Analyzing , comparing, customizing and enhancing Emotion Recognition Models

Maryam Alipourhajiagha
Polytechnique Montréal

Amine Ouakib
Université de Montréal

Aleix Pagés
Université de Montréal

Abstract

In today's digital world, the ability of machines to understand human emotions in text is crucial. Emotional awareness enhances human-computer interactions, making them more engaging and empathetic. With the rise of text-based communication platforms, detecting emotions such as happiness, sadness, and anger in textual dialogues has become essential. In our research, we first compared different transformer-based models with different datasets to see their average performance metrics and then implemented SS-BED, a novel deep-learning model that combines sentiment and semantic analysis to improve emotion detection. Additionally, we developed a hybrid model combining machine learning (SVM) and deep learning (CNN, Bi-GRU). The objective of our work is twofold: to provide the open-source implementation and weights of these models, which are currently lacking, and to improve upon existing approaches. One key advantage of the hybrid models we implemented is their ease of deployment in environments where resources are limited. These models are well-suited for applications where memory and quick inference time are prioritized over achieving the highest possible accuracy. Such scenarios include mobile app agents and smart devices with low memory and RAM capacities, where rapid processing and efficient resource usage are essential.

The code are available at the following repositories[1].

1 Introduction

The remarkable progress of natural language processing (NLP) in recent years, fueled by artificial intelligence, is widely recognized. However, there is still a lot of work to do in order to face some challenges in the domain, such as linguistic ambiguity, contextual understanding, or figurative language, among others. In addition, the use of NLP as a tool for the detection of emotions from text using NLP adds a layer of complexity.

In addition, there are other challenges to take into account like (Plaza-del Arco et al., 2024) the absence of demographic and cultural aspects, the poor fit of emotion categories, the lack of a common systematic nomenclature, or the dearth of interdisciplinary research (De Bruyne, 2023) .

Throughout this project, we conducted a comprehensive study of various deep learning models, such as BERT, RoBERTa, ELECTRA, and fine-tuned GPT-2, applied to multiple datasets (ISEAR, Diar AI, and GoEmotions) for text emotion detection. Our analysis showed that BERT, and transformer-based models in general, consistently outperformed the others. During our experiments, we encountered two papers that presented hybrid approaches claiming to achieve promising results. The first paper, (et al., 2019), utilized an LSTM-based approach, while the second paper, (et al., 2022), focused on an GRU-based model. To contribute to the community, we replicated their proposed models in PyTorch from scratch, as no open-source implementation was available.

In the first paper, the authors used a large dataset consisting of 17.62 million tweet conversation pairs extracted from the Twitter Firehose, spanning from 2012 to 2015. This data, along with corresponding responses, served as the foundation for training their model. While this dataset was not publicly released, we used the SemEval 2019 Task 3 dataset, which follows the same structure, for our experiments.

In the second paper, the authors employed a hybrid

[1] SS-BED Emotion Classifier, BiGRU Hybrid Model and Transformer-Based Models.

dataset combining three sources: ISEAR, WASSA, and Emotional-Stimulus, each containing text labeled with corresponding emotions. These datasets encompass three distinct text types: standard sentences, social media tweets, and conversational dialogues. A key advantage of this hybrid approach is its versatility in processing diverse textual inputs, including full sentences, tweets, dialogues, emotion-related keywords, and lexicon-based emotional expressions.

The EmoContext dataset is strongly grounded in Ekman's theory of six basic emotions: *happiness*, *sadness*, *anger*, *fear*, *surprise*, and *disgust*. However, it explicitly categorizes only three of these: *happy*, *sad*, and *angry*, while the remaining emotions are grouped under an *others* label. This design choice reflects the relative prominence of these three emotions in social media interactions, as emotions such as *disgust* and *shame* occur less frequently in such contexts. Additionally, the dataset places particular emphasis on the role of *emoticons* as key indicators for emotion detection. This focus further strengthens its connection to Ekman's framework (**?**), which associates specific facial expressions with corresponding emotional states. By leveraging emoticons as proxies for facial cues, the dataset aligns with Ekman's theory while adapting it to the domain of digital communication.

The hybrid dataset (combining ISEAR, WASSA, and Emotion-Stimulus) used in the GRU-based model is theoretically grounded in both Ekman's basic emotions framework and Lazarus' Appraisal Theory(Lazarus, 1991). This alignment is particularly relevant since ISEAR and Emotion-Stimulus explicitly incorporate the causal antecedents of the labeled emotions. Furthermore, the dataset addresses cultural variability through ISEAR's annotations, which span 37 countries across all continents, ensuring broad cross-cultural representation

The motivation behind our approach is rooted in the fact that, while transformer-based models excel in most tasks, including vision tasks, they are computationally expensive, especially in terms of sequence length, as their complexity grows quadratically. In contrast, RNN-based models (such as LSTM and GRU) exhibit linear complexity with respect to sequence length. Despite the impressive performance of transformers, we believe there is still value in exploring architectures like RNNs and identifying areas where they can be improved, offering a balance between efficiency and effectiveness.

2 Related Work

Emotion detection from text has gained substantial attention in recent years, with researchers exploring various methods to improve accuracy and interpretability. Traditional sentiment analysis models primarily relied on lexicons and statistical approaches, such as the NRC Emotion Lexicon (Mohammad and Turney, 2010) and SentiWordNet (Baccianella et al., 2010), which mapped words to predefined emotional categories. However, these models struggled with contextual nuances and implicit emotional expressions. The emergence of deep learning and transformers revolutionized the field by introducing models capable of learning complex semantic and syntactic patterns from large-scale text data (Cortiz, 2021).

Several works have integrated psychological theories into deep learning architectures to enhance emotion classification. The EmoAtlas framework , for instance, merges artificial intelligence with network science and psychological lexicons to model emotional relationships more effectively (Semeraro et al., 2025). Another significant development is ECR-BERT , which incorporates cognitive appraisal theories, particularly the OCC model, into BERT-based architectures, enabling improved explainability and reasoning behind emotional predictions (Wan et al., 2024). Additionally, in (Li et al., 2024), authors highlighted the importance of transparency in emotion detection systems. These studies demonstrate that integrating cognitive-emotional models into NLP can enhance both performance and interpretability.

Despite these advancements, several gaps remain. Existing works primarily focus on a single psychological model, such as OCC or Plutchik's wheel of emotions, without systematically comparing their effectiveness within different NLP frameworks. Additionally, while some studies incorporate external knowledge sources like emotion lexicons, they do not explore how different preprocessing techniques influence model performance. This project aims to address these gaps by systematically evaluating multiple emotion-cognitive models within diverse deep learning architectures, assessing their effectiveness

	turn1	turn2	turn3	label
0	don't worry i'm girl	hmm how do i know if you are	what's your name ?	0
1	when did i ?	saw many times i think - -	no . i never saw you	3
2	by	by google chrome	where you live	0

Figure 1: EmoContext Dataset Samples

across various components of a framework.

3 Datasets

3.1 Transformer Models

The GoEmotions dataset, developed by Google in 2020, consists of 58,000 Reddit comments labeled with 27 fine-grained emotions. (Demszky et al., 2020). The SemEval 2019 EmoContext dataset comprises 30,000 annotated dialogues labeled with emotions such as happy, angry, sad, and others. (Huang et al., 2019). The ISEAR (International Survey on Emotion Antecedents and Reactions) dataset, featuring 6,027 records from 37 countries, includes labels 7 classes that is one of the few datasets rooted in psychology (ISE, n.d.). The preprocessing steps we did were text cleaning by removing special characters, URLs, and punctuation, followed by tokenization and lowercasing for uniformity. Stopword removal will eliminate irrelevant words. To address the class imbalance, data augmentation and oversampling are done. The analysis we did on the ISEAR, Diar AI, and GoEmotions datasets shows that the ISEAR is balanced in terms of emotion frequency and text length however the other two are not balanced in terms of emotion frequency. Look at **??**, **??**, **??** in Appendix.

3.2 SS-BED Model

To evaluate conversational context on the SS-BED model, we use the SemEval 2019 Task 3 (EmoContext) dataset (Chatterjee et al., 2019), which consists of textual dialogues extracted from user interactions with a conversational agent (Figure 1). Each dialogue is represented as a tuple of three turns:

- **User Turn-1:** The first utterance of the user.
- **Conversational Agent Turn-1:** The agent's response.
- **User Turn-2:** The second utterance of the user, responding to the agent.

The task is formulated as a four-class emotion classification problem, using the following categories:

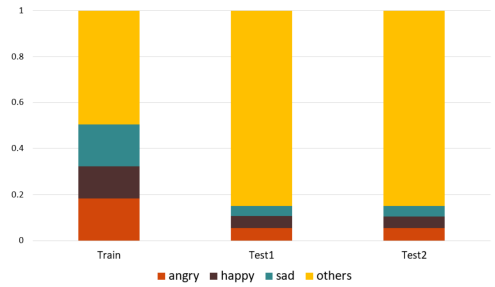

Figure 2: EmoContext Comparison of Class Distribution

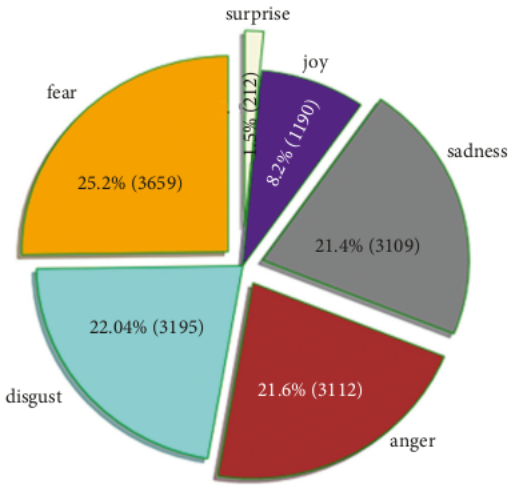

Figure 3: Bi-GRU+CNN Combined Dataset Distribution

"Happy", "Sad", "Angry", and "Others". The dataset is divided into a training set of 30,160 dialogues, and two test sets containing 2755 and 5509 dialogues, respectively (Figure 2).

Additionally, the dataset places particular emphasis on the role of *emoticons* as key indicators for emotion detection. This focus further strengthens its connection to Ekman's framework (Ekman, 1992), which associates specific facial expressions with corresponding emotional states. By leveraging emoticons as proxies for facial cues, the dataset aligns with Ekman's theory while adapting it to the domain of digital communication.

3.3 Bi-GRU+CNN Hybrid Model

For implementing the hybrid CNN+BiGRU-based model, we pre-processed the same combination of datasets used in the reference paper on which our model architecture is based Bharti(et al., 2022).

To carry out this task we developed two different

datasets in order to compare them :

- **Dataset 1:** We combined the same three datasets than the authors (ISEAR, WASSA, and Emotion-Stimulus) and then randomly selected the same number of samples per category as reported by the authors of the original paper. The result is shown in Figure 3. A total of 14513 sentence distributed on T six categories : "fear," "surprise," "joy," "sadness," "disgust," and "anger," which correspond to the classification proposed by Ekman. In the combined dataset, it can be observed that the "neutral" category was not included. Additionally, we can see that the "surprise" and "joy" categories are significantly less frequent. This is solely due to the distribution resulting from the combination of the three datasets.
- **Dataset 2:** We combined three established datasets—ISEAR, Emotion-Stimulus, and DAIR-AI—to create our second dataset. This selection involved replacing WASSA with DAIR-AI for two principal reasons. First, while both contain English Twitter data, DAIR-AI provides substantially more comprehensive coverage with 42,000 tweets compared to WASSA's 7,000. Second, and more crucially, DAIR-AI aligns perfectly with ISEAR and Emotion-Stimulus by employing Ekman's six basic emotion categories, whereas WASSA uses only four core emotions (anger, fear, joy, and sadness). This substitution enhances the consistency and coherence of our experimental framework. As a result, the combination of categories can be seen in Figure 4.

Finally, the datasets were split into training and test sets using an 80:20 ratio.

4 Methods

4.1 Transformer Models

SOTA models Evaluated on Diar AI, GoEmotions, ISEAR results are reported here. See the results of experiments on github here. The models are Naive Bayes, BERT (base-uncased-emotion), DistilBERT (base-uncased-emotion), GPT-2 (emotion model), RoBERTa (base-emotion), ELECTRA (emotion model). Transformer models, such as BERT and RoBERTa, have demonstrated strong performance in NLP tasks due to their ability to capture contextual dependencies (Adoma et al., 2020).

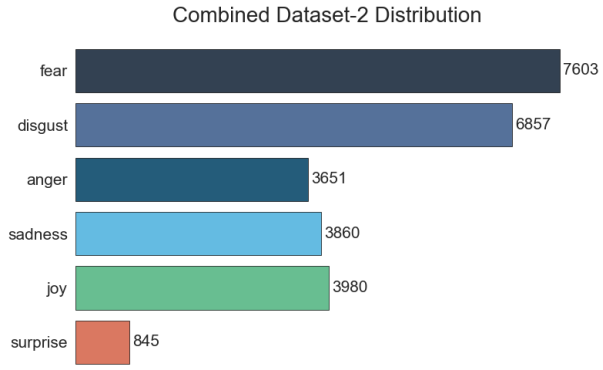

Figure 4: Bi-GRU+CNN Combined Dataset-2 Distribution

Model	Test Acc	Macro F1
Naïve Bayes	0.8385	0.7600
BERT	0.9265	0.8800
ELECTRA	**0.9340**	**0.8900**
RoBERTa	0.9275	0.8800
DistilBERT	0.9270	0.8800

Table 1: Comparison of Model Performance on Dair AI

Model	GoEmotions F1	ISEAR F1
BERT	0.46	0.76
RoBERTa	0.74	0.7431
DistilBERT	0.47	0.6693
GPT-2	0.60	0.77

Table 2: Model Performance on GoEmotions and ISEAR Datasets

RoBERTa and fine-tuned GPT-2 demonstrate particularly strong results. BERT-based models, especially when combined with psycholinguistic features, also show competitive performance. DistilBERT offers a lightweight alternative with slightly lower performance but still valuable in resource-constrained scenarios.

Incorporating emotion lexicons into ELECTRA through Knowledge-Embedded Attention (KEA) improved performance in distinguishing closely related

emotions, outperforming previous models.

In a study (Suresh and Ong, 2021) comparing various models, GPT-2 achieved a macro F1 score of 0.52 on GoEmotions, suggesting that while it performs well, it may not surpass other transformer models like BERT and RoBERTa on the GoEmotions dataset, which is aligned with our findings.

4.2 Sentiment and Semantic Based Emotion Detector Model - SS-BED

The proposed model SS-BED which you can see its architecture in Figure 5, is designed to detect emotions in textual dialogues. What sets SS-BED apart is its ability to combine **semantic** and **sentiment** encodings for emotion classification. This dual encoding allows the model to better understand the underlying emotion of a sentence, even when the sentiment conveyed by individual words might be ambiguous.

- **Semantic Encoding**: This layer captures the meaning of the words in the context of the entire sentence, helping the model understand the deeper intent behind the words.
- **Sentiment Encoding**: The sentiment layer, using Sentiment-Specific Word Embeddings (SSWE), focuses on the emotional polarity of individual words. This helps the model detect emotional nuances like irony or conflicting sentiments within a sentence.

The model uses two LSTM layers to process these encodings separately. The semantic representation uses GloVe embeddings, which are pre-trained to capture word meanings based on their co-occurrence in large corpora. GloVe is particularly good at modeling word similarity and meaning in context. On the other hand, the sentiment representation uses SSWE, a specialized embedding technique that incorporates sentiment-specific information, making it more sensitive to the emotional tone of a sentence.

The authors trained their model using only the second utterance from the user, without considering the context of Turns 1 and 2 in the dialogue. To improve upon this approach, we conducted several experiments. Initially, we tried simply concatenating all turns and treating them as a single flat sequence. However, as seen in the results section, this did not improve performance, since LSTMs are sequential learners. When fed with a long context (e.g., 60 tokens), LSTMs may "forget" the emotionally charged

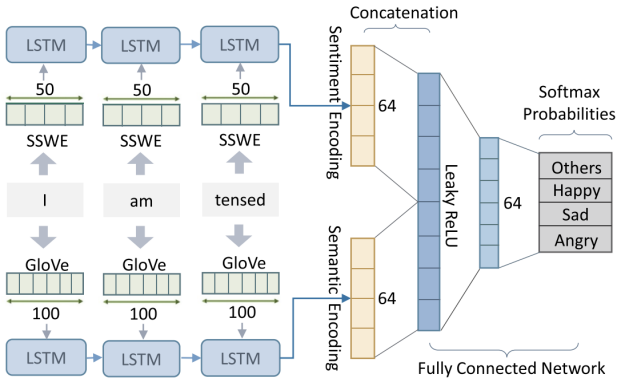

Figure 5: The architecture of Sentiment and Semantic Based Emotion Detector (SS-BED) Model

part at the end (Turn 3). To address this, we experimented with BiLSTMs, which provided a slight improvement in performance. This led us to a new idea: why not have a separate pipeline for processing the context? As a result, we enhanced the architecture by adding sentiment and semantics layers for the context as well. See Figure 6. So instead of two encodings to be combined before fully connected layers we have four which two of them are for the context and the other two are for the target. As we see in the results it improved the metrics to a great extent.

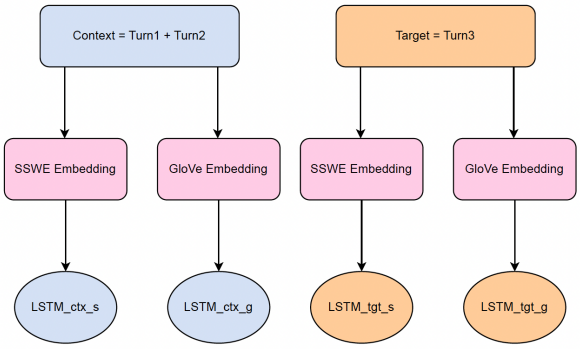

Figure 6: Enhanced Pipeline

4.3 Bi-GRU+CNN Hybrid Model

The proposed hybrid model combines deep learning and machine learning algoirhtms to predict emotions. The overall system diagram is shown in Figure7.

The architecture leverages deep learning (CNN and Bi-GRU) for feature extraction and machine

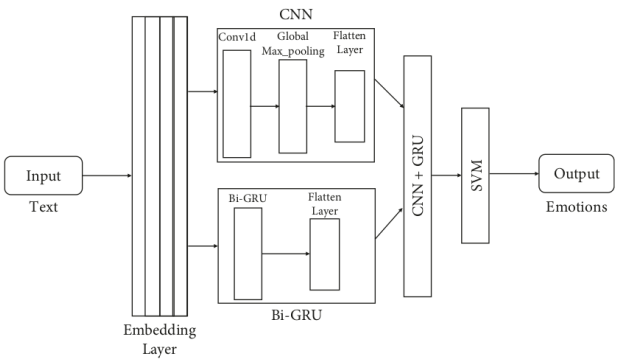

Figure 7: The architecture of the Bi-GRU+CNN Hybrid Model

learning (SVM) for classification. The pipeline begins by transforming input text into dense embeddings using Word2Vec. These embeddings are simultaneously processed by both the CNN and Bi-GRU RNN networks, modified as encoders by removing their final layers. Each encoder produces a latent vector for the given input embedding vector. Finally, these latent vectors are then concatenated and fed to the SVM classifier for final emotion prediction.

5 Baselines and Evaluation

We will evaluate models against both traditional baselines and state-of-the-art emotion-aware architectures. For lexicon-based baselines, we include the NRC Emotion Lexicon, a word-based emotion scoring method using Plutchik's model, which is simple yet effective for rule-based emotion classification (Mohammad and Turney, 2010) (Baccianella et al., 2010).

In terms of machine learning baselines, we consider Naive Bayes, a classic ML approach that relies on traditional feature engineering techniques for sentiment classification and is commonly used as a baseline for low-resource settings (Pang et al., 2002). Another key baseline is LSTM with Word2Vec, a recurrent neural network model that evaluates the impact of sequential dependencies on emotion recognition and is typically compared against transformer-based methods (Hochreiter and Schmidhuber, 1997).

5.1 Evaluation

We will explore the following quantitative metrics:

5.1.1 Quantitative Metrics (Classification Performance)

Precision, Recall, and F1-score, Macro-F1 Score (Opitz and Burst, 2021).

6 Experimental Details

6.1 Transformer Models

In this experiment, we evaluated several models for emotion classification tasks using three datasets: ISEAR (7 emotion classes) and Diar AI (6 emotion classes) and GoEmotions (11 emotion classes). We did a through hyperparameter tuning for the models.

Experimental Approach

- For Naive Bayes, BERT, DistilBERT, and RoBERTa, grid search was used to explore different hyperparameters.
- For GPT-2 and ELECTRA, Optuna was employed for hyperparameter optimization.

Key Hyperparameters Tested

Hyparameter	Values Tested
$\alpha prior$	0, 1, 10, 100, 1000, 10000
$\alpha likelihood$	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0
Use bigrams	True, False
stopwords	True, False

Table 3: Naive Bayes Hyperparameters

Hyperparameter	Values Tested
Epochs	3, 8
Batch size	32, 64
Learning rate	2e-5, 3e-5
Weight decay	0.01, 0.001
Fine-tune last layers	True, False

Table 4: BERT/DistilBERT Hyperparameters

Implementation Details

- All transformer models were configured for emotion classification tasks, with 7 emotion classes for

Hyperparameter	Values Tested
Epochs	3, 5, 10
Batch size	32, 64
Learning rate	2e-5, 3e-5, 5e-5
Weight decay	0.01, 0.001, 0.003, 0.0001
Fine-tune last layers	True, False

Table 5: RoBERTa Hyperparameters

Hyperparameter	Values Tested
Learning rate	1e-5 to 5e-5 (log scale)
Batch size	8, 16, 32
Epochs	3, 5
Weight decay	0.0001 to 0.1 (log scale)
Fine-tune last layers	True, False

Table 6: GPT-2/ELECTRA Hyperparameters

ISEAR data and 6 emotion classes for Diar AI.

- **Gradient clipping** was applied to BERT to prevent exploding gradients.
- **Optimizer**: The AdamW optimizer with linear learning rate warmup was used for all models.

6.2 SS-BED Model

In the experiments, the batch size was set to 128 for the basic SS-BED model and 4000 for the enhanced context model. Both models processed input sequences with a maximum length of 30 for the basic experiment and 60 for the enhanced model. The SSWE embedding dimension was 50, and the GloVe embedding dimension was 100. The models employed a hidden dimension of 64 in the LSTM layers, with 2 LSTM layers. The dropout rate was set to 0.25 for the basic model and 0.3 for the enhanced model. The learning rate for both models was 0.005, and they were trained for 8 epochs. Training was conducted on GPU (if available) or CPU. These hyperparameters were selected to control the model's complexity and optimize performance during training.

6.3 Bi-GRU Hybrid Model

Through systematic hyperparameter optimization, we determined the following optimal configurations for each dataset:
Dataset 1: batch_size=128, cell_size=64, dropout_keep=0.3, epochs=1000, learning_rate=1e-4, sequence_length=50, SVM_C=1.5, vocabulary size=10 000, embedding dimension=100.
Dataset 2: batch_size=256, cell_size=128, dropout_keep=0.3, epochs=150, learning_rate=3e-4, sequence_length=50, SVM_C=5, vocabulary size=10000, embedding dimension=100.

7 Results

7.1 Transformer Models

ELECTRA achieved the highest validation accuracy (0.9460) and test accuracy (0.9340). All transformer models significantly outperformed the Naive Bayes baseline. The best performing model in terms of macro F1 was ELECTRA (0.8953). Parameter tuning shows that smaller learning rates (2e-5 to 5e-5) generally performed better. Batch sizes of 32-64 were optimal across transformer models. Also we computed the correlation matrix to analyze the relationships between different emotions in the GoEmotion dataset, helping to identify how emotions co-occur or differ. By looking at this matrix 8 we see that for example, recognizing the close correlation between emotions like joy and excitement or anger and disgust can enhance model accuracy by accounting for these nuanced emotional connections. Based on these exploratory analysis we find that between the psychological models we have Plutchick will serve the best for this kind of data see Table 8.

7.2 SS-BED Model

Table 9 reports the ablation results for the SS-BED model. Feeding the entire context as a single flattened sequence alongside the target utterance (SS-BED-FC) yields a modest gain in Recall, but it reduces Precision, F1, and overall Accuracy. Introducing a BiLSTM substantially boosts performance because each layer now aggregates both forward and backward hidden representations. Our proposed architecture goes further: by encoding the dialogue context separately from the target utterance and then fusing the two representations, it surpasses every

other variant and both baselines, the one-layer LSTM with word2vec embeddings and the original SS-BED model.

7.3 Bi-GRU Hybrid Model

In Table 7 we evaluate our models using the same metrics as Bharti et al. (precision, recall, F1-score, and accuracy). Given the significant class imbalance—particularly for underrepresented emotions like surprise (1.5% of samples)—we focus on weighted metrics to better reflect real-world performance. While Bharti et al. do not explicitly specify their averaging method, the observed discrepancies suggest they likely adopted a similar weighting scheme. Our analysis reveals two key findings:

Model + Dataset	Prec.	Rec.	F1	Acc.
Bharti et al. + Dataset 1	82.39	80.40	81.27	80.11
Our Model + Dataset 1	76.56	71.80	72.14	71.80
Our Modeel + Dataset 2	81.94	**81.47**	80.89	**81.47**

Table 7: Performance Comparison Bi-GRU Model

- **Dataset 1 Performance:** Our model underperforms relative to Bharti et al., likely due to differences in architecture,training or data preprocessing.
- **Dataset 2 Improvement:** The enhanced performance stems from:
  - **Increased Dataset Size:** By incorporating DAIR-AI (partial subset) alongside ISEAR and Emotion-Stimulus we constructed Dataset 2 (26.3K samples) with approximately twice the volume of Dataset 1 (14.5K samples). This strategic substitution enhanced data diversity while maintaining balance.
  - **Better balance:** Improved representation of minority classes (e.g., *surprise*)

These results underscore the importance of dataset scale and class distribution in emotion classification tasks.

8 Future Work

8.1 SS-BED Model

- Apply token-level attention to Turn3 or use sentence-level attention between context and target representations.
- Add emotion intensity regression
- Train simultaneously on emotion + sarcasm detection

8.2 Bi-GRU+CNN Model

- Add attention layers in order to improve context understanding.
- Grid search of hyperparameters (Optuna, Keras Tuner) for better performance.
- Use of contextual BERT embeddings to capture word meanings based on surrounding text
- Apply data augmentation techniques on the dataset

9 Task Division

Our work is being carried out in a highly collaborative manner, with all team members contributing to each phase of the project while also focusing on different aspects, including model development, data pre-processing, and data analysis. Besides, the interpretation of results is a collective effort and it is conducted as a team.

References

n.d. Isear dataset. Kaggle. Accessed: Feb. 09, 2025.

A. F. Adoma, N.-M. Henry, and W. Chen. 2020. Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*, pages 2200–2204.

Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019. Semeval-2019 task 3: Emocontext contextual emotion detection in text. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 39–48.

D. Cortiz. 2021. Exploring transformers in emotion recognition: A comparison of bert, distillbert, roberta, xlnet and electra. *arXiv*.

Luna De Bruyne. 2023. The paradox of multilingual emotion detection. In *Proceedings of the 13th Workshop on*

Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, pages 458–466. Association for Computational Linguistics.

Dora Demszky, Daniel Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. *arXiv*.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

Bharti et al. 2022. Text-based emotion recognition using deep learning approach. *Computational Intelligence and Neuroscience*.

Chatterjee et al. 2019. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

C. Huang, A. Trabelsi, and Osmar R. Zaïane. 2019. Ana at semeval-2019 task 3: Contextual emotion detection in conversations through hierarchical lstms and bert. *arXiv*.

Richard S Lazarus. 1991. Progress on a cognitive-motivational-relational theory of emotion. *American psychologist*, 46(8):819.

Y. Li, J. Chan, G. Peko, and D. Sundaram. 2024. An explanation framework and method for ai-based text emotion analysis and visualisation. *Decision Support Systems*, 178:114121.

Saif Mohammad and Peter Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34.

J. Opitz and S. Burst. 2021. Macro f1 and macro f1. *arXiv*.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. *arXiv*.

Flor Miriam Plaza-del Arco, Alba Curry, Amanda Cercas Curry, and Dirk Hovy. 2024. Emotion analysis in nlp: Trends, gaps and roadmap for future directions. *arXiv preprint arXiv:2403.01222*.

A. Semeraro et al. 2025. Emoatlas: An emotional network analyzer of texts that merges psychological lexicons, artificial intelligence, and network science. *Behavior Research Methods*, 57(2):77.

V. Suresh and D. C. Ong. 2021. Using knowledge-embedded attention to augment pre-trained language models for fine-grained emotion recognition. *arXiv preprint arXiv:2108.00194*.

B. Wan, P. Wu, Chai Kiat Yeo, and G. Li. 2024. Emotion-cognitive reasoning integrated bert for sentiment analysis of online public opinions on emergencies. *Information Processing and Management*, 61(2):103609.

10 Appendix

In the next following pages several results are appendended.

Model	Best Val	Test Acc	Macro F1	Weighted F1	Precision (Avg)	Recall (Avg)
Naïve Bayes	0.8350	0.8385	0.7600	0.8343	0.8191	0.7394
BERT	0.9425	0.9265	0.8800	0.9262	0.8860	0.8792
ELECTRA	**0.9460**	**0.9340**	**0.8900**	**0.9341**	0.8962	**0.8954**
RoBERTa	0.9385	0.9275	0.8800	0.9271	0.9002	0.8791
DistilBERT	0.9415	0.9270	0.8800	0.9269	0.8880	0.8780

Table 8: Comparison of Model Performance on Dair AI

		Models				
		LSTM	**SS-BED**	**SS-BED-FC**	**SS-BED-BiLSTM**	**Ours**
Happy	Percision	41.2	45.9	33.7	54.5	**50.4**
	Recall	47.7	52.6	57.1	72.3	**62.7**
	F1	43.2	48.3	42.4	62.8	**55.3**
Sad	Percision	39.8	45.9	36.2	50.9	**54.2**
	Recall	49.3	52.6	74.8	83.4	**75.6**
	F1	43.2	48.3	48.9	63.6	**63.5**
Angry	Percision	40.6	45.9	42.3	49.7	**56.2**
	Recall	48.1	52.6	**86.4**	49.2	85.9
	F1	42.5	48.3	56.7	49.0	**67.3**
F1 MICRO		49.8	56.7	49.61	58.83	**62.56**
F1 MACRO		48.5	55.6	48.71	57.79	**61.82**

Table 9: Performance comparison of different SS-BED variants

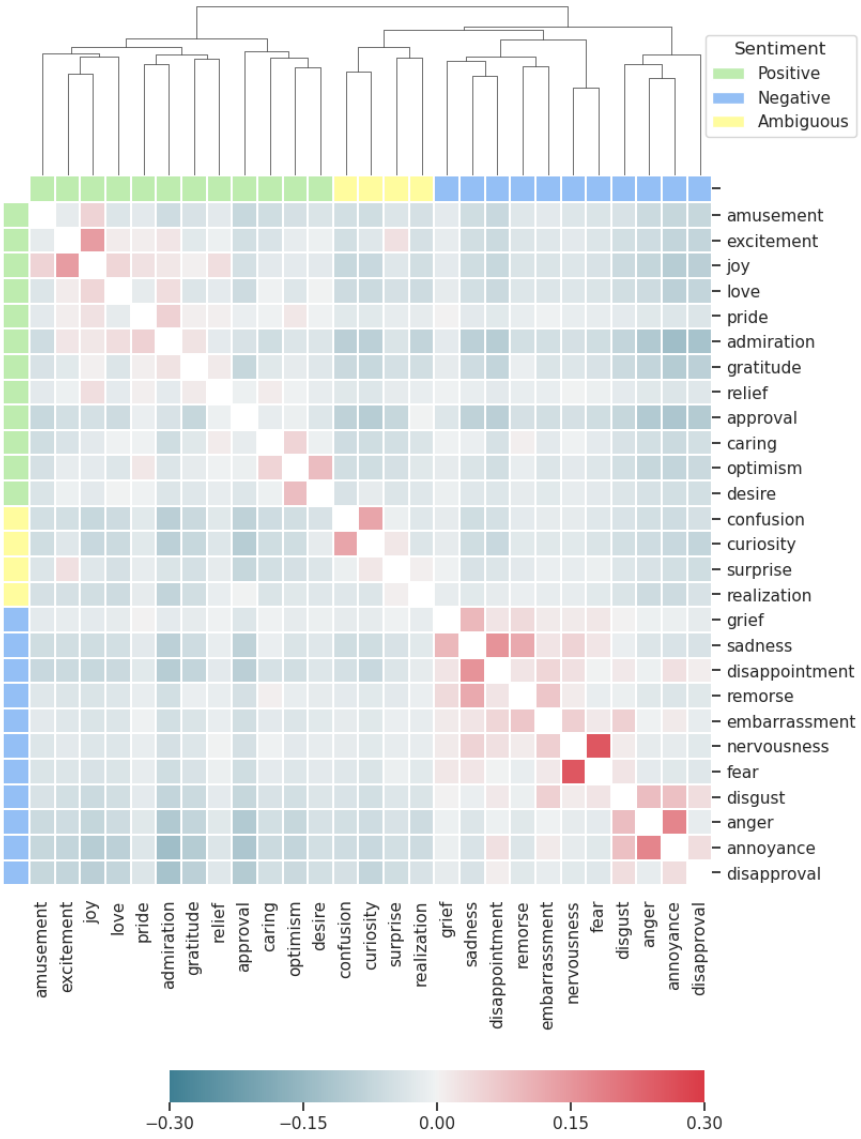

Figure 8: Heatmap Correlation Matrix