-to-Text Transfer Transformer (T5)

Another model we explore is the T5 architecture, with the intention of generating emotionally adjusted text based on specified scenarios and desired emotion intensities. This model addresses the challenging task of not only understanding emotions in text, but also generating new text with controlled emotional content. Applications for this task include improving empathetic communication in chatbots and tailoring responses in mental health support systems.

Unlike representational engineering, which directly manipulates hidden states, the T5 model architecture implements an encoder-decoder framework, where both components utilize transformer layers with self-attention mechanisms. The input format is structured as: "adjust: [emotion] | intensity: [value] | scenario: [text]", where emotion is one of six categories, intensity ranges from 1-10, and the scenario describes the context. The mathematical formulation of the attention mechanism in T5 can be expressed as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

where $Q$, $K$, and $V$ are learned query, key, and value matrices, and $d_k$ is the dimension of the key vectors.

The model is trained using teacher forcing, where during training, the model receives the ground truth target text as input to the decoder. The loss function is the cross-entropy loss over the vocabulary distribution:

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{v=1}^{V} y_{i,v}\log(p(v|x_i))$$

where $y_{i,v}$ is the true token distribution and $p(v|x_i)$ is the predicted probability of vocabulary token $v$ given input $x_i$. The model architecture includes 6 encoder and decoder layers, each with a hidden dimension of 512 and 8 attention heads.

# 4 Results

Our research project is a multi-model integration process, designed as an end-to-end system. The ultimate goal is to input a sentence along with a target emotion intensity and generate an output sentence that is semantically similar to the input but reflects the specified emotion intensity.

To achieve this objective, we implemented a three-step approach and integrated these steps into a cohesive pipeline:

1. First, we employed an Emotion Classifier to extract the emotion of the input sentence.
2. Second, we utilized Emotion Intensity Estimation to determine the intensity of the emotion in the input sentence.
3. Finally, we combined the input sentence, its classified emotion (obtained from Step 1), and the target intensity to generate the final output sentence.

Throughout this process, we incorporated rigorous evaluations for each model. Additionally, for the final output sentences, we conducted human evaluations by inviting participants from the LIGN167 cohort to score the results based on quality and relevance.

## 4.1 Emotion classification

For the task of emotion classification, we utilized BERT as the backbone for training. The dataset comprised six distinct emotions, with over 80 scenarios created for each emotion. Each scenario included 10 sentences with varying levels of emotion intensity, resulting in a total of 4,000 samples.

After constructing the dataset, we explored improvements to the standard 768-dimensional BERT embeddings. Specifically, we experimented with two approaches:

1. Directly transforming sentences into embeddings.
2. Utilizing BERT's hidden embeddings as training data.

Our results indicated that training the model directly on raw sentences yielded the best performance. Using a Google Colab A100 GPU, the model achieved a validation accuracy of 98% after five epochs when trained on raw sentences. In contrast, the embedding-based approach resulted in a validation accuracy of approximately 95%, showcasing a clear advantage for the former methodology.

These findings highlight the importance of preserving the semantic integrity of raw sentences during training for optimal classification performance. Further analysis is needed to explore whether embedding modifications could yield improvements under different experimental setups.

## 4.2 Emotion Intensity Estimation

For the task of emotion intensity estimation, we utilized the same dataset as in the emotion classification step. BERT was again selected as the underlying model for this experiment.

During the training process, the intensity values in the dataset were used as the target outputs. A masking strategy was applied to train the model effectively. The experimental results aligned closely

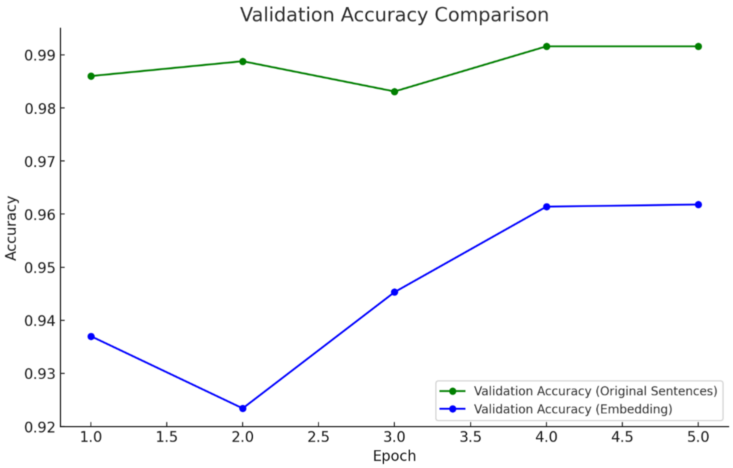

Figure 2: Result of emotion classifcations with sentence + embedding comparison

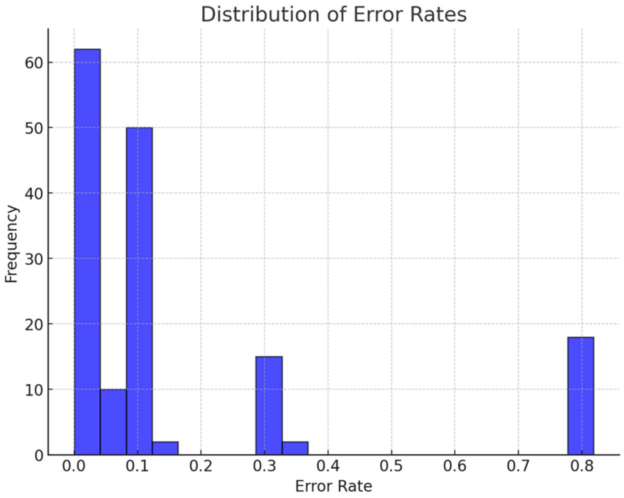

Figure 3: Result of emotion intensity error rates

with our expectations, demonstrating that the majority of the predictions had an error range between 0 and 0.1, as shown in Figure 3.

These results highlight the model's capability to predict emotion intensity with high precision, underscoring the efficacy of leveraging pre-trained language models like BERT for this task. Further refinement of the masking strategy or dataset augmentation could potentially improve performance in future experiments.

### 4.3 Emotion Transfer

The final step of the pipeline integrates the previously developed models into a third model for emotion transfer. In this phase, we experimented with two approaches: T5-based fine-tuning and Representation Engineering.

#### 4.3.1 Representation Engineering

Representation Engineering was explored as an initial approach; however, it did not yield satisfactory results. Despite efforts to reimplement and refine the method, the BLEU scores consistently fell below 30, indicating significant limitations in its effectiveness.

### 4.3.2 T5-Based Fine-Tuning

The T5-based model demonstrated a noticeable improvement over Representation Engineering. Generated sentences were evaluated by human annotators, including participants from the LIGN167 cohort, who assessed the outputs based on the following criteria:

- Alignment with the target emotion and intensity.
- Preservation of the semantic content of the original scenario.

### 4.3.3 Evaluation Results

Despite the improvements achieved with the T5 model, the results revealed persistent challenges. Many generated sentences struggled to align with the requested emotional adjustments. Instead of expressing the specified emotion (e.g., high-intensity joy, deep sadness, or strong empathy), the outputs often defaulted to expressions of apprehension, nervousness, or vague unease. This trend was observed across various input scenarios and requested emotion-intensity combinations.

For example, prompts designed to elicit intense joy or deep sadness frequently resulted in mild or unrelated emotional states. Similarly, requests for strong empathy often produced awkward or self-focused outputs rather than genuinely empathetic responses.

Human evaluations supported these observations. On a scale of 1 to 10 (where 10 indicates strong alignment between the requested emotion/intensity and the generated output), most scores ranged between 4 and 6. These ratings suggest that while the model occasionally produced sentences partially aligned with the target emotion and scenario, it rarely achieved the desired intensity. Many outputs appeared emotionally flat or mismatched, failing to convey the nuanced adjustments intended.

### 4.3.4 Challenges and Future Directions

These results underscore the difficulties in controlling emotional tone and intensity in text generation. While the pipeline successfully integrates emotion classification, intensity estimation, and controlled generation, the final stage requires further refinement. Key areas for improvement include:

1. Enhancing the generation model's ability to internalize and accurately express the specified emotion and intensity.
2. Maintaining semantic consistency between the input and output sentences while applying the desired emotional adjustments.
3. Producing outputs that feel natural, contextually appropriate, and emotionally resonant.

By addressing these challenges, future efforts aim to improve the emotional fidelity and coherence of generated sentences, bringing the system closer to achieving consistent and controllable emotional intensity transformations.

## 5 Conclusion

This work explored three models for emotion-related natural language tasks: sentiment analysis, emotion intensity estimation, and emotion intensity transfer. While the sentiment analysis model demonstrated high accuracy (98.6%) and effectively identified emotions in sentences, the emotion intensity regressor exhibited promising results with reasonable predictions compared to actual intensity values. However, our experiments with emotion intensity transfer revealed significant challenges.

The first method, leveraging T5-based transformers, generated sentences that maintained emotional alignment but occasionally failed to meet the desired intensity targets. The second approach, Representation Engineering (RepE), faced notable limitations, primarily due to dataset constraints and the reliance on contextually learned data from large language models (LLMs), which lack nuanced human-like intensity modulation. Additionally, issues such as the lack of a precise intensity definition and the inherent complexity of natural language further impacted performance.

Overall, this study highlights the potential and limitations of current NLP techniques in emotion-related tasks. While sentiment analysis and intensity estimation models showed strong capabilities, further advancements in data quality, model architecture, and representation understanding are essential to achieve seamless emotion intensity transfer. Future work should prioritize refining intensity definitions, leveraging larger and more diverse datasets, and developing models capable of producing smoother intensity adjustments aligned with human-like language behavior.

## References

**Zhou, Shang, Feng Yao, Chengyu Dong, Zihan Wang, and Jingbo Shang.** 2024. “Evaluating the Smooth Control of Attribute Intensity in Text Generation with LLMs.” [Link]

**Zou, Andy, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks.** 2023. “Representation Engineering: A Top-Down Approach to AI Transparency.” [Link]