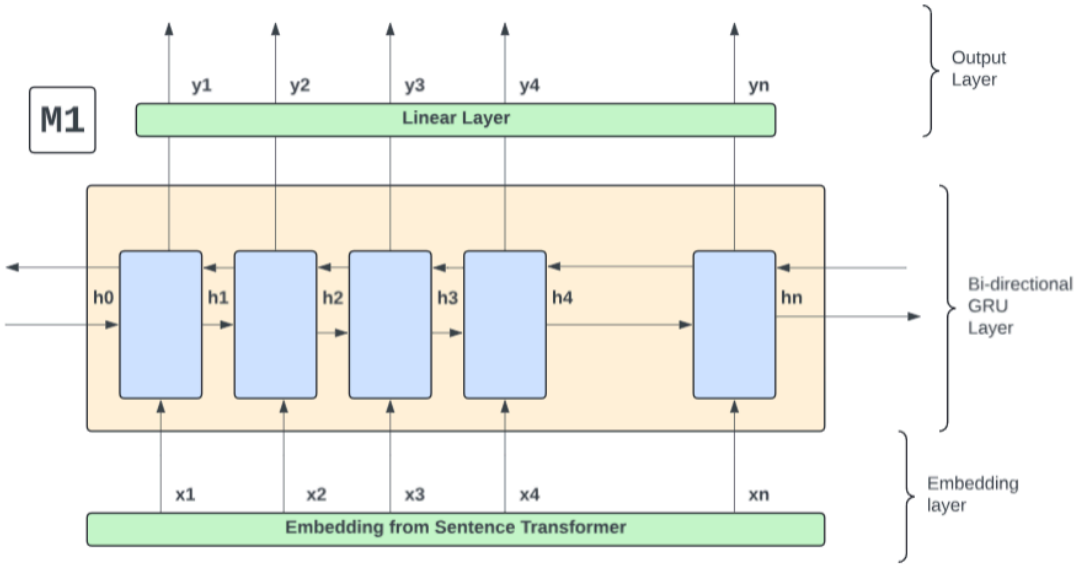
NLP Assignment 4 Report

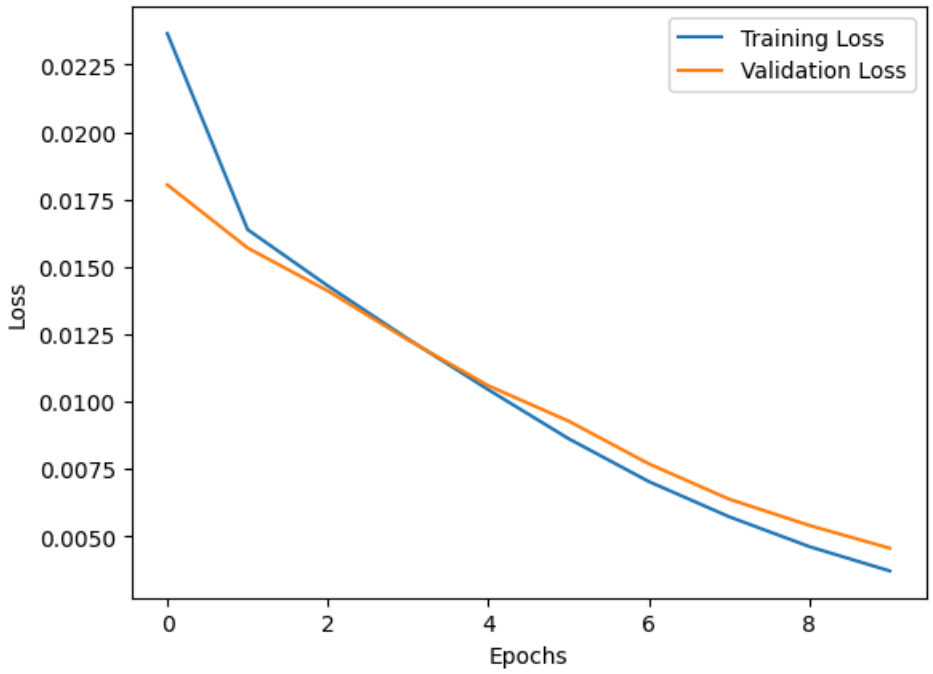
TASK 1 - ERC (Emotion Recognition in conversation)

1. M1 - Model Architecture

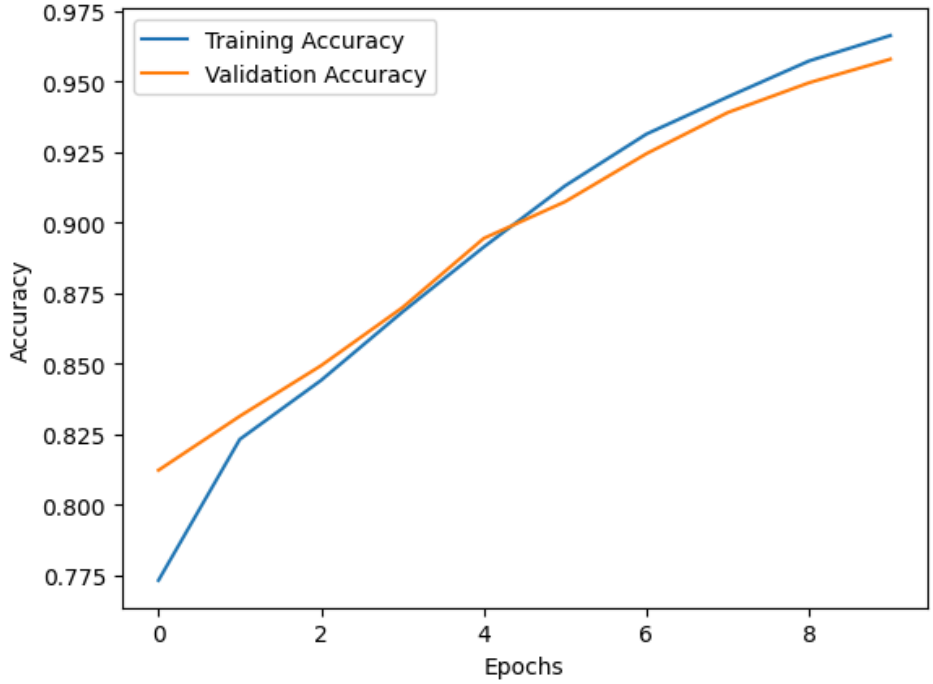


2. Plots:

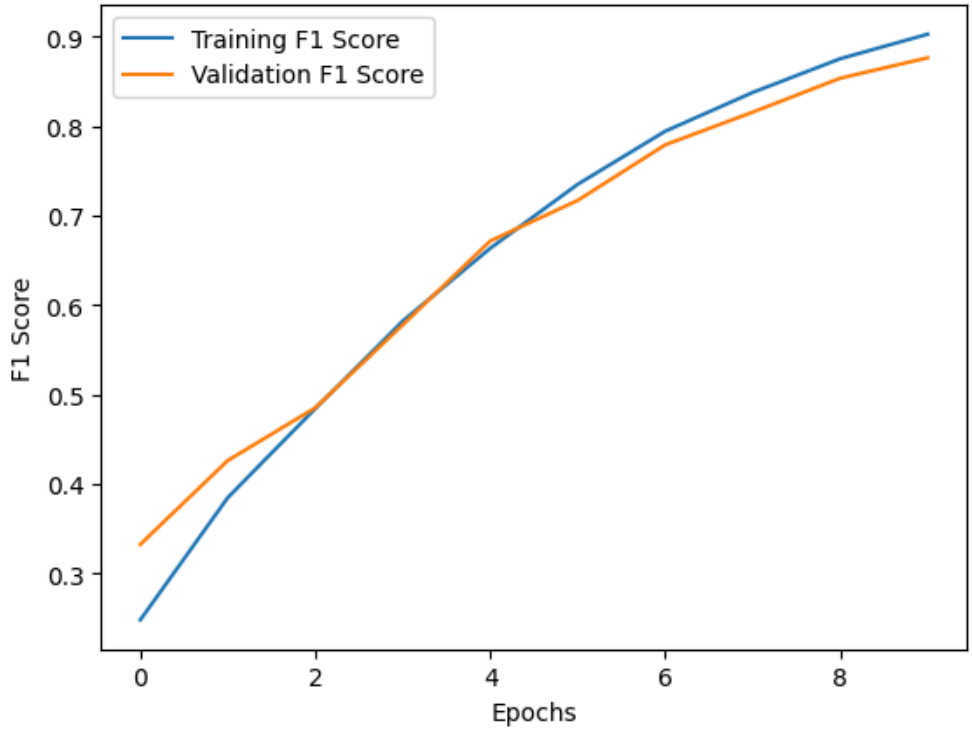
Training Loss vs Validation Loss



Training Accuracy vs Validation Accuracy



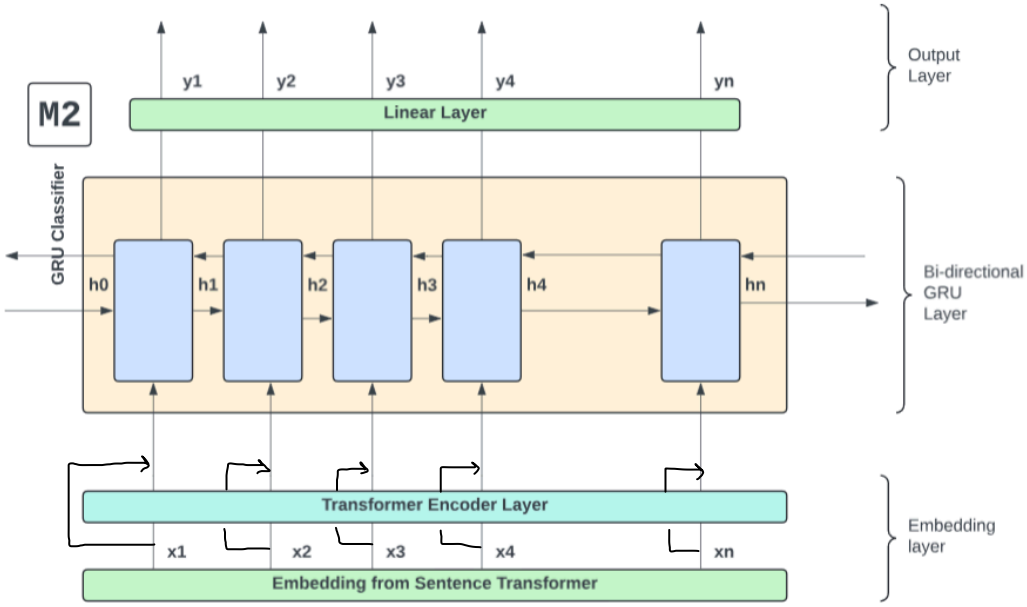
Training F1 Score vs Validation F1 Score



3. **The intuition behind the models, splits and everything relevant**
   No particular data split is used. We have used all of the training data for training and validation data for evaluation purposes. We have utilized only the utterances in the dialogue conversation to perform our emotion classification task.

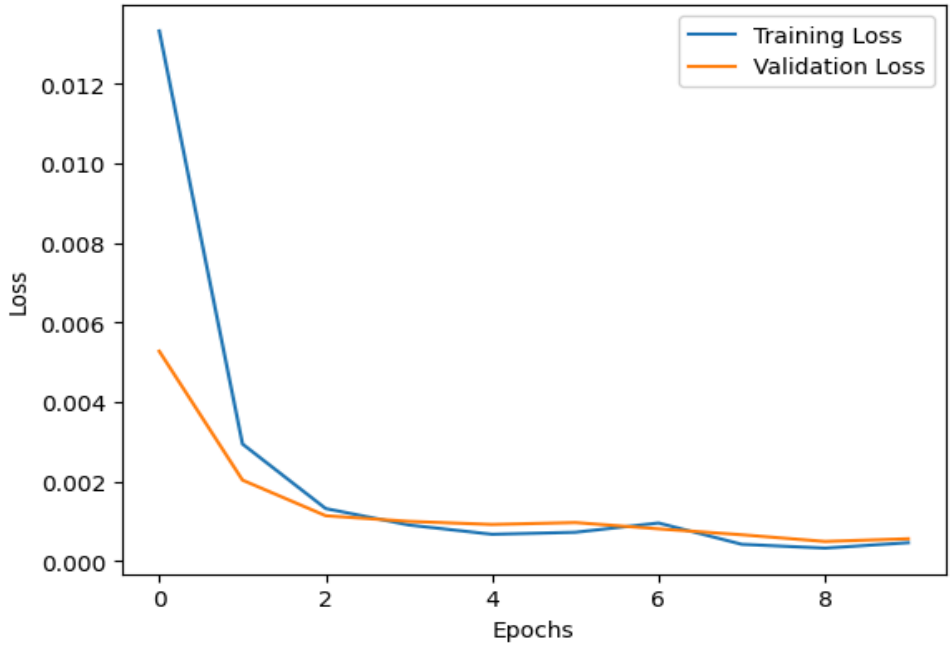
We thought of using some pre-trained transformers to obtain word embeddings of all utterances. We felt that by using pre-trained transformers, we could get efficient embeddings to represent the utterance rather than training a custom Embedding layer for the same. We used the '`all-MiniLM-L6-v2`' Sentence Transformer for this purpose. We then used a model architecture consisting of a **bi-directional GRU Classifier** and an **output linear layer** with 7 classes to do the emotion classification task.

4. M2 - Model Architecture

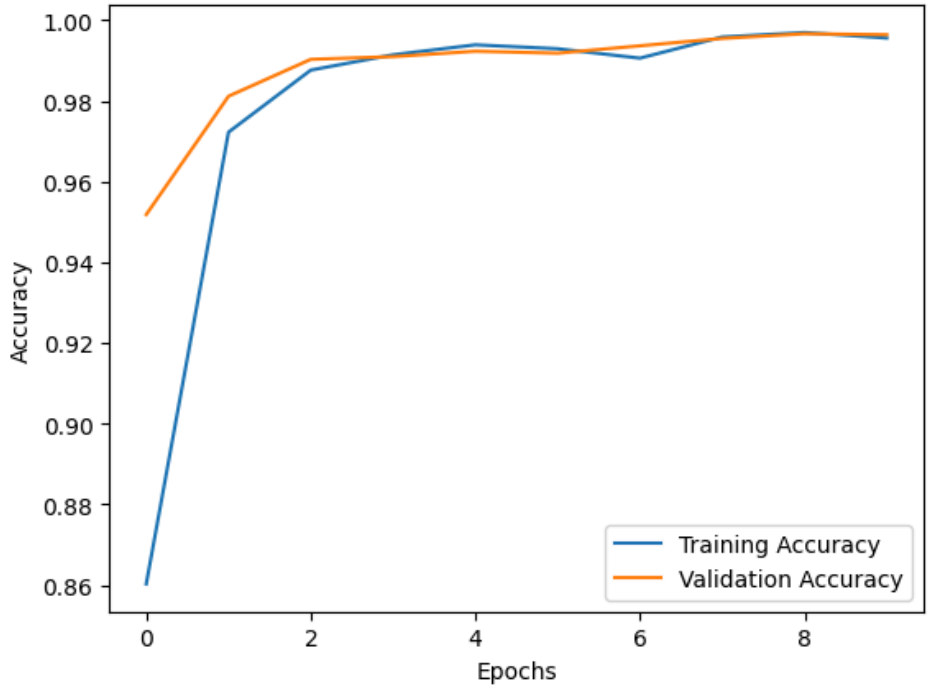


5. Plots

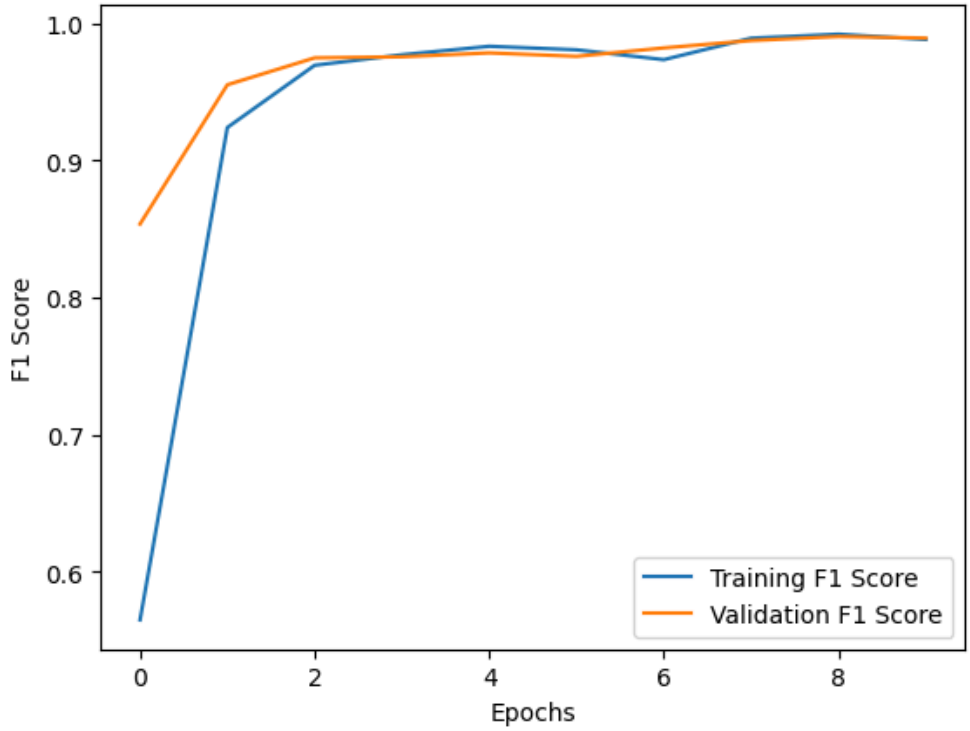
Training Loss vs Validation Loss



Training Accuracy vs Validation Accuracy



Training F1 Score vs Validation F1 Score



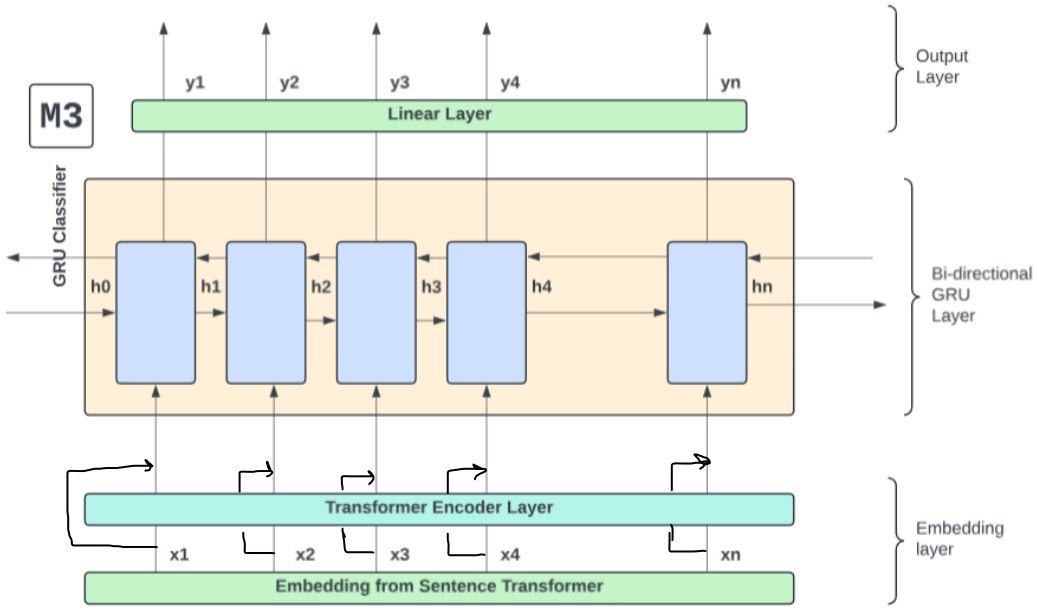
6. **The intuition behind the models, splits and everything**
   No particular data split is used. We have used all of the training data for training and validation data for evaluation purposes. We have utilized only the utterances in the dialogue conversation to perform our emotion classification task.

   We thought of using some pre-trained transformers to obtain word embeddings of all utterances. We felt that by using pre-trained transformers, we could get efficient embeddings to represent the utterance rather than training a custom Embedding layer. We used the '`all-MiniLM-L6-v2`' Sentence Transformer for this purpose.
   We then used a model architecture consisting of a **Transformer Encoder with a residual connection** (to allow the gradient to flow back for good learning) followed by a **Bi-directional GRU recurrent** model and **one output linear layer** with 7 classes.

7. **Which model was better and why**
   In Model 2, we used a Transformer Encoder with a residual connection and the GRU network. Intuitively, we know that the Transformer Encoder involves using a self-attention mechanism to identify and extract the relevant text from the utterances required to classify emotions. So, we expect the M2 model to perform better than the M1 model,, which only uses a single GRU layer. Practically, the M2 model met our expectations, and it indeed performed better than the M1 model.
   We observed that Model M2 achieved **0.989 macro F1** and **0.996 weighted F1** on the Validation set while Model M1 made up of only BiGRU layer achieved **0.876 macro F1** and **0.957 weighted F1** on the Validation set.

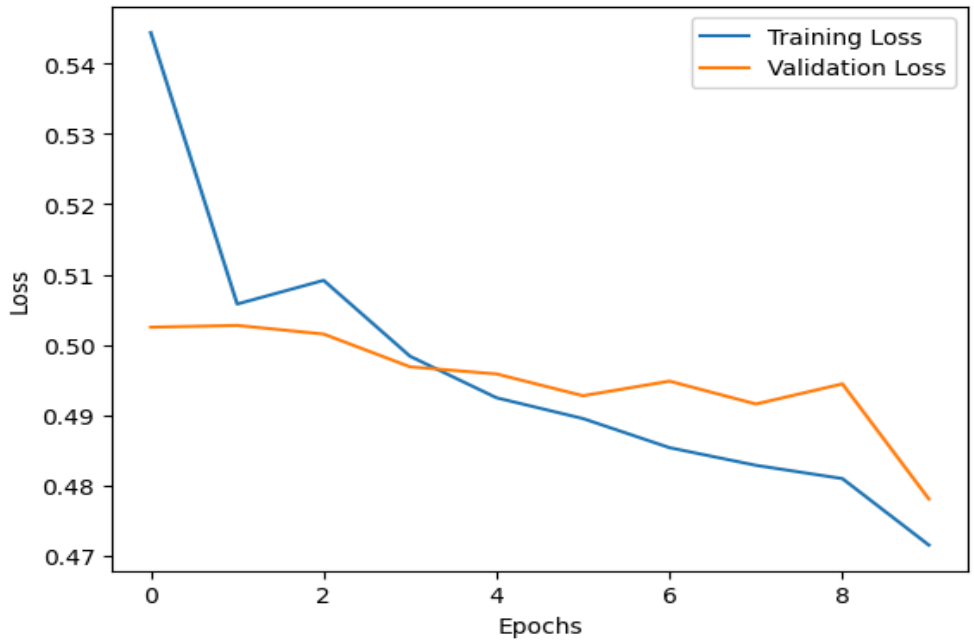
TASK 2 - EFR (Emotion Flip Reasoning)

1. M3 - Model Architecture

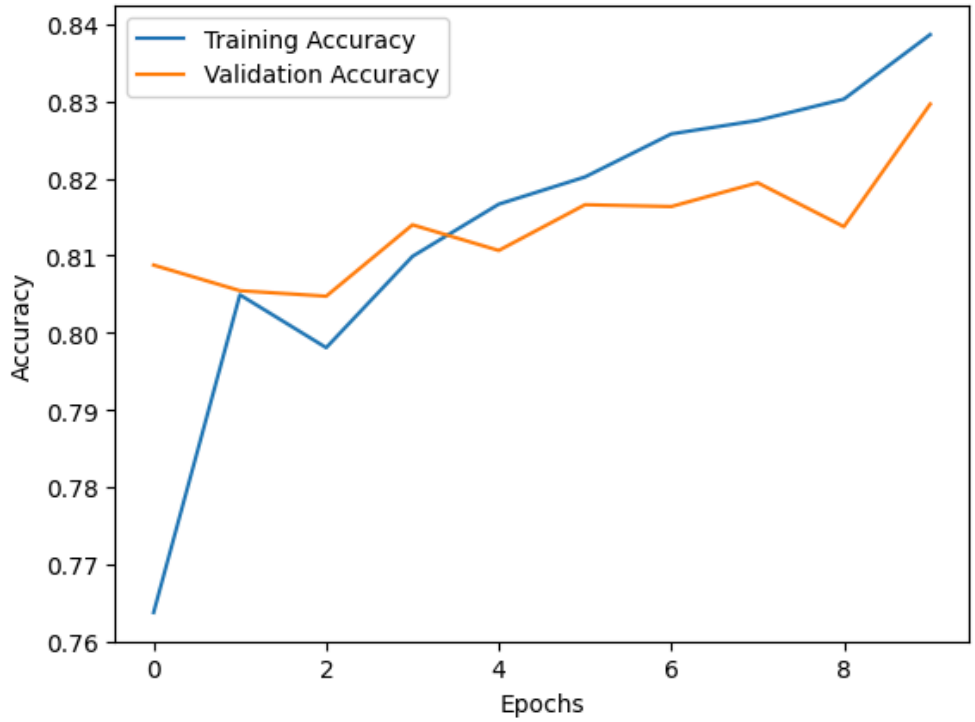


2. Plots

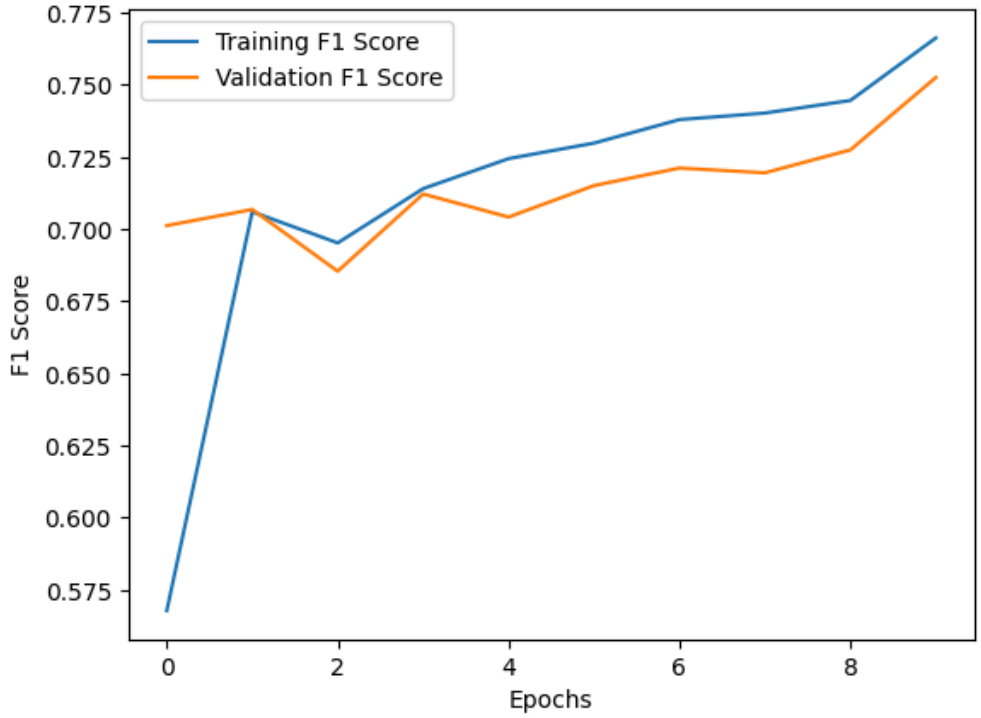
Training Loss vs Validation Loss



Training Accuracy vs Validation Accuracy



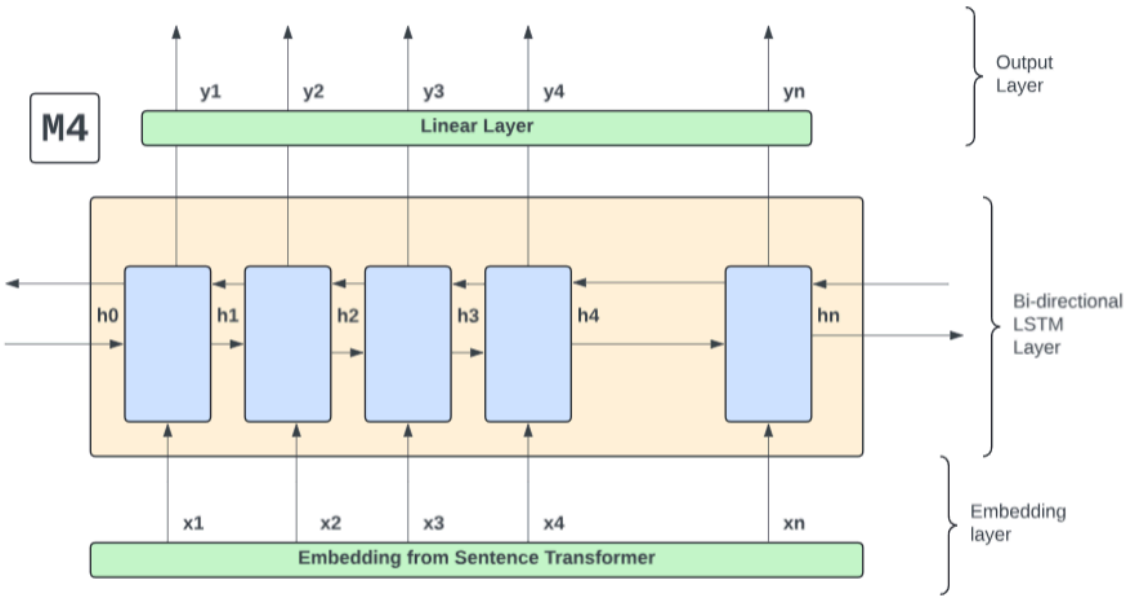
Training F1 Score vs Validation F1 Score



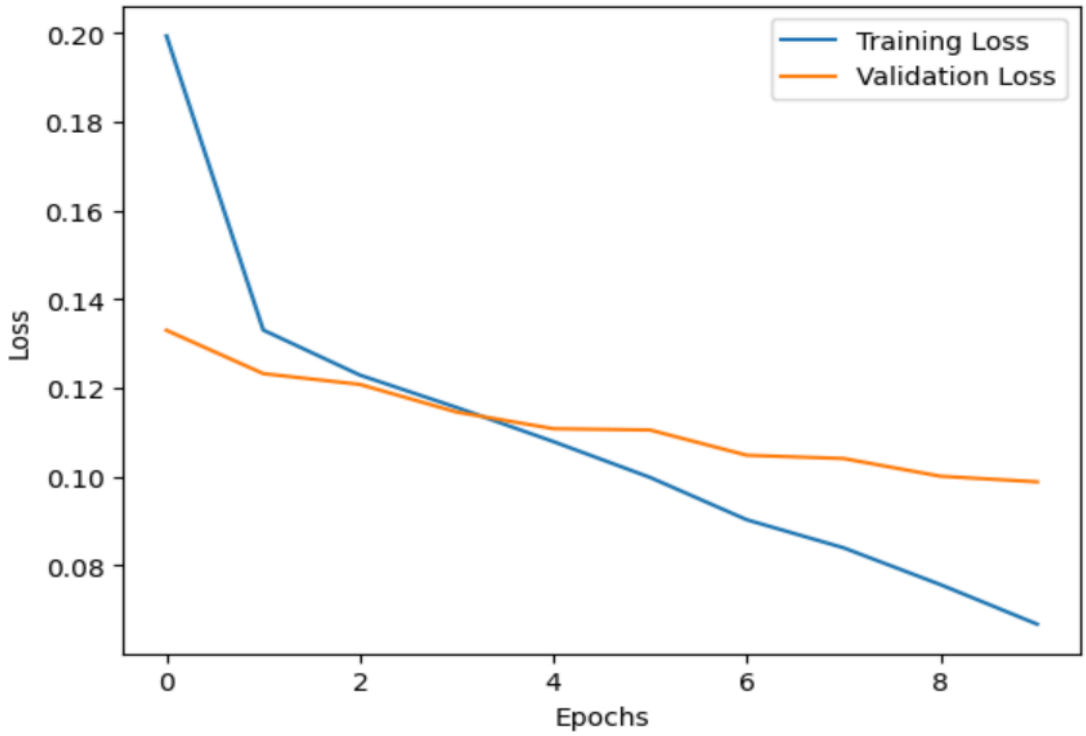
3. **The intuition behind the models, splits and everything**
   We have used all of the training data for training and validation data for evaluation

Purposes. We have utilized only the utterance and its embeddings to perform the EFR task. Also, we have restricted the context size to 5 within a dialogue to accomplish the EFR task (as this was also used in the paper, so we followed the same intuition and restricted it to 5). Also, in this task, we used the last utterance in the dialogue as the target utterance for EFR; therefore, I decided to use utterance embeddings with some combinations of the last utterance embedding in the dialogue.
So, we added the embedding of the last utterance in the dialogue to all the utterances embedding in the context size to generate some combinations. We figured out that using this particular combination, we were able to perform better in this EFR task (as compared to the case of utterance embeddings without combining the embedding of last utterance).
We thought of using some pre-trained transformers to obtain word embeddings of all utterances. We felt that using pre-trained transformers could get efficient embeddings to represent the utterance rather than training a custom Embedding layer. We used the '`all-MiniLM-L6-v2`' Sentence Transformer for this purpose.
We then used a model architecture consisting of a **Transformer Encoder with a residual connection** (to allow the gradient to flow back for good learning) followed by a **Bi-directional GRU recurrent** model and **one output linear layer** with 2 classes.

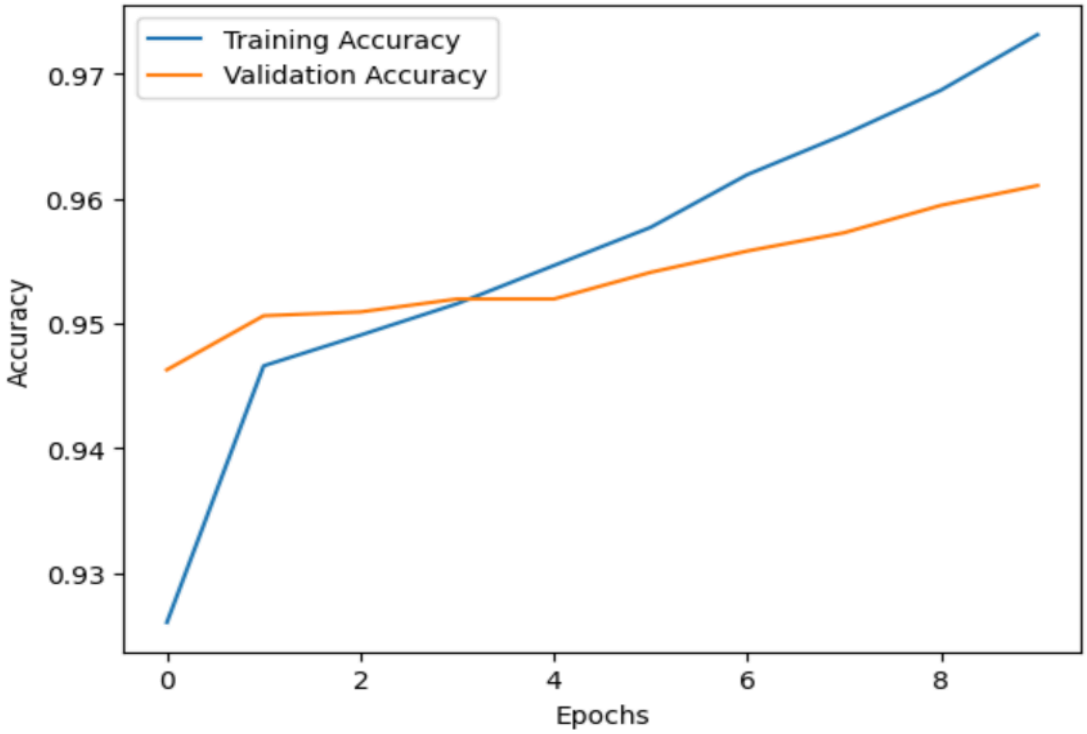
4. **M4 - Model Architecture**



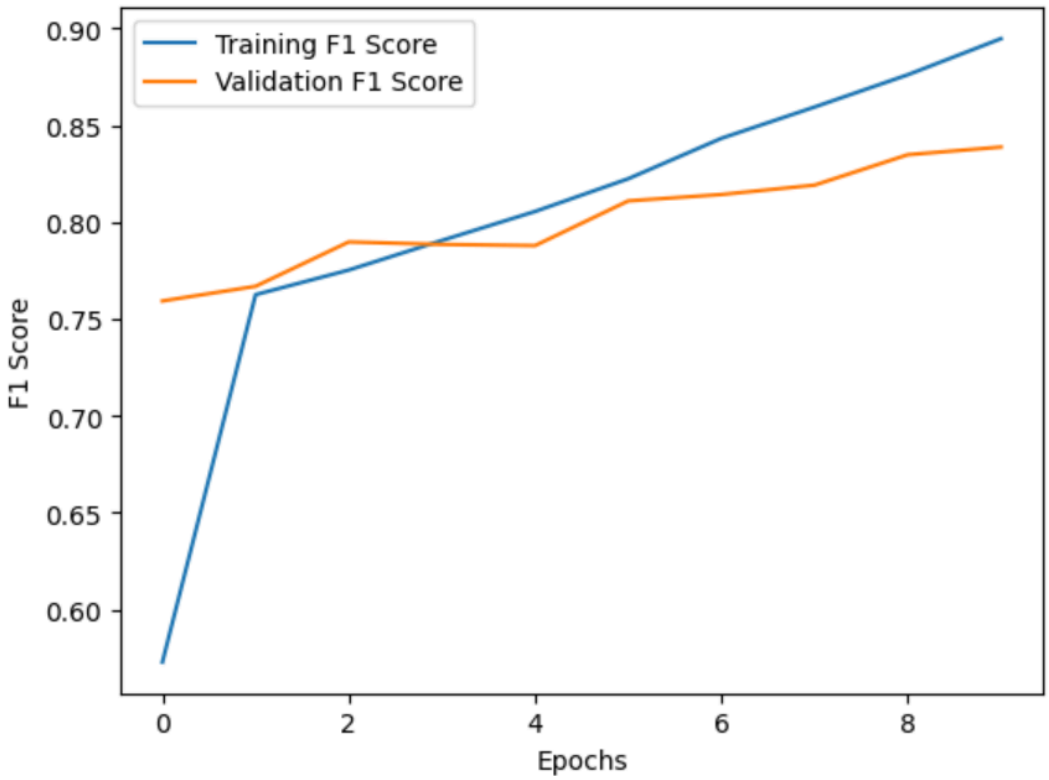
5. **Plots**
   Training Loss vs Validation Loss



Training Accuracy vs Validation Accuracy



Training F1 vs Validation F1



6. **The intuition behind the models, splits and everything**
   We have used all of the training data for training and validation data for evaluation Purposes. We have utilized only the utterance and its embeddings to perform the EFR task. Unlike in the Model M3, here we have used the whole diaglogue (all the utterances in the dialogue).
   We used the last utterance in the dialogue as the target utterance for EFR. Unlike the Model M3 we haven't used any combinations in the utterance embeddings this time for Model M4.
   We thought of using some pre-trained transformers to obtain word embeddings of all utterances. We felt that using pre-trained transformers could get efficient embeddings to represent the utterance rather than training a custom Embedding layer. We used the '`all-mpnet-base-v2`' Sentence Transformer for this purpose.
   This time, we then used a model architecture consisting of only a **Bi-directional LSTM recurrent** model and **one output linear layer** with 2 classes.

7. **Which Model was better and why**
   Like in Task1 for ERC, we saw the Transformer based GRU model performed better than the only GRU model, due to the fact that the Transformer uses a Self-attention mechanism to learn the relevant inputs responsible for output generation.
   We expected similar results here too. But we observed that Model M4 consisting of only the BiLSTM layer, achieved **0.8386 macro F1** and **0.960 weighted F1** on the Validation set. In comparison Model M3, made up of both Transformer Encoder and BiGRU layer

achieved **0.752 macro F1** and **0.822 weighted F1** on the Validation set. This could be because of the following reasons:

a. We used ‘`all-mpnet-base-v2`‘ Sentence Transformer for utterance embeddings for Model M4, while we used ‘`all-MiniLM-L6-v2’` Sentence Transformer for Model M3. The former is more richer in terms of features and performance (also mentioned in Sentence Transformers official website)
b. The size of embeddings generated by the former ST is also larger than the latter one, which results in more rich features in utterances embeddings for Model M4. This was the prominent reason.

Because of the above reasons, we feel the embeddings generated by the ST in Model M4 is so rich that even the BiLSTM layer can learn it so well.

Contributions:

1. Khushdev Pandit: Model M1, M2, M3, Report Generation
2. Arjun Mehra: Model M1, Report Generation, Architecture diagrams
3. Pankaj: Model M4
4. Apurv Dube: Attempted Model M4