

# Enhanced Text Communication System Using Sentiment-Based Prompting for End-User

## 1 Abstract

Today’s chat applications are built around the same building blocks which were used when they were emerging - one-on-one chats, groups, multiple modalities (text, audio, video, emojis, etc.), and any new chat application has pretty much the same functionality. On the other hand, more and more people are communicating over text than any other form of communication. But text has one problem - the tone (or body) of the conversation is not always apparent, and may require further clarification. It may eventually even lead to misunderstandings while communicating. This happens so often that multiple modalities are then equipped in order to clarify, elaborate or communicate the same sentiment. This paper presents an application to enhance our texting experience, and add another dimension to communication through this medium. The chat application presented in this paper does the following:

- Understanding conversation sentiment in real-time
- Provide live analysis about the sentiment of parts of the conversation or last (immediate) text
- Based on the sentiment understanding and context of the conversation, provide suggestions/prompts to the user to respond appropriately, or tailor it before sending

This application aims to solidify communication through text seamlessly by improving user understanding and judgment, while retaining the existing methods.

## 2 Introduction

**Problem Definition:** Despite the widespread use of chat applications in both personal and professional communication, current platforms lack the ability to interpret or convey the emotional tone behind textual messages. As a result, users often experience miscommunication, misunderstandings, or emotional ambiguity, especially in sensitive conversations. This limitation is inherent to text as a medium, which lacks non-verbal cues such as tone, facial expressions, or gestures. This project aims to address this gap by developing an intelligent chat interface that can analyze the sentiment of messages in real-time and offer contextually appropriate prompt suggestions to the user before they respond. The system enhances communication clarity and helps users craft more empathetic, constructive replies by leveraging Natural Language Processing (NLP) and Large Language Models (LLMs).

**Problem Analysis:** Text-based communication suffers from the inability to transmit non-verbal emotional signals, leading to a disconnect between intent and interpretation. Even neutral-sounding

messages can be misunderstood as passive-aggressive, rude, or dismissive, especially when the reader is under stress or in a professional setting.

Attempts to solve this (like emojis, reactions, or voice notes) only partially bridge the gap and are often not suitable for formal contexts. While sentiment analysis models exist, they are rarely integrated into messaging platforms in a live and actionable form.

Furthermore, users may not be aware of how their messages might be perceived emotionally. There is a need for a system that can detect emotional tone, and based on both sentiment and conversation context, provide real-time suggestions for rephrasing or responding appropriately. Such a system must work efficiently and unobtrusively, fitting naturally into the user’s workflow.

## 3 Methodology / System Architecture

**Architecture:**

Figure 1. System Architecture

### Components (Step 4 – Implementation):

- **Preprocessing:** Cleaning and tokenizing messages
- **Sentiment Classification:** Using pre-trained BERT
- **Regression Layer:** For fine-grained sentiment scoring
- **Prompt Generator:** Using LLM APIs (OpenAI, Falcon)
- **Frontend/Backend:** Built with Flask, integrated with HuggingFace models

## 4 Experiments and Evaluation (Step 5)

**Dataset Used:** Enter text dataset name

**Preprocessing:** Standard NLP preprocessing steps applied  
**Metrics:**

- Classification: Accuracy, F1-score
- Regression: MSE, RMSE,  $R^2$
- Prompt Evaluation: BLEU score

| Model           | metric1 | metric2 | metric3 |
|-----------------|---------|---------|---------|
| Baseline (BERT) | 0.00    | 0.00    | 0.00    |
| Our Model       | 0.00    | 0.00    | 0.00    |

Table 1. Evaluation Results

## 5 Applications and Conclusion

**Applications:** Corporate communication, education platforms, therapy bots

**Limitations:** Handling sarcasm, multilingual support, LLM latency

**Future Work:** Personalization, lightweight deployment, on-device models

References

References

- [1] Reference 1 detail .
- [2] Reference 2 detail .
- [3] Reference 3 detail .

Team Contributions

| Team Member        | Contribution Area |
|--------------------|-------------------|
| Deepshikhar Tyagi  |                   |
| Ishwer Kumar       |                   |
| Manish Kumar Singh |                   |
| Rishabh Mehrotra   |                   |
| Sanket Jain        |                   |
| Shubham Saha       |                   |
| Yuvashree Pamujula |                   |

Appendix

Architecture in Detail

Insert expanded architecture description and figure here.

Sample Chat Screenshots

Include sample screenshots of chat UI with prompt suggestions.

Hyperparameters and Prompt Templates

Describe model settings and sample LLM prompt templates.

GitHub

Link to full source code and demo README.