NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

SC4001 Neural Networks and Deep Learning

Group Project

Group Members:

Name	Matriculation No	Contributions
Sethi Aryan	U2123583E	Data Cleaning and Preprocessing. Baseline models, BERT_CNN and RoBERTa_CNN model training. Report writing.
Shourya Kuchhal	U2123334A	Data Cleaning and Preprocessing. Baseline, BERT_CNN and RoBERTa CNN model training. Report writing
Mukherjee Tathagato	U2120365K	Report writing, editing and RoBERTa analysis

Abstract

This study delves into the domain of Text Emotion Recognition (TER), a critical intersection of natural language processing and affective computing. It presents a comprehensive examination of contemporary deep learning techniques, emphasizing the integration of contextual information for emotion recognition in textual data. Our work predominantly focuses on exploring the efficacy of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), particularly Bidirectional Long Short-Term Memory (BiLSTM) networks with and without attention mechanisms, and the innovative use of Bidirectional Encoder Representations from Transformers (BERT) and Robustly Optimized BERT Pretraining Approach (RoBERTa) combined with CNN for TER. We employed two distinct datasets, Crowdflower and Emo2019, to assess model performance across varying linguistic domains. The results highlight the importance of capturing both local and global contextual information in improving classification accuracy. While RNNs with attention mechanisms yield improvements, transformer-CNN hybrids, particularly RoBERTa+CNN, consistently outperform other models in clean data settings. On noisier data, such as Crowdflower, RoBERTa also demonstrates resilience when evaluated on more balanced subsets. This research also reveals the challenges of dataset variability and the limitations of relying solely on pre-trained embeddings, underscoring the importance of domain adaptation and task-specific fine-tuning. While our findings affirm the strengths of advanced transformer models in TER, they also emphasize the need for careful model selection and tuning to address domain-specific constraints.

Contents

1. Introduction

Understanding the emotions conveyed by a string of words has been a perennial problem for both Deep Learning (DL) and Machine Learning (ML) models alike. Something so intuitive for us as humans has proven to be quite a challenge even for models with complex architecture. Being able to understand the emotion behind a sentence holds value in many contexts, such as in Social Media (e.g. Twitter Content Moderation) and also for more granular sentiment analysis which can help Customer Service teams across a wide-range of Businesses that use online communications to reach their customers (e.g. Whatsapp Business, Facebook Messenger, etc.).

In this paper, we will briefly cover existing attempts in building DL models for Text Emotion Recognition (TER), and the challenges they faced. One of the primary difficulties is the ability for the model to produce a prediction based on both Global and Local Information, among many others. With these challenges and considerations in mind, we aim to propose a new model, and a new direction for future DL models to take in refining their TER capabilities.

2. Literature Review

The most common techniques employed for TER involve the use of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) in various configurations.

We find that the two elements that are were mostly experimented with in various techniques that surround Text Emotion Recognition (TER) were:

1. **Feature Representation: How is word embedding done?**
2. **Model Architecture: What strategy is used to capture contextual data?**

2.1 Word Embeddings

Many techniques make use of pre-trained vectors for word embedding such as word2vec and GloVe (Kim, 2014). In contrast, others have prepared their word embeddings manually, as seen in the papers by Zhou et al. (2016) and Lai et al. (2015). Pre-trained word embeddings are time-efficient and are especially beneficial when their pre-training context resembles the data in question. They allow for quick deployment of models as you do not need to generate your own embeddings, which would increase training time. However, this can sometimes result in the need to manually handle out-of-vocabulary words, which is not always ideal. Thus there is a trade-off between practicality and accuracy when choosing to use common pre-trained vectors.

2.2 Model Architecture

Regarding model architecture, the use of Attention Layers and RNNs, typically in the form of a Bi-Directional LSTM, is noted for capturing more contextual data. Compared to CNNs, a recurrent structure appears to offer improved context capture from sentences (Lai et al., 2015). Bi-directional processing allows the use of both past and future information within the sentence.
Furthermore, some parts of a sentence may contribute more significantly to the overall emotion conveyed, which justifies the use of an Attention Module (Zhou et al., 2016; Li, 2023). The introduction of the

Attention mechanism has yielded superior results compared to models without it. However, this also results in a more complex model that requires more time to train.

Hence, our goal is to find a way to reduce the need for excessive training data and complex models while maintaining an accurate model that is able to make use of both local and global information.

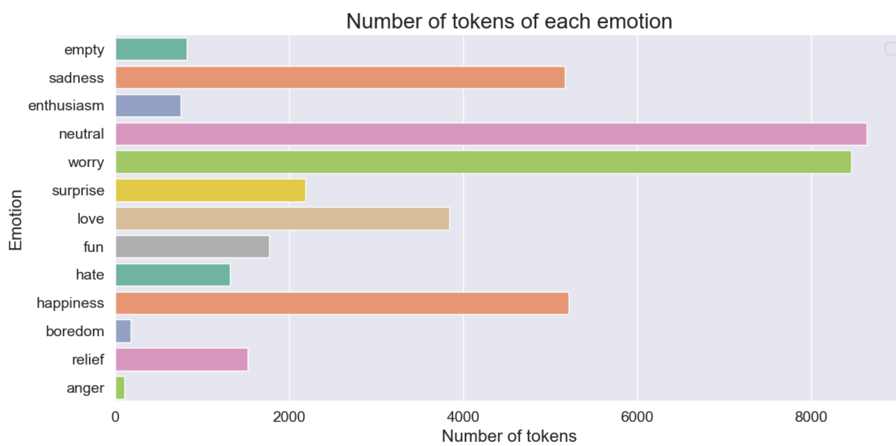
3. Methodology

3.1 Datasets

We use two datasets to understand how our models fair when dealing with conversational and discrete texts.

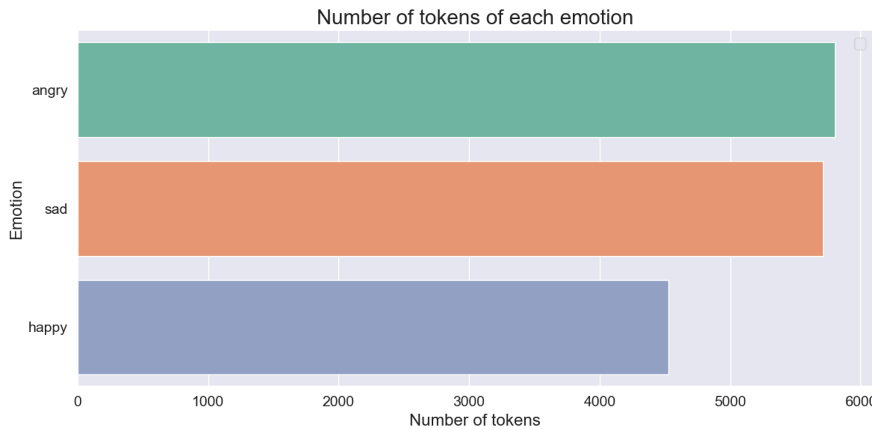
3.1.1 Crowdflower (Crowdflower, 2016)

The text sentiment analysis dataset provided by Crowdflower consists of 40,000 entries of tweets across 13 labels marked for its emotional content. Additionally, the data provides the authors for each text or tweet, allowing for supplementary global information to train the models. As seen in the data distribution of the Crowdflower dataset below, it is severely imbalanced. Due to this, we have decided to only keep the top 5 labels (Worry, Neutral, Happiness, Sadness, Love) for training and classification.



3.2.2 Emo2019 (Chatterjee, 2019)

The Emo2019 dataset provided by Hugging Face consists of approximately 35,700 entries that have already been split into train (30,200) and test (5500) sets. This dataset includes conversational textual dialogues that have been labeled with four emotion classes - Happy, Sad, Angry, and Others. The label ‘Others’ has been dropped to focus on specific emotions.



3.2 Preprocessing

Given the complexity and variability of human language, data preprocessing is a critical step to prepare the data which will be fed into the models.

Datasets have been split in 8:2 ratio into train sets and test sets. Train set data are further split in the same ratio to obtain validation data sets.

The 3 data sets are then loaded in batches of 16, with the training data shuffled to be used for training.

3.2.1 Preprocessing of data for training of our baseline models

We did the following for our Baseline Models:

1. **Label Encoding:** Sentiments / Emotions were converted into a numerical format

2. **Tokenization:** Using Keras Tokenizer, the sentences were turned into sequences of integers. The default setting was used, which also removed punctuation.

3. **Sequence Padding:** As our input data for all our models must be of the same size, we standardized the lengths of all input sequences to the maximum length by applying padding.

4. **Embeddings:** Depending on the model, we either learned the embedding during the training process, or used pre-trained GloVe vectors
    a. The rationale for this is to have at least one model that does not benefit from the Global context that GloVe vectors provide (Pennington, 2014)

3.3 Baseline Models

For fair comparisons, we will consider a CNN Model, and two models based on RNNs. The first RNN model will incorporate a bidirectional LSTM and use pre-trained embeddings, the second will be the same, but with an Attention Layer added. This is to establish comparisons to the techniques we discussed earlier in our review of related work.

3.3.1 CNN Model Architecture

- Embedding Layer: Converts input tokens into dense vectors of fixed size (embedding_dim=100).
- Convolutional Layers: A series of convolutional layers with filter sizes of 3, 4, and 5. Each convolutional layer has n_filters=100.
- Max Pooling: Applied to each convolutional output to reduce dimensionality.
- Dropout Layer: Applied with a probability of 0.5 to reduce overfitting.
- Fully Connected Layer: Transforms the concatenated output of the convolutional layers to the output size, which matches the number of classes in the dataset.
- Output: Produces a logit for each class in the output.

3.3.2 Bidirectional LSTM (BiLSTM) Model Architecture

- Embedding Layer: Utilizes pre-trained GloVe embeddings to convert input tokens into vectors. The embeddings are not frozen and can be fine-tuned.
- Bidirectional LSTM: Processes the embeddings with a hidden dimension of 256 and 2 layers. It captures sequential information in both forward and backward directions.
- Dropout Layer: Applied to the LSTM output with a probability of 0.5.
- Fully Connected Layer: Maps the LSTM output to the size of the output classes.
- Softmax Activation: Applied to the output of the fully connected layer to obtain class probabilities.

3.3.3 Bidirectional LSTM with Attention (AttBiLSTM) Model Architecture

- Embedding Layer: Same as the BiLSTM model, using pre-trained GloVe embeddings.
- Bidirectional LSTM: Identical to the BiLSTM model with a hidden dimension of 256 and 2 layers.
- Attention Layer: Applies attention to the LSTM output, focusing the model on specific parts of the input sequence for making predictions.
- Dropout Layer: Applied after attention application with a probability of 0.5.
- Fully Connected Layer: Maps the attentive LSTM output to the size of the output classes.
- Softmax Activation: Generates class probabilities from the output of the fully connected layer.

3.4 Proposed Model

This section goes through the model architecture of our proposed solution to capture both local and global information for emotion recognition.

Instead of going through the preprocessing steps described earlier for our baseline models, we take a different approach.

3.4.1 Bidirectional Encoder Representations from Transformers

For our proposed model, we used the Bidirectional Encoder Representations from Transformers (BERT) model to handle data preprocessing and word representations. BERT is a transformers model pre-trained on a large corpus of English data including BookCorpus and Wikipedia (Devlin, 2019), and the variant we used is BERT Base uncased with 110M parameters.

With BERT, traditional preprocessing steps may become redundant as it is designed to work with raw text to understand context. BERT uses its own tokenizer to break texts down into tokens that are available in its vocabulary, and it requires special tokens which are used as an aggregate representation for classification. BERT also provides contextualized word embeddings based on the words' context in the sentence, compared to fixed representations that traditional methods provide. This is especially important in emotion recognition where the sentiment of a word can change based on surrounding words.

BERT is able to handle global information through its attention mechanism which considers the entire context of a sentence. The model is composed of multiple attention layers and as information passes through each layer, it captures more abstract or high-level representations.

3.4.2 Robustly Optimized BERT Pretraining Approach (RoBERTa)

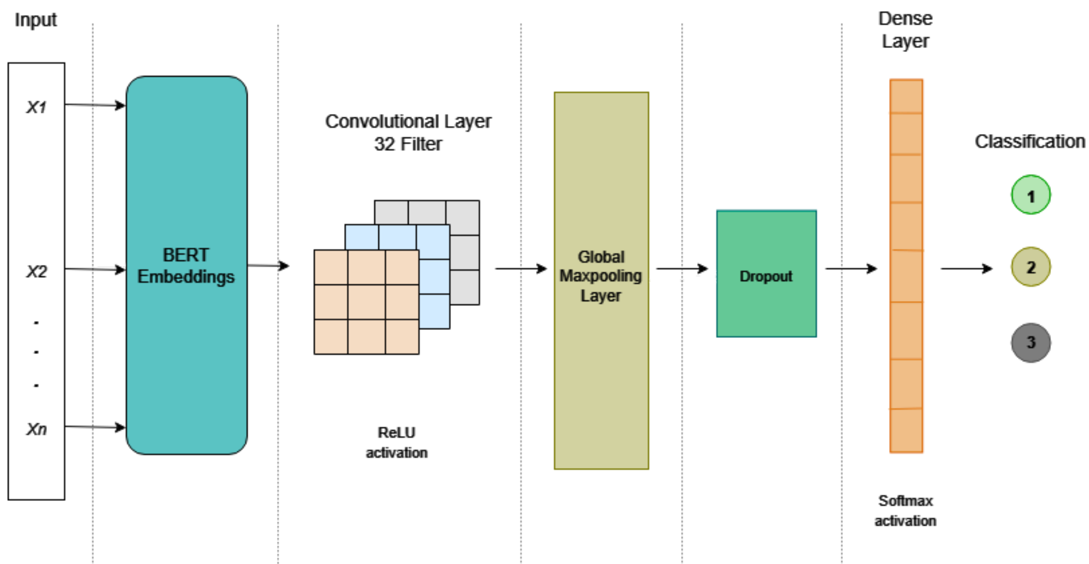
In addition to BERT, we explored the use of RoBERTa to enhance the quality of contextual embeddings used for emotion recognition. RoBERTa is an improvement over the original BERT architecture (Liu et al., 2019) primarily through training on a much larger corpus (Common Crawl News, OpenWebText, etc.) and using dynamic masking, removal of next sentence prediction, and longer training durations. These changes allow RoBERTa to capture nuanced details in emotion recognition tasks, making it useful for this project.

For our implementation, we used the RoBERTa-base model (125M parameters). Similar to our BERT pipeline, RoBERTa handles tokenization internally using a byte-level Byte-Pair Encoding (BPE) tokenizer. Text inputs are encoded into embeddings that incorporate both local and global context via self-attention across multiple transformer layers.

To ensure fair comparison, we retained the same CNN architecture used in the BERT+CNN model. The final hidden states for each token from RoBERTa was passed through the same layers as in the previous section. This setup allows us to evaluate the impact of different pretraining strategies (BERT vs. RoBERTa) on the downstream task of Text Emotion Recognition, while maintaining consistency in the classification head and experimental design.

Convolutional Neural Network

After the pre-trained BERT and RoBERTa model are used to capture local and global information, the embedding vector will subsequently be fed into the CNN. The CNN's structure is formed by a 1D convolutional layer, a global max pooling layer, and a dense output layer with softmax activation for classification.



4. Experimentation and Results

4.1 Experiment Design

We perform two experiments: one on each dataset. All of the models are then trained on the preprocessed training data for each dataset. We implemented early-stopping to prevent over-fitting.

After training, we run the model predictions on the test data and compute the Precision, Recall, F1-Scores to assess how well they fared in classifying the emotions based on the given texts.

4.2 Results

4.2.1 Experiment 1: EMO2019 Dataset

Table 1: Classification Report of Different Models on EMO2019

Model	Metric	Precision	Recall	F1-Score	Support
CNN	Angry	0.84	0.86	0.85	298
	Happy	0.90	0.80	0.85	284
	Sad	0.77	0.84	0.80	250
	Accuracy			0.83	832
	Macro Avg	0.84	0.83	0.83	832
	Weighted Avg	0.84	0.83	0.83	832
BiLSTM	Angry	0.87	0.83	0.85	298
	Happy	0.88	0.82	0.85	284
	Sad	0.77	0.87	0.81	250
	Accuracy			0.84	832
	Macro Avg	0.84	0.84	0.84	832
	Weighted Avg	0.84	0.84	0.84	832
AttBiLSTM	Angry	0.88	0.92	0.90	298
	Happy	0.94	0.88	0.91	284
	Sad	0.85	0.86	0.85	250
	Accuracy			0.89	832
	Macro Avg	0.89	0.89	0.89	832
	Weighted Avg	0.89	0.89	0.89	832
BERT+CNN	Angry	0.90	0.94	0.92	298
	Happy	0.93	0.97	0.95	284
	Sad	0.94	0.85	0.89	250
	Accuracy			0.92	832
	Macro Avg	0.92	0.92	0.92	832
	Weighted Avg	0.92	0.92	0.92	832
RoBERTa+CNN	Angry	0.93	0.93	0.93	298
	Happy	0.98	0.92	0.95	284
	Sad	0.88	0.94	0.91	250
	Accuracy			0.93	832
	Macro Avg	0.93	0.93	0.93	832
	Weighted Avg	0.93	0.93	0.93	832

We can observe from the Macro F1 scores that each approach shows steady improvement in the accuracy of text classification. From CNN, to BiLSTM to AttBiLSTM and eventually our proposed models, which display the best performance on the emo2019 (Chatterjee, 2019) dataset. Notably, there is an improvement in the Macro F1 score from BERT+CNN to RoBERTa+CNN, suggesting more extensive pretraining and optimized architecture provide benefits in capturing nuanced emotional expressions in the dataset.

4.2.2 Experiment 2: Crowdflower Dataset

Table 2: Classification Report of Different Models on Crowdflower

Model	Emotion	Precision	Recall	F1-Score	Support
CNN	Happiness	0.43	0.21	0.28	1034
	Love	0.42	0.51	0.46	809
	Neutral	0.41	0.58	0.48	1683
	Sadness	0.43	0.19	0.26	1107
	Worry	0.40	0.48	0.43	1630
	Accuracy			0.41	6263
	Macro Avg	0.42	0.39	0.38	6263
	Weighted Avg	0.42	0.41	0.39	6263
BiLSTM	Happiness	0.34	0.55	0.42	1034
	Love	0.48	0.38	0.42	809
	Neutral	0.47	0.49	0.48	1683
	Sadness	0.46	0.08	0.13	1107
	Worry	0.41	0.51	0.45	1630
	Accuracy			0.42	6263
	Macro Avg	0.43	0.40	0.38	6263
	Weighted Avg	0.43	0.42	0.40	6263
AttBiLSTM	Happiness	0.45	0.36	0.40	1034
	Love	0.56	0.39	0.46	809
	Neutral	0.51	0.47	0.49	1683
	Sadness	0.51	0.08	0.13	1107
	Worry	0.38	0.73	0.50	1630
	Accuracy			0.44	6263
	Macro Avg	0.48	0.41	0.40	6263
	Weighted Avg	0.47	0.44	0.41	6263
BERT+CNN	Happiness	0.17	0.16	0.16	1034
	Love	0.13	0.11	0.12	809
	Neutral	0.27	0.27	0.27	1683
	Sadness	0.16	0.07	0.10	1107
	Worry	0.25	0.38	0.30	1630
	Accuracy			0.22	6263
	Macro Avg	0.20	0.20	0.19	6263
	Weighted Avg	0.21	0.22	0.21	6263
RoBERTa+CNN	Anger	0.10	0.05	0.06	22
	Happiness	0.87	0.85	0.86	1042
	Sadness	0.84	0.87	0.85	1033
	Accuracy			0.85	2097
	Macro Avg	0.60	0.59	0.59	2097
	Weighted Avg	0.85	0.85	0.85	2097

We can once again observe the increase in accuracy from the baseline CNN model to the AttBiLSTM model. However, all models perform considerably worse on this dataset. Furthermore, our proposed BERT+CNN model struggles noticeably when compared to the rest. Interestingly, RoBERTa+CNN diverges from this trend, achieving a substantially higher weighted F1-score of 0.85 despite being evaluated on only three emotion classes. This indicates its robustness on more balanced subsets, though a full five-class comparison remains limited due to label coverage.

5. Discussion

5.1 Interpretation of Results

From the results in **Table 1 and 2**, we can see that the strategies described in related works to introduce better capture of local and global information do help in improving on the accuracy of a more basic CNN model. The use of a Bi-Directional LSTM showed marginal improvements, but greater accuracy gains were made when combined with the Attention Layer. Demonstrating that there is merit in using an Attention mechanism to help the model focus on important information within the sentence.

However, these improvements come at the cost of model complexity. Furthermore, while we did use pre-trained GloVe vectors for word embeddings, it is best practice to further fine-tune these embeddings so that the model is better tuned towards the task at hand.

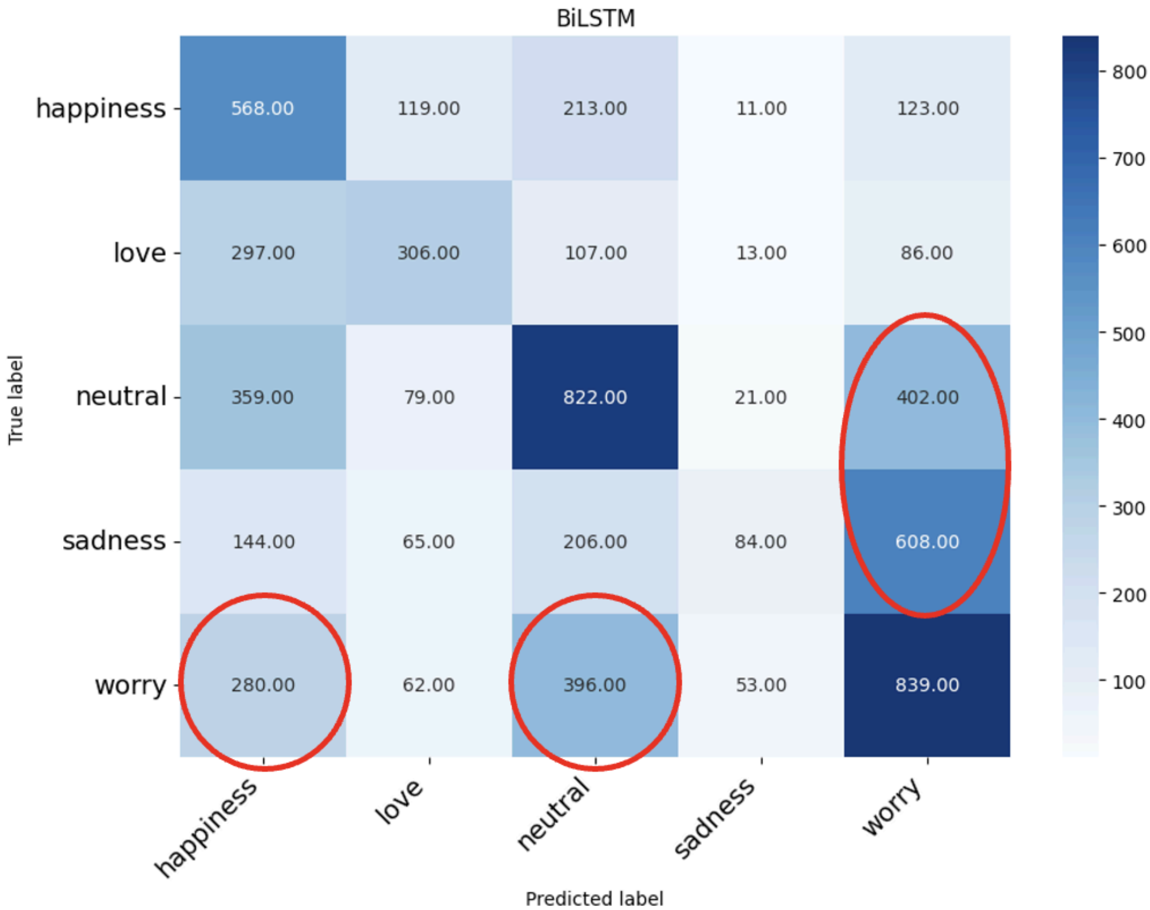
Our proposed models (BERT/RoBERTa + CNN) however show much better performance with a less complex setup. As seen from the results in Table 1, they perform considerably better when classifying the emotions from the text. This could be due to the contextually rich embeddings generated by BERT and RoBERTa, which can help capture the nuances that are necessary in recognising emotions in a conversational setting. Furthermore, both models are composed of multiple attention layers that allow for a more global understanding of sentence structure.

However, in Table 2, we observe that the use of BERT decreases the accuracy of the model. This may be due to the Crowdflower dataset being inherently more difficult to work with and significantly different from the data on which BERT was pre-trained. Interestingly, while BERT+CNN struggled with the five-class Crowdflower task, RoBERTa+CNN (evaluated on a focused three-class subset) achieved notably high accuracy and F1 scores. This suggests RoBERTa's pretraining on a larger and more diverse corpus may contribute to better generalization, even on noisier or domain-shifted data.

Furthermore, the data set contains a large amount of URLs and punctuations that affect the accuracy of prediction after being encoded by the BERT and RoBERTa tokenizers. Such punctuations are filtered in the tokenizer used for BiLSTM models, potentially giving them an advantage in handling noisy inputs in this particular dataset.

5.2 Further Investigation

To further understand why there is such a difference in our models, we plotted confusion matrices to visualize the misclassification done by the models. We noticed some pairs of classes were misclassified far more than others. Uncovering the raw inputs for these classes revealed that perhaps the dataset is not labeled as clearly as we previously assumed.



<u>**Confusion Matrix of BiLSTM Model on Crowdflower Dataset**</u>

By evaluating the raw inputs of the Crowdflower data set, we observed inputs such as:

"Spent the night finally relaxing with Nogard on WoW after finishing some work. Needed to take a small break from art. I really missed this"

Labeled as "worry". Such a sentence should not be interpreted to be a worrying statement as no word suggests so. Such miscategorised sentences would affect the accuracy of the models in future predictions. We are unable to evaluate the total number of wrongly categorized sentences as it would require manual selection from the large data set.

5.3 Limitations and Assumptions

Our study is not without its limitations. For practicality, we did not spend time on extensive parameter tuning. This might have led to suboptimal configurations affecting the models' performance, particularly on the Crowdflower dataset. Future work with more computational power could address this limitation by undertaking a more exhaustive search for the best model parameters. However, we believe that the point still stands that leveraging BERT does prove to be advantageous and serves as a good starting point to build better models for TER. Additionally, while RoBERTa+CNN performed well on a filtered subset of Crowdflower, its evaluation was limited to only three emotion classes, making direct comparisons with other models on the full dataset less conclusive.

Our assumption that pre-trained embeddings would be universally effective across different datasets might have contributed to the underperformance on the Crowdflower dataset. This highlights the importance of domain adaptation in natural language processing tasks.

Conclusion

In conclusion, this investigation rigorously examines the challenging domain of Text Emotion Recognition (TER), an interdisciplinary endeavor that intersects the nuanced spheres of natural language processing and affective computing.

The primary emphasis of this study has been the strategic leverage of both local and global contextual information to refine the predictive capabilities of TER models. This approach, synergizing transformer-based models like BERT and RoBERTa with the robust feature extraction capabilities inherent in CNN architectures, demonstrates enhanced performance.

The divergent performance metrics observed across the datasets, underscore the importance of domain-specific adaptation when developing these models. Notably, RoBERTa+CNN achieved the highest performance on the Emo2019 dataset and showed strong robustness even on a filtered version of the noisier Crowdflower dataset.

Looking ahead, there exists substantial scope for refinement and further exploration. This includes fine-tuning the BERT model used during training to better fit the target domain, whether for use in social media, customer service or any other context where understanding the emotion conveyed by text alone is paramount. As TER models continue to evolve, their successful deployment will depend not only on model architecture but also on thoughtful alignment with real-world data and use-case needs.

References

Chatterjee, A., Narahari, K. N., Joshi, M., & Agrawal, P. (2019). SemEval-2019 Task 3: EmoContext Contextual Emotion Detection in Text. In J. May, E. Shutova, A. Herbelot, X. Zhu, M. Apidianaki, & S. M. Mohammad (Eds.), Proceedings of the 13th International Workshop on Semantic Evaluation (pp. 39–48). Association for Computational Linguistics. https://doi.org/10.18653/v1/S19-2005

CrowdFlower. (2016, July 15). Sentiment Analysis: Emotion in Text [Dataset]. CrowdFlower. https://www.crowdflower.com/data-for-everyone/

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. North American Chapter of the Association for Computational Linguistics. Retrieved from https://arxiv.org/abs/1810.04805

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint arXiv:1408.5882*. Retrieved from https://arxiv.org/abs/1408.5882

Lai, S., Xu, L., Liu, K., & Zhao, J. (2015). Recurrent Convolutional Neural Networks for Text Classification. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (pp. 2267-2273). Retrieved from https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9745

Li, J., Wang, X., & Zeng, Z. (2023). A Dual-Stream Recurrence-Attention Network with Global-Local Awareness for Emotion Recognition in Textual Dialogue. arXiv preprint arXiv:2307.00449. Retrieved from https://arxiv.org/abs/2307.00449

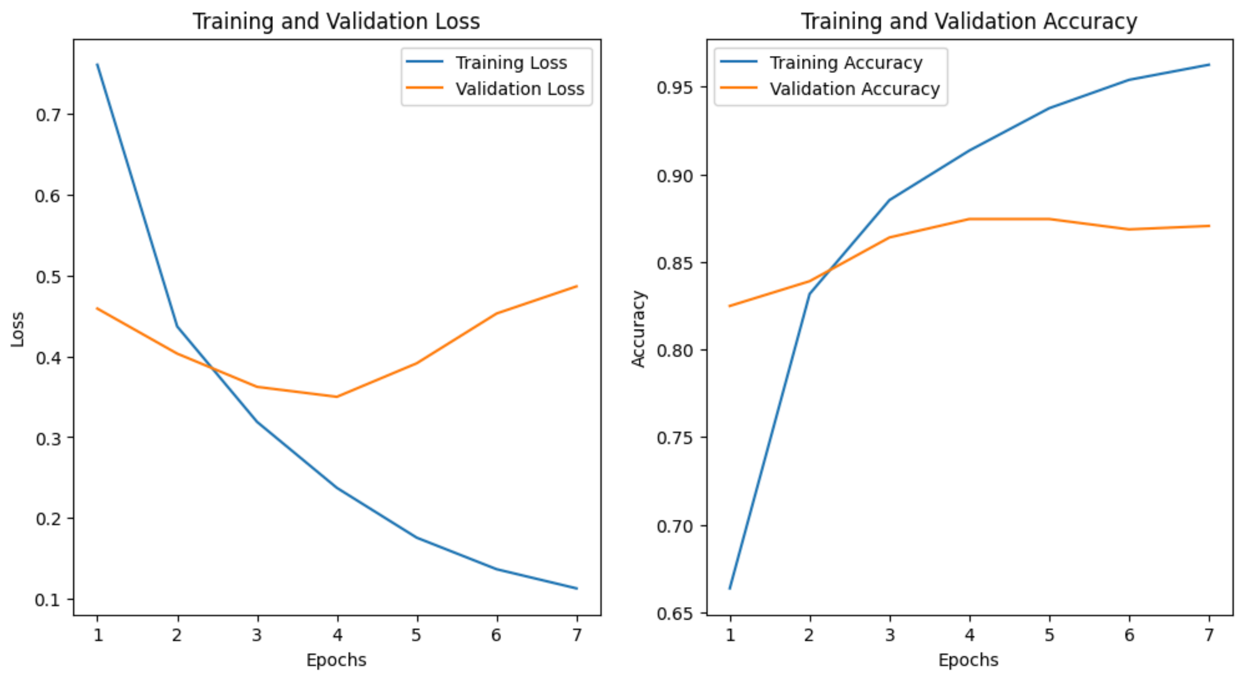
Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. arXiv preprint arXiv:1907.11692. https://arxiv.org/abs/1907.11692

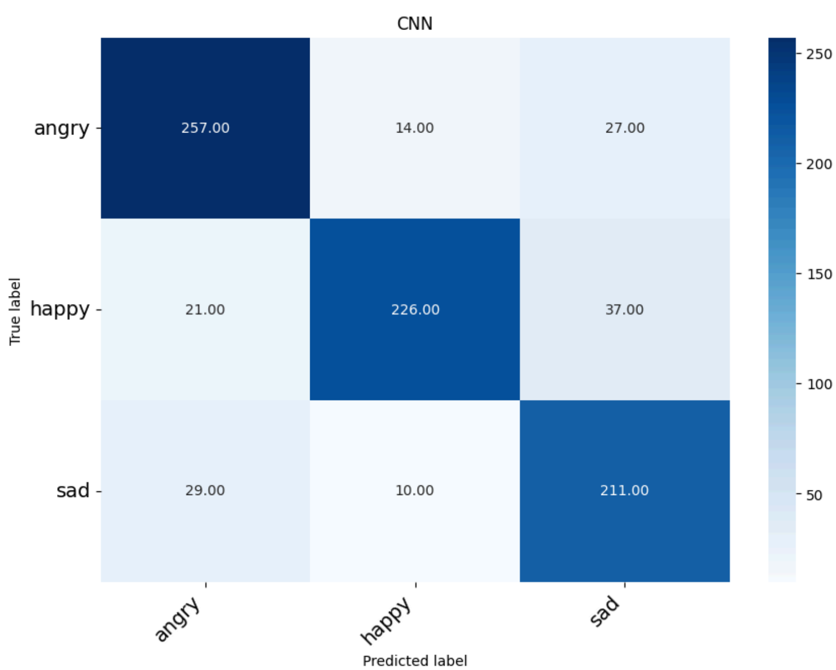
Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global Vectors for Word Representation. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1532-1543). Doha, Qatar: Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., & Xu, B. (2016). Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Volume 2: Short Papers), 207-212. Retrieved from https://www.aclweb.org/anthology/P16-2034

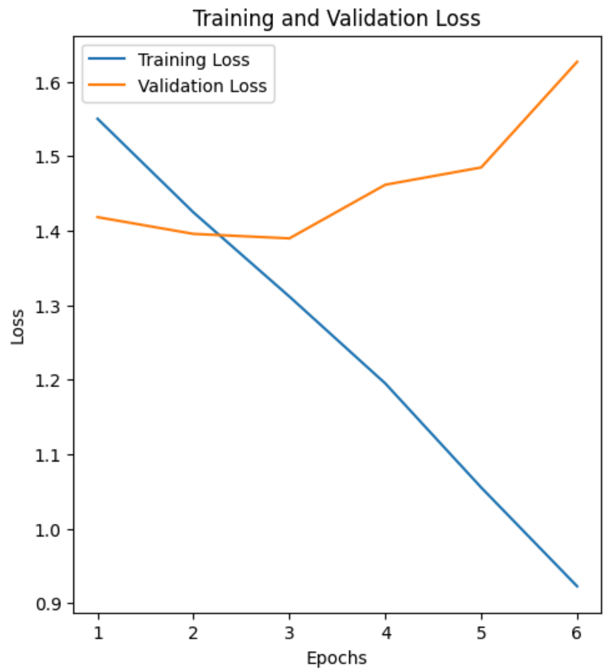
Appendix

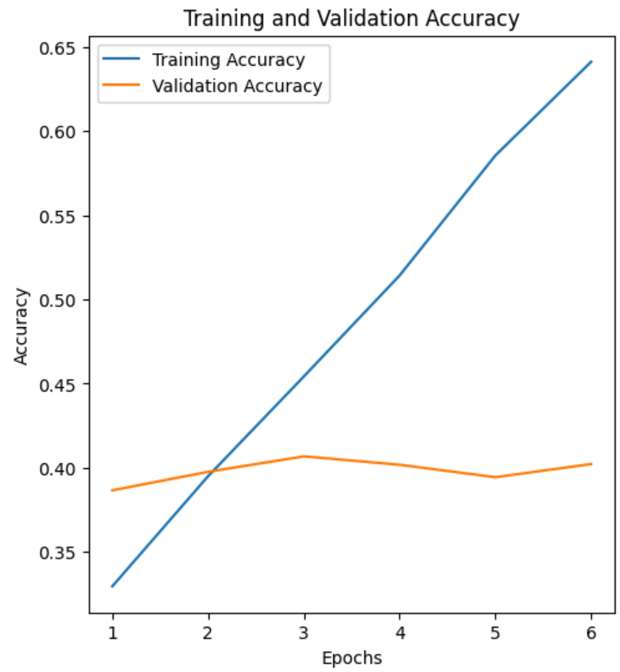
CNN, EMO2019

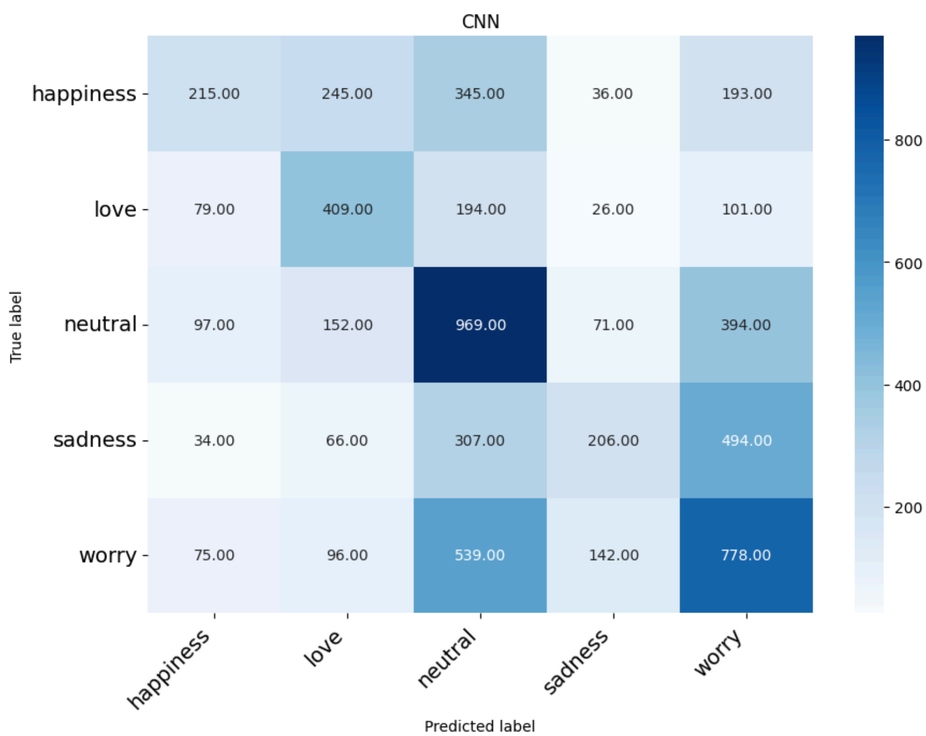




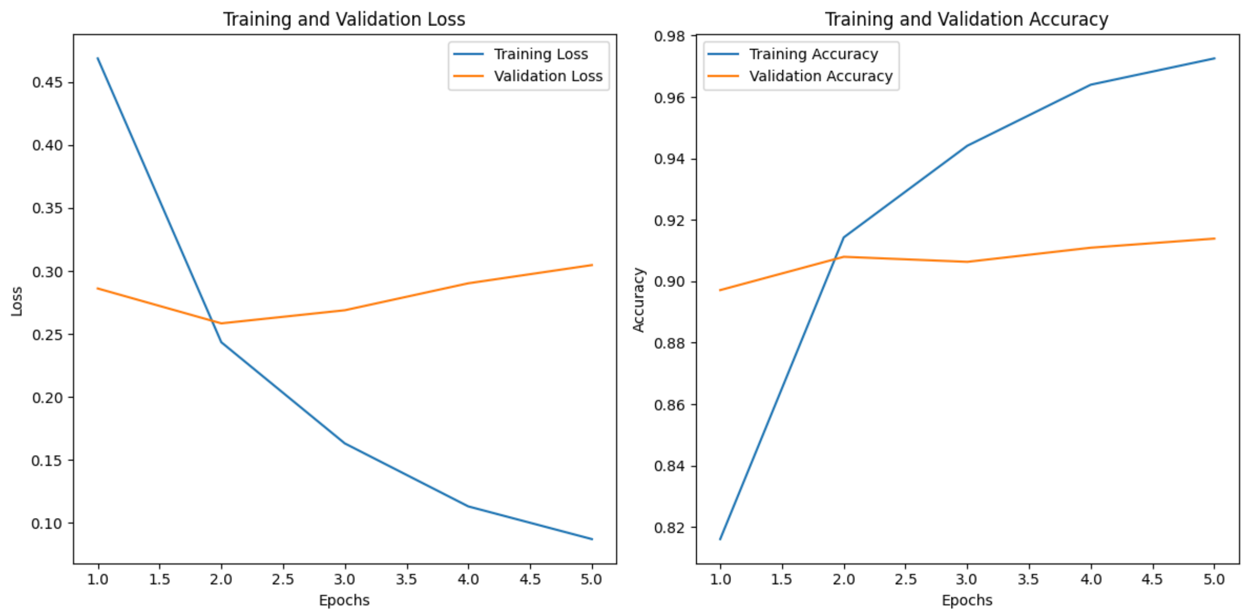
CNN, CROWDFLOWER

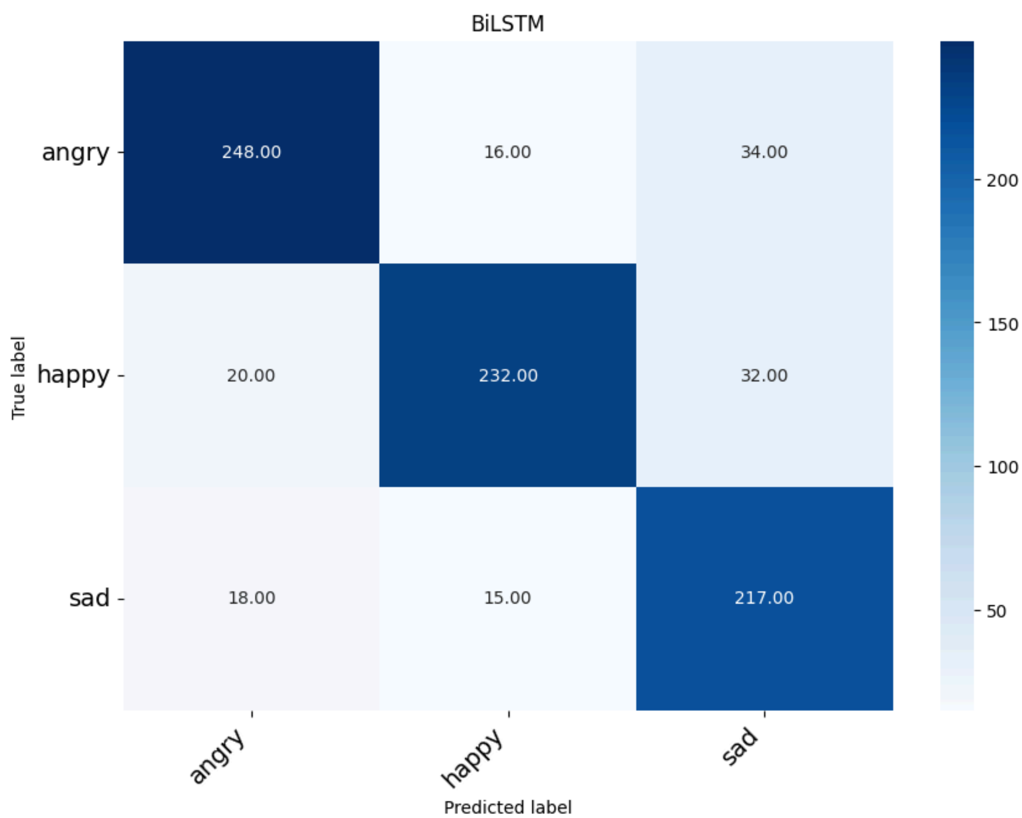




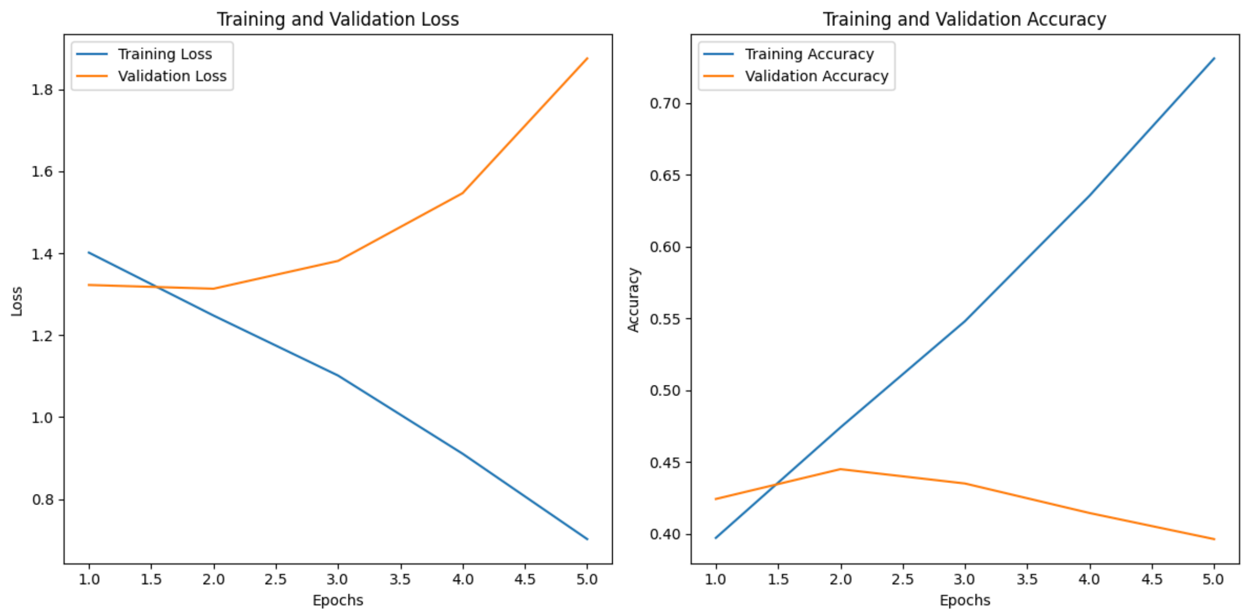


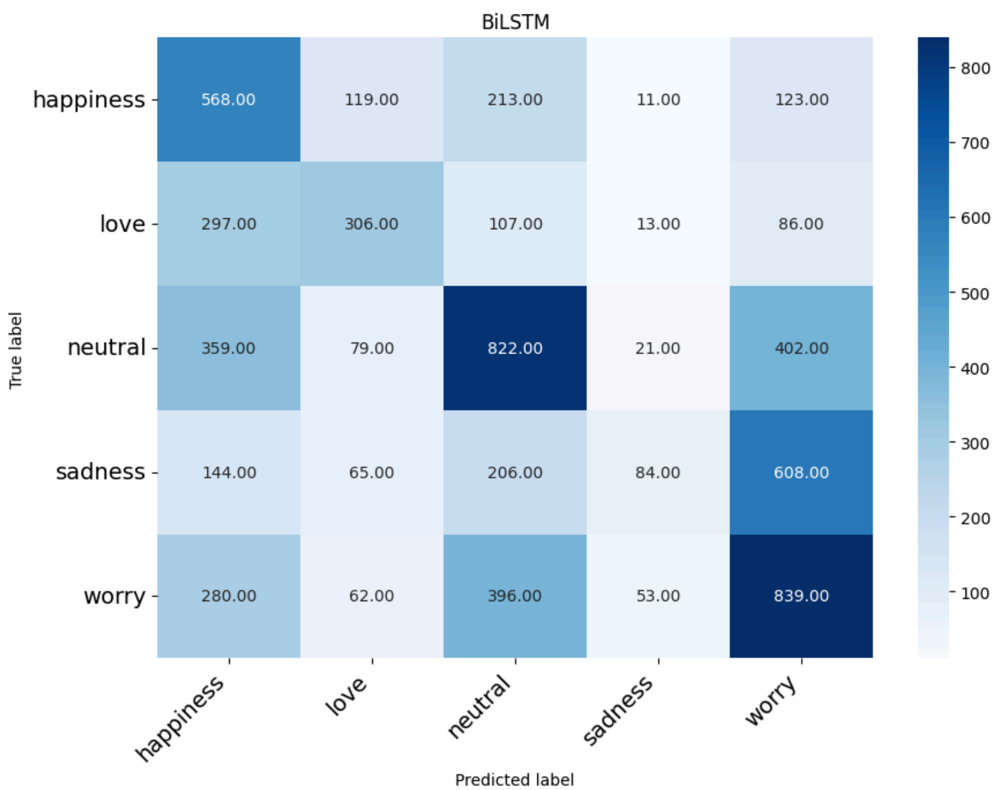
BiLSTM, EMO2019



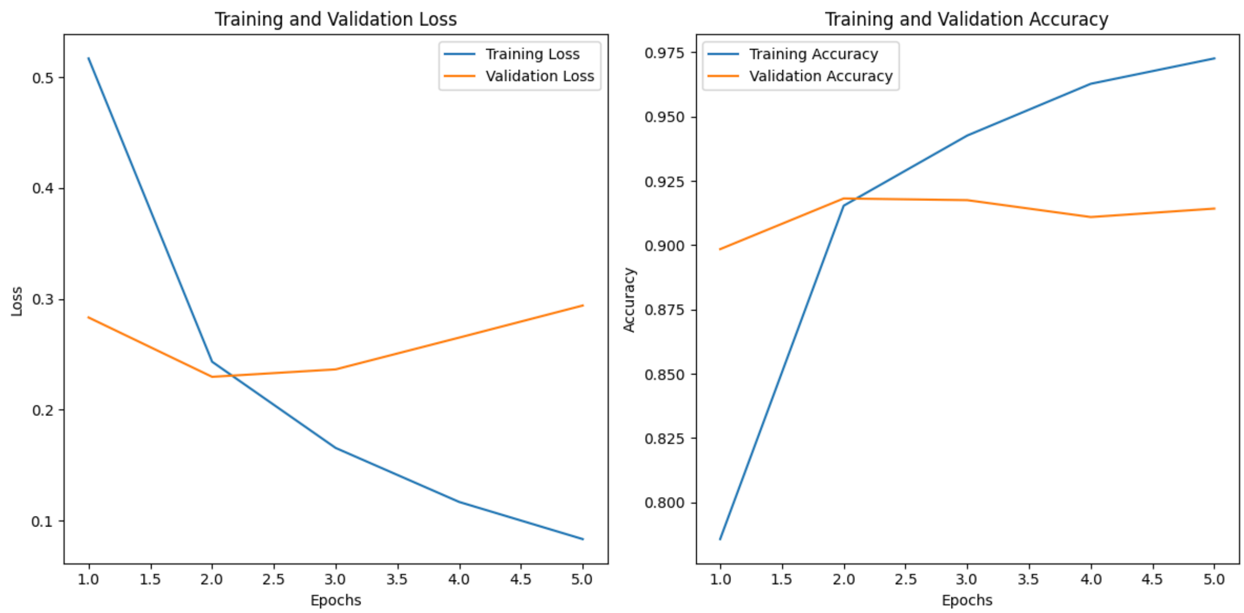


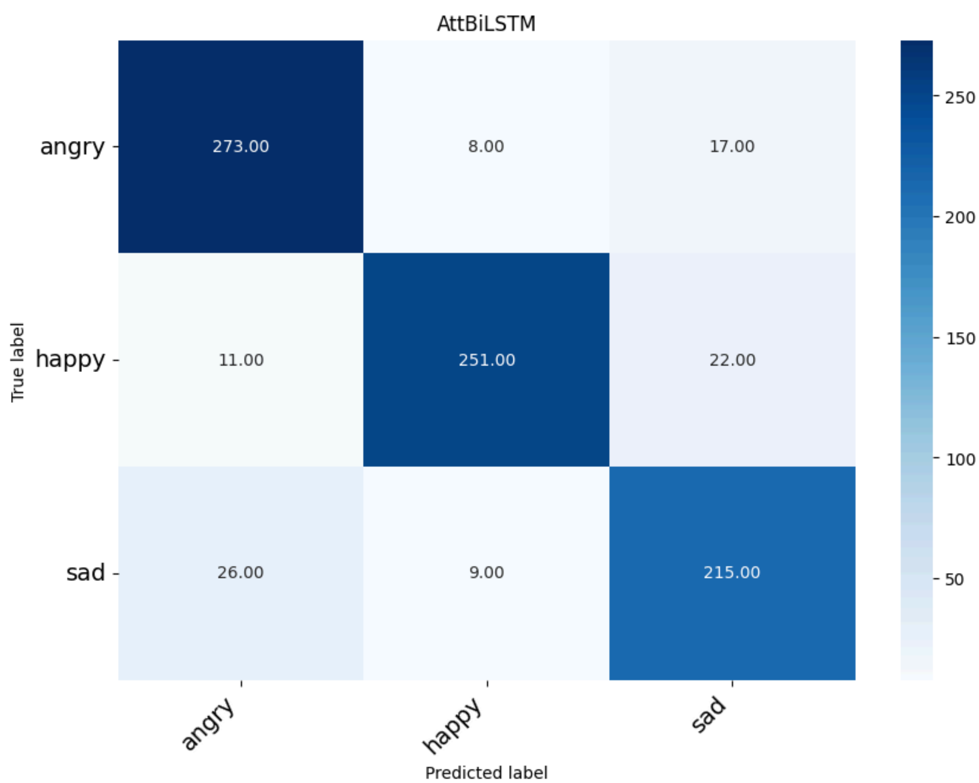
BiLSTM, CROWDFLOWER



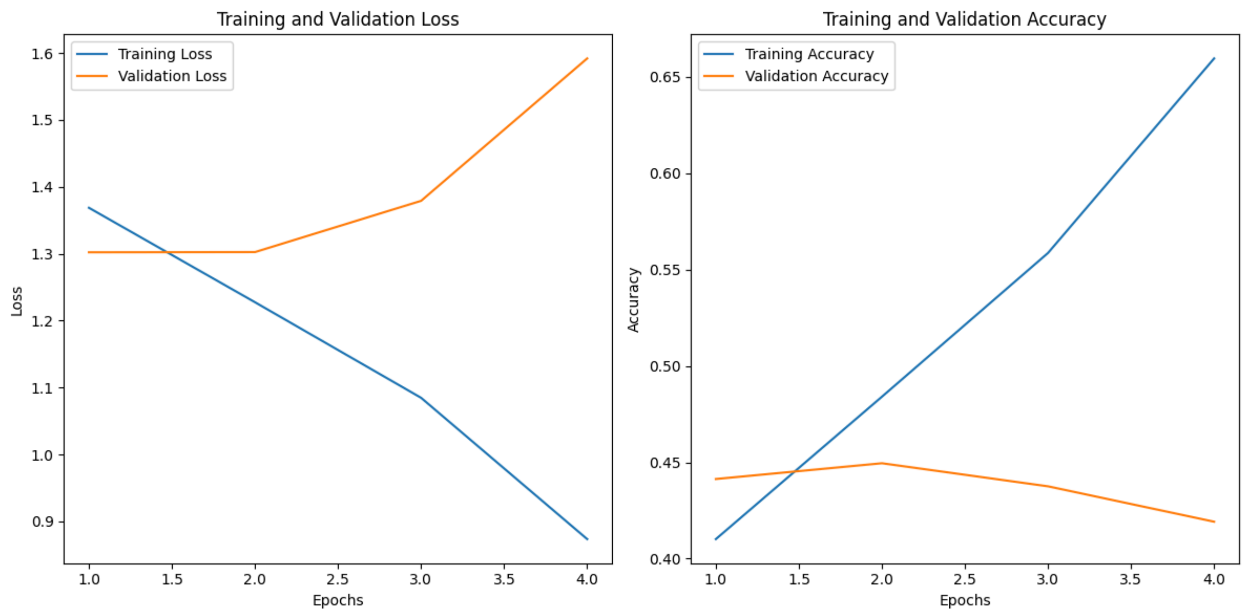


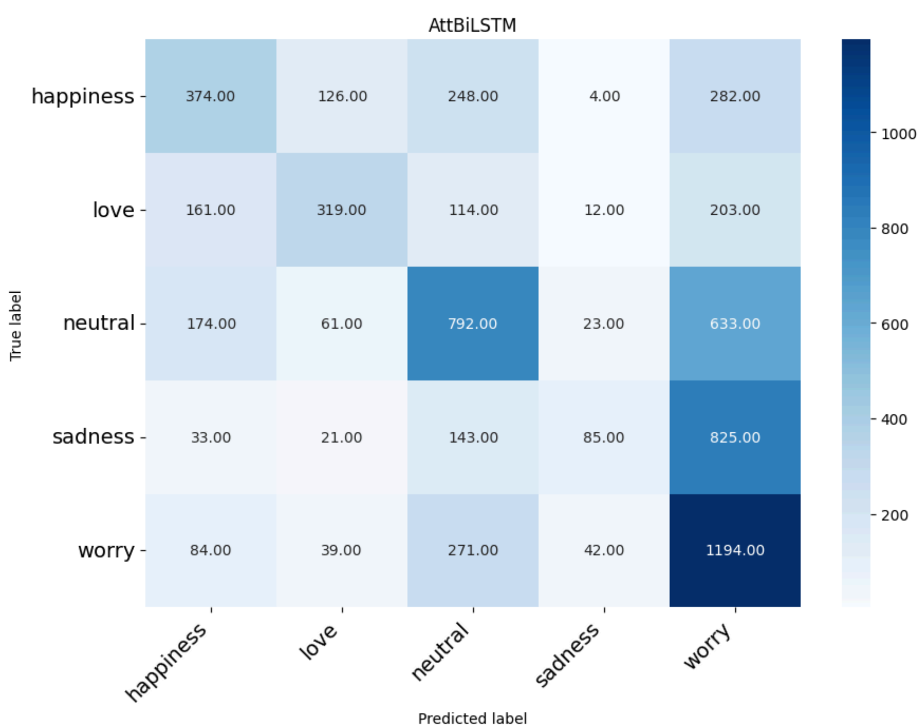
AttBiLSTM, EMO2019



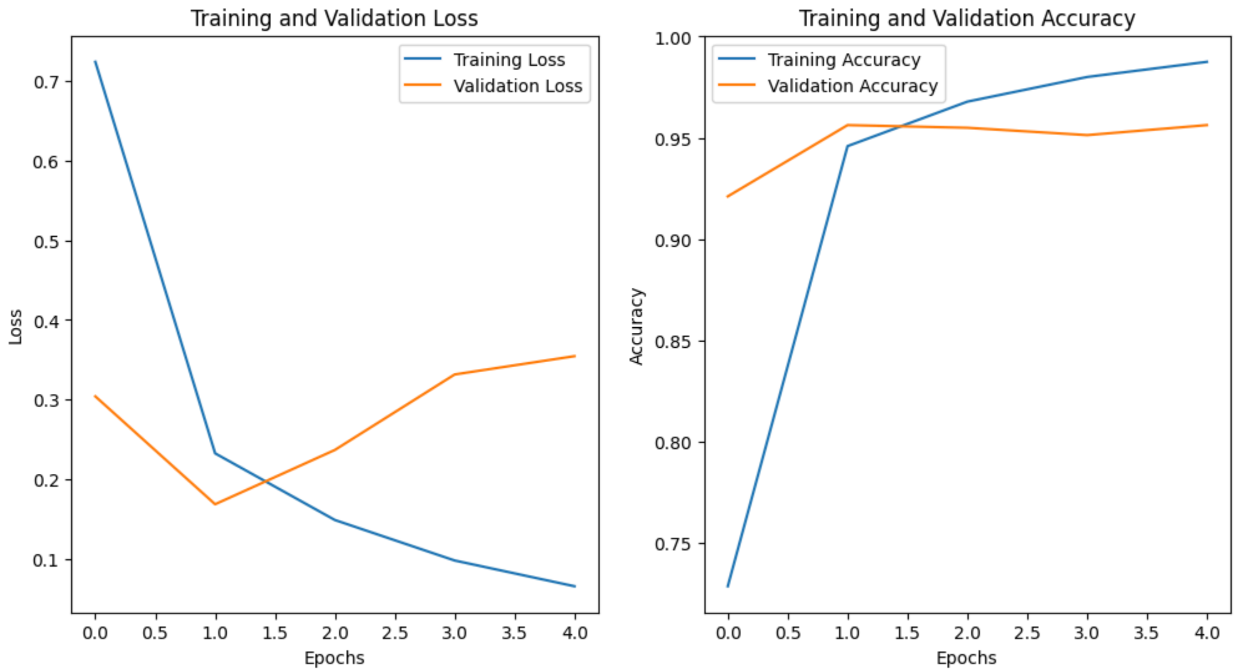


AttBiLSTM, CROWDFLOWER

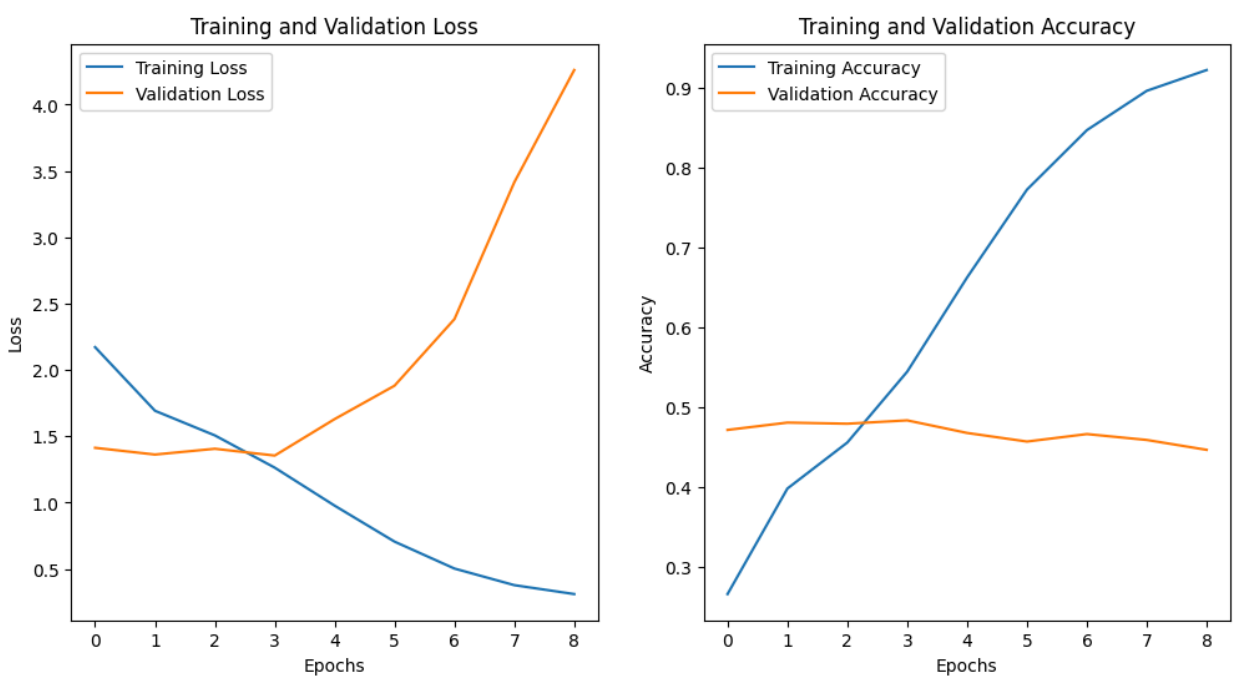




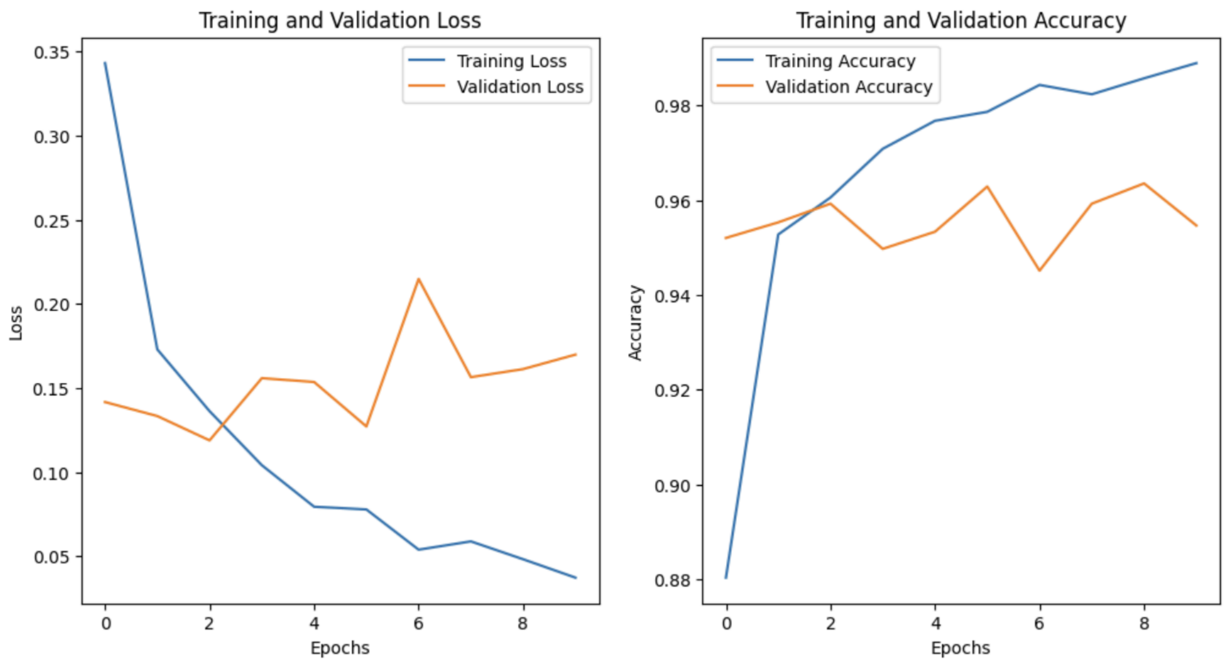
BERT+CNN, EMO2019



BERT+CNN, CROWDFLOWER



RoBERTa + CNN, EMO2019



RoBERTa + CNN, CROWDFLOWER