**Project Proposal: Emotion-Aware Chatbot Using Transformer-Based Sentiment Analysis**

Group 1

Team Member**:**

Ana Luiza Young Pessoa,

Lisha Tu,

Ji Weng,

Min Yao

# 1. Final Topic Area: NLP

This project focuses on the field of Natural Language Processing (NLP), specifically in building an emotion-aware conversational system. We chose the NLP domain because the course has provided in-depth coverage of Transformer-based models applied to text sentiment analysis, which aligns well with our team’s interests and technical background. Moreover, emotion recognition is a challenging and impactful task within NLP, allowing us to explore the real-world potential of generative AI techniques.

## 1.1. Introduction

In today’s customer service industries, understanding user emotions in real time is critical for delivering personalized and empathetic interactions. This project proposes an emotion-aware chatbot that leverages a Transformer-based machine learning model to analyze conversational text, predict emotional states (for example, joy, frustration, confusion), and adapt responses dynamically. In addition to a traditional sentiment analysis system that classifies text as simply "positive" or "negative," this chatbot will categorize emotions into nuanced classes, enabling more human-like and context-aware dialogue.

This chatbot aims to solve challenges of current chatbot systems:

* Capture nuanced emotions: Binary sentiment labels (positive/negative) overlook critical emotional states (for example, sarcasm, urgency).
* Contextualize conversations: Short-term interactions lack memory of emotional context across dialogue turns.
* Adapt dynamically: Rule-based systems cannot generalize to unseen emotional expressions or multilingual inputs.

An example scenario: A customer writes, “Great, another delayed shipment!” (sarcasm). Traditional systems might misclassify this as “positive,” while our model is expected to detect “frustration.” This project targets on developing a chatbot with enhanced output capability (the granular level of identified emotions) and adaptability (the capability of emotion detection in complex contextual scenarios).

## 1.2. Theoretical Background

According to a literature review of 83 articles from 2019 to 2022 by Gamage et al. (2024), emotion detection approaches can be categorized into three key types: heuristic methods, AI-based methods, and hybrid methods.

* **Heuristic Methods**: Heuristic methods rely on keyword recognition, rule-based logical structures, and statistical techniques ground in emotional lexicons and corpora. Heuristic methods utilize predefined lists of emotional terms, organized hierarchically or graphically, to identify emotions in text. Keyword recognition searches keywords with emotion labels to detect explicit emotions but struggle with implicit or ambiguous expressions. Rule-based methods utilize text process techniques such as tokenization, part-of-speech tagging, and dependency parsing, which enhance the detection process.  However, the major limitation is that they may miss nuanced emotional expressions not included in the lexicon.
* **AI-Based Methods**: AI-based methods consist of conventional supervised learning and contemporary transfer learning approaches. Conventional methods require large, labelled datasets to train models using algorithms like Support Vector Machines and deep learning techniques (e.g., LSTMs, GRUs). Transfer learning methods leverage pre-trained contextual language models (like BERT and GPT) to fine-tune emotion detection with smaller datasets and higher accuracies. Despite their effectiveness, these methods face challenges with generalizability and the interpretation of ambiguous emotional expressions.
* **Hybrid Methods**: Hybrid methods combine elements of both heuristic and AI-based approaches, aiming to enhance accuracy and refine emotion categorization. Examples include using lexicon-based emotion annotation to train classifiers or integrating rule-based semantics with lexicon ontology. These methods leverage the strengths of both categories to improve emotion detection in various contexts, such as social media analysis and healthcare.

In this project, the transfer learning method is utilized for emotion detection. Transfer learning is a machine learning technique that incorporates a pre-training model and fine-tuning it for a different and related task. The pre-trained models, such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), have been trained on vast lexicons and corpora. These models are then fine-tuned on a smaller dataset specific to the emotion detection task.

The main advantages of the transfer learning model are:

* It reduces the need for large, labelled datasets required for training, which also reduces the required computing power and time during this limited project duration.
* Fine-tuned models from a pre-trained model often achieve higher accuracy compared to those trained directly on smaller and domain-specific datasets.

However, the transfer learning method still faces several challenges, including the requirement for high-quality labeled data for fine-tuning, limited generalizability across domains and contexts.

# 2. Dataset Selection

This project plans to use two datasets for the fine-tuning task:

**(1) Primary Dataset: MELD (Multimodal Emotion Lines Dataset)**

(https://github.com/declare-lab/MELD)

MELD is an enhanced version of the Emotion Lines dataset, proposed by Chen, S.Y., Hsu, C.C., Kuo, C.C. and Ku, L.W., extended to include text, audio, and visual modalities for each utterance, making it a multimodal, multi-party conversational emotion recognition dataset. It contains more than 1400 dialogues and 13,000 dialogue utterances from the TV series Friends. Each utterance in a dialogue has been annotated with 7 emotions: Anger, Disgust, Sadness, Joy, Neutral, Surprise and Fear. The dataset includes multi-speaker conversations, ideal for context modeling and emotion recognition.

**(2) Secondary Datasets**: **Customer Support on Twitter Dataset**

(https://www.kaggle.com/datasets/thoughtvector/customer-support-on-twitter)

The dataset is a large corpus of consumer tweets and companies’ responses. The dataset provides materials for natural language and conversational models understanding and analysing. It records real consumer requests to specific problems. The brevity of tweets can enhance the training model’s ability to analyze emotions from low message size, while the low message size is convenient for neural networks training.

# 

# 3. Model Selection

We plan to use a Transformer-based architecture for emotion classification. Specifically, we will fine-tune a pretrained DistilBERT model from the Hugging Face library due to its computational efficiency and strong performance on NLP tasks. DistilBERT retains over 95% of BERT's performance while being significantly smaller and faster, which makes it ideal for our limited timeframe and resources. Fine-tuning a pretrained model allows us to leverage learned language representations and improve accuracy without requiring massive labeled datasets. This approach aligns with best practices in modern NLP projects.

# 4. Research Questions

This project aims to investigate the following research questions:

* Can a fine-tuned Transformer model (e.g., DistilBERT) outperform traditional models (e.g., LSTM) in multi-class emotion classification tasks?
* How effectively can the model detect complex emotional expressions such as sarcasm or mixed sentiment in short text conversations?
* To what extent does adding multimodal features (e.g., audio) improve the model's ability to classify emotions accurately?
* These questions are specific, feasible within the course timeline, and grounded in key generative AI topics discussed in class, such as transfer learning, contextual modeling, and representation learning.

# 5. Plan of Action

## 5.1. Technology Stack

The project involves the following steps for the chatbot training:

**(1) Data Preprocessing**

This audio/textual data is preprocessed first. Specific steps include data cleaning and segmentation, feature extraction, data augmentation, normalization, context processing, etc. Some techniques that might be used in each step are listed as follows:

| Step | Text Data | Audio Data |
| --- | --- | --- |
| Cleaning & Segmentation | Lowercasing, punctuation, stop-word removal | Noise reduction, voice activity detection |
| Feature Extraction | Tokenization, embeddings | MFCC, spectrogram, prosodic/pitch/energy features |
| Data Augmentation | Paraphrasing, back-translation | Add noise, pitch/time shift, speed perturbation |
| Normalization | Standardize/lemmatize words | Normalize loudness, equalize sample length |
| Context Processing | Conversation history/context window | Temporal windowing, acoustic context |
| Pre-trained Embeddings | BERT/Transformer word embeddings | wav2vec, VGGish, ECAPA-TDNN audio embeddings |

**(2) Setting up Transformer Architecture**

A pre-trained model – the Transformer – is set up in this step. Fine-tuning the transformer on emotion data leverages their deep semantic power. These models can be DistilBERT (efficient) or GPT-3-small (for generation).

One or more neural network layers are added on top of the transformer output, followed by an activation for emotion categories. Thus, the model can convert the rich language features from the transformer into probability scores for each emotion.

**(3) Fine-Tuning and Training**

TensorFlow/Keras or PyTorch are used for model training and deployment. These frameworks offer scalable training, GPU support, and easy deployment.

* Possible Tools
* Hugging Face Transformers for pretrained models.
* NLTK for text preprocessing (stopword removal, lemmatization).
* Streamlit for chatbot interface prototyping.
* Training Workflow
* Input: User queries are preprocessed and tokenized with special markers (e.g., [CLS], [SEP]).
* Model Forward Pass: The transformer encoder processes the input and produces embeddings. The emotion classifier turns these into a probability distribution over emotion classes.
* Loss Calculation: Use cross-entropy loss for single-label or binary cross-entropy for multi-label emotions.
* Evaluation: Use metrics like accuracy, macro/micro-averaged F1-score, per-class F1, confusion matrix, and pay special attention to misclassifications in complex (sarcasm, negation) cases.
* Adaptation: Use the detected emotion to adapt chatbot replies (choose templates or generate text with a language model).
* Deployment: Export trained models (e.g., TensorFlow SavedModel, ONNX, or TorchScript) and integrate with the chatbot backend (API endpoint or in-app).
* Visualization and Evaluation
* To visualize model outcomes and evaluate performance, data visualization tools such as Matplotlib, Seaborn, and Plotly can be used for enhanced interpretability.
* Version Control and Development
* Git and GitHub are tools that will be used for code management, version control, and collaboration among team members.

## 5.2. Expected Outcomes

* Performance Metrics:
* Target 85% accuracy and F1-score >0.82 on MELD test set.
* Comparison baseline: LSTM-based model (expected 10–15% lower accuracy).
* Functional Chatbot: Real-time emotion detection with response adaptation, such as escalate to human agent if “anger” is detected.
* Scalability: Model deployable via TensorFlow Serving for low-latency inference.

## 5.3. Key milestones

| Phase | Tasks | Duration |
| --- | --- | --- |
| 1. Data Prep | Dataset collection, cleaning, tokenization | 2 weeks |
| 2. Model Dev | Fine-tuning Transformer, hyperparameter tuning | 3-4 weeks |
| 3. Integration | Chatbot API development (Python/Flask) | 2 weeks |
| 4. Testing | Accuracy validation, latency optimization | 2 weeks |

# 6. Team Contribution

Each team member is responsible for a distinct component of the project to ensure efficient collaboration and clear accountability:

* **Min Yao**: Data preprocessing (initial notebook) and model fine-tuning
* **Lisha Tu**: Data preprocessing and partial model development
* **Ana Luiza Young Pessoa**: Model integration and system testing
* **Ji Weng**: Model evaluation (e.g., accuracy, F1-score, confusion matrix) and result visualization and analysis

This task distribution reflects each member’s strengths and ensures steady progress toward the project goals. Responsibilities may still be adjusted or supported collaboratively as needed.

# 7. Summary

This project aims to bridge the gap between automated chatbots and human-like empathy by integrating Transformer-based emotion analysis. By leveraging contextual understanding and scalable deep learning, the proposed system will enhance customer satisfaction, reduce escalations, and set a new standard for emotionally intelligent AI.

# Reference

[1] G. Gamage, Daswin De Silva, N. Mills, Damminda Alahakoon, and Milos Manic, “Emotion AWARE: an Artificial Intelligence Framework for adaptable, robust, explainable, and multi-granular Emotion Analysis,” *Journal of Big Data*, vol. 11, no. 1, Jul. 2024, doi: <https://doi.org/10.1186/s40537-024-00953-2>.

[2] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, R. Mihalcea. MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversation. ACL 2019.

[3]Chen, S.Y., Hsu, C.C., Kuo, C.C. and Ku, L.W. EmotionLines: An Emotion Corpus of Multi-Party Conversations. arXiv preprint arXiv:1802.08379 (2018).