Sentiment Analysis on Video Feeds Using Classified Textual and Audio Inputs

John Markowicz, Eric Gaudreau

Department of Data Science, Boston University

DS 340: Introduction to Machine Learning and AI

Kevin Gold

4/29/2024

Introduction

Our topic was reaction analysis on labeled text and audio inputs. We are interested in exploring the applications machine learning algorithms have for representing audio and textual data and the ability of different models to dissect the complex layers of human expression. By expanding on and improving upon the methodology presented, we foresee this technology being impactful in the field of education by helping people learn emotion recognition in their interpersonal dynamics with real-time feedback. Alternatively it could be used in customer feedback analysis to categorize feedback and identify areas for improvement with customers or it could be used as a virtual voice acting coach, providing actors with real-time feedback on their vocal performance.

Our data set, referenced as MELD, comprises over 10,000 dialogues and utterances extracted from the Friends TV series, featuring diverse characters to ensure the models were trained on a variety of voice tones. Additionally, it included labels indicating specific emotions (anger, disgust, sadness, joy, neutral, surprise, and fear) and general sentiment (positive, negative, and neutral) for each video clip as well as the corresponding textual transcription. We decided to investigate if our models were more effective at recognizing general sentiment, which comprised only three classes with an equal number of data in each class as compared to emotion for which the data was skewed. This expanded our total number of models, allowing us to compare their accuracies and analyze the difference in results.

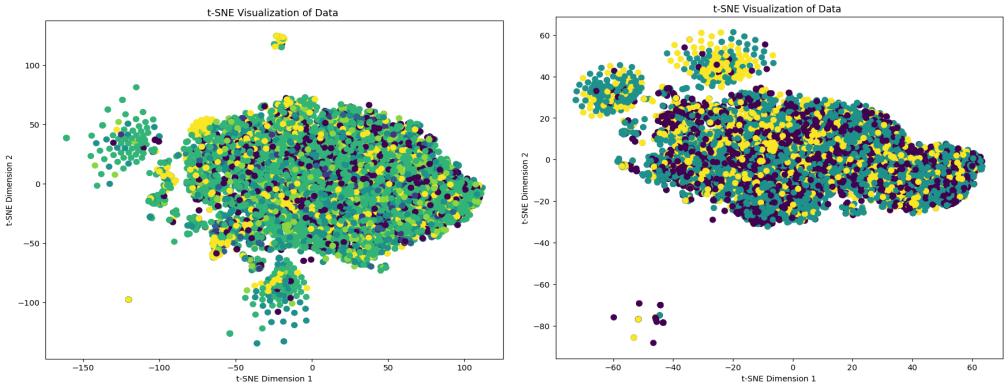
```
Emotion
neutral     4906
joy         1722
surprise    1272
anger       1243
sadness      783
disgust      287
fear         276
Name: count, dtype: int64 Sentiment
neutral     2310
positive    2310
negative    2310
Name: count, dtype: int64
```

Methodology

Feature Extraction

To represent our data in a way ML models could process, we first cleaned the data to strengthen contextual meaning. For our textual data, preprocessing involved lemmatization to remove word tense, and the removal of stopwords to eliminate terms irrelevant to semantic meaning. BERT was utilized for extracting text features, or dimension representations of data in vector space, which resulted in the textual dimensions of (6927, 76, 768). An audio Spectrogram transformer from Hugging Face was used to extract the audio features. We utilized a transformer resampler to scale our audio from 48HZ to 16HZ. The feature dimensions for audio data were (6927, 1024, 128). These feature embeddings were then used for inputs to each of our proposed models enabling the capturing of dependencies within the data.

To visualize the extracted text features we used T-SNE, which is an algorithm used for representing high-dimensional data in a lower-dimensional space. It minimizes the Kullback-Leibler (KL) divergence between the original high-dimensional distributions and a lower-dimensional t-distribution, effectively preserving the local structure of the data points.



Specific Emotion General Sentiment

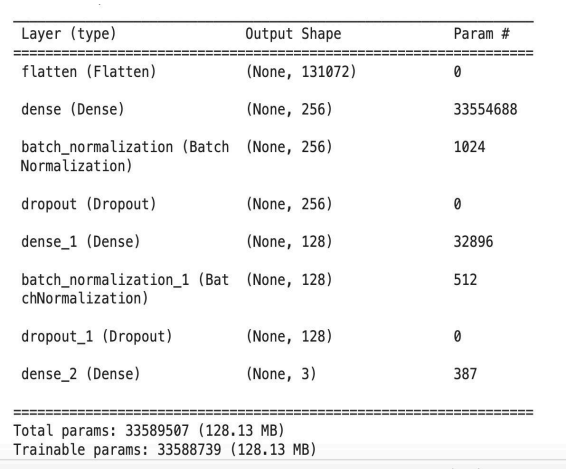
A challenge presented when visualizing the results of T-SNE is the overlap between clusters in the lower-dimensional representation. As mentioned later in the results section, our intuition as to why XGBoost and logistic regression achieved roughly 50% accuracy is imaging a linear boundary through the visualized data; given the overlap, it is possible to imagine 50% of the predicted class falling on one side of the boundary and 50% falling on the other.

Choosing Models

We selected a feed-forward ANN to model our audio feature embeddings, while logistic regression and XGBoost were used to model textual feature embeddings as baseline models. We hypothesized that a bidirectional LSTM with stacking would outperform the baseline models for the textual data. Regarding the feed-forward neural network for audio, this choice was motivated by its capability to capture non-linear relationships in the audio data. Feed-forward neural networks are adept at learning complex patterns in data and can effectively model the acoustic features present in audio signals, making them suitable for audio-based tasks. We chose logistic regression and XGBoost to model text features because they are well-established, computationally efficient, and effective for classification tasks, especially when dealing with high-dimensional data. The decision to implement a bidirectional LSTM for text features was based on its capability to capture contextual information from both past and future words in a sentence, which can enhance the model's understanding of text semantics and improve performance in emotion recognition tasks.

So in total, we developed 8 models; four models for predicting general sentiment, and 4 models for specific emotion (Text: XGBoost, logistic regression, bidirectional LSTM neural network; Audio: Feed forward neural network).

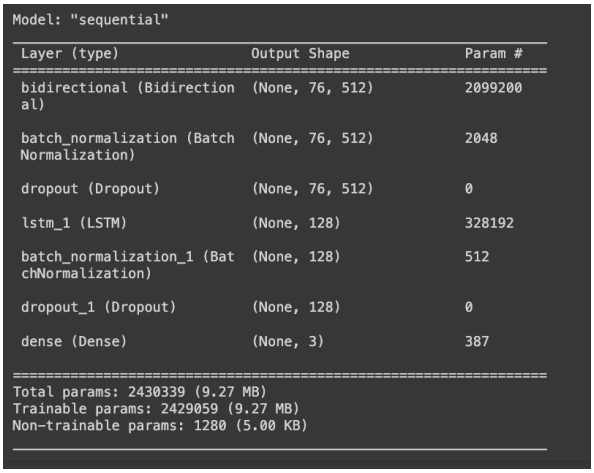
For each deep learning model, we implemented batch normalization and dropout to combat overfitting and reduce model complexity.



Layer (type)	Output Shape	Param #
flatten (Flatten)	(None, 131072)	0
dense (Dense)	(None, 256)	33554688
batch_normalization (Batch Normalization)	(None, 256)	1024
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 128)	32896
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 3)	387

Total params: 33589507 (128.13 MB)
Trainable params: 33588739 (128.13 MB)

Feed Forward ANN (Audio Model)



Model: "sequential"

Layer (type)	Output Shape	Param #
bidirectional (Bidirectional)	(None, 76, 512)	2099200
batch_normalization (Batch Normalization)	(None, 76, 512)	2048
dropout (Dropout)	(None, 76, 512)	0
lstm_1 (LSTM)	(None, 128)	328192
batch_normalization_1 (BatchNormalization)	(None, 128)	512
dropout_1 (Dropout)	(None, 128)	0
dense (Dense)	(None, 3)	387

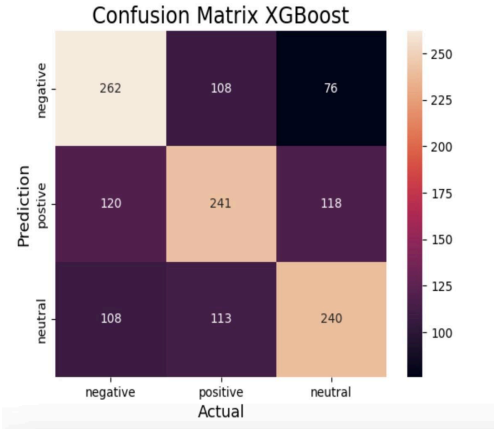
Total params: 2430339 (9.27 MB)
Trainable params: 2429059 (9.27 MB)
Non-trainable params: 1280 (5.00 KB)

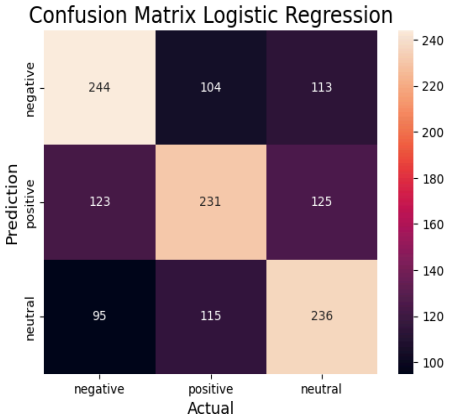
BiDirectional LSTM (Text Model)

Results

We used confusion matrices to assess the accuracy of our XGBoost and logistic regression models. These matrices provided a comprehensive visualization of the model's performance by displaying the true/false positives/negatives. For the text and audio neural networks we evaluated the training and validation loss and accuracy for every epoch. Higher accuracy and lower loss on validation set typically means the model is improving.

Predicting General Sentiment

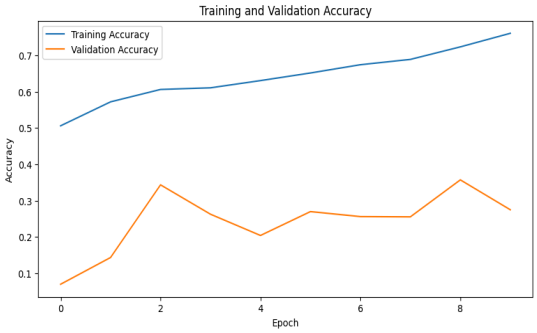




Accuracy: .52

Accuracy: .512

BiDirectional LSTM for Sentiment Text Features

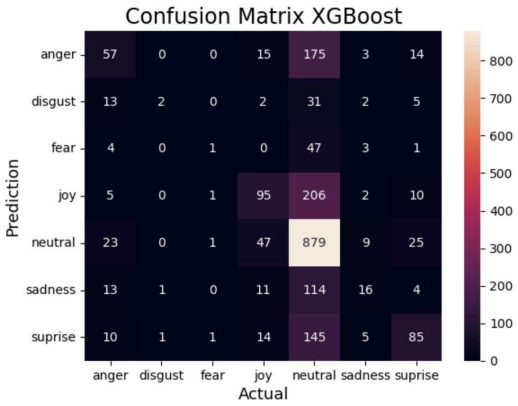


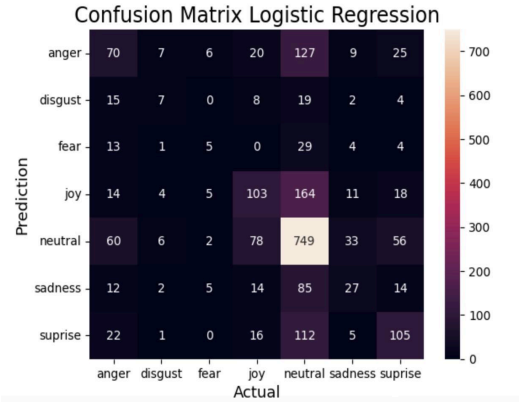
Feed Forward Neural Network for Sentiment Audio Features

We observed XGBoost outperformed logistic regression on each test set, and negative sentiment was the most frequently correct prediction, likely due to its clear distinction in linguistic cues.

In the case of our deep learning models, they did not outperform the baseline models, as there are signs of overfitting with validation loss increasing as training loss decreases. However, we theorize the answer to this occurrence may be explained by the audio ANN results. As shown in the ANN results, there is a large jump in validation accuracy from 20% to over 70%. Given all training data was based on high dimensional transformer feature embeddings, we expect the loss function surface to be highly complex meaning there is a high potential for the model to fall into a local minima. We believe that if implemented with a lower learning rate and large number of epochs, which would require more computational power not available to us during this project, the LSTM model would easily outperform the baseline models.

Predicting Specific Emotion

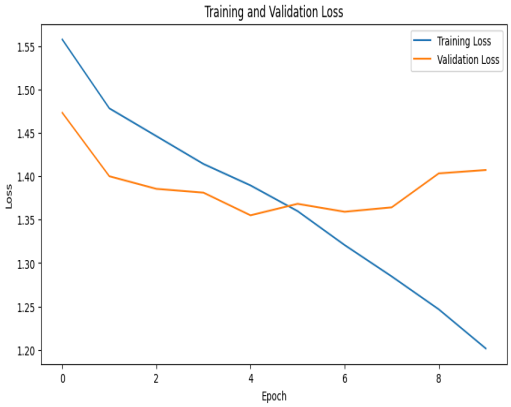


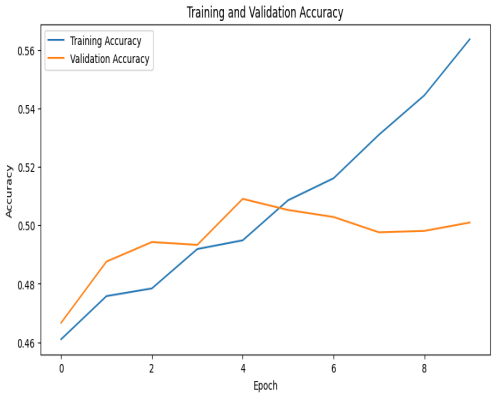


Accuracy: 0.54

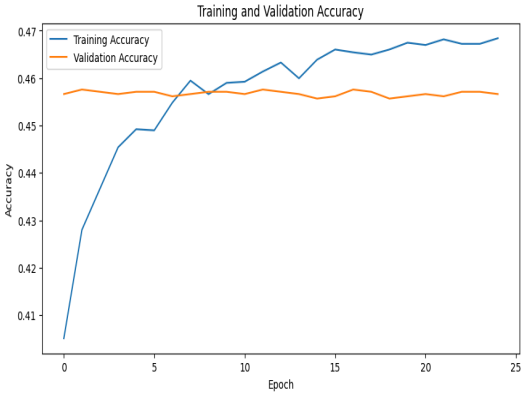
Accuracy: 0.502

BiDirectional LSTM for Emotion Text Features





Feed Forward Neural Network for Emotion Audio Features



Similar to general sentiment, XGBoost and logistic regression were able to distinguish intense emotions such as anger, surprise, and joy with relatively high accuracy. Given our skewed dataset, each emotion was confused most often with neutral, however, fear, disgust and sadness were confused often with anger as these have analogous methods of expression.

For the deep learning models, each achieved roughly the same level at 50% accuracy. This is likely explained by the fact that the training data was highly biased towards the neutral class, making it quite easy to learn the boundaries for neutral and possibly generalize slightly to the other emotion labels. As previously discussed, due to the intricate nature of the training data from transformer feature embeddings, the loss function is very complex, and has a heightened risk of the model converging to local minima. We assume that, based on the increasing training accuracy, with more epochs and a smaller learning rate we would see an increase in the accuracy of these models.

Conclusion

To assess our models' performance in terms of comparing them to human capabilities in recognizing specific emotions, we conducted a limited evaluation on a subset of 30 audio clips, where we attempted to guess the associated emotion after viewing the video clip. From this evaluation, we achieved a score of 16 out of 30 correct predictions, resulting in an accuracy of approximately fifty-three percent. This indicates that our best-performing models performed comparably to our capabilities in recognizing specific emotions.

During this project, we learned deep learning models have high computational demands, particularly when handling high-dimensional data. Surprisingly, XGBoost and logistic regression demonstrated comparable accuracy to these deep learning models, showing their significance as powerful baseline models because they were very easy to fit into complex data. Perhaps if the data is too complex, it is better to go with a more simple or reliable model. We also learned that transformer embeddings are extremely complex and their loss function surface is difficult to learn with limited computational power.

To address these limitations and enhance future iterations of our work, we propose several ideas. Firstly, exploring data augmentation techniques that support a balanced training dataset. Second, scale the dimensions of feature embeddings to correspond with the level of compute available. Our study shows the importance of considering model complexity, leveraging baseline models, and addressing limitations such as data skewness and representativeness in future research endeavors aiming to improve emotion and sentiment recognition tasks.

Works Cited

Audio Spectrogram Transformer, huggingface.co/docs/transformers/en/model_doc/audio-spectrogram-transformer. Accessed 23 Apr. 2024.

Dave Cote, M.Sc. “Hybrid (Multimodal) Neural Network Architecture : Combination of Tabular, Textual and Image Inputs...” Medium, Medium, 18 Nov. 2022, medium.com/@dave.cote.msc/hybrid-multimodal-neural-network-architecture-combination-of-tabular-textual-and-image-inputs-7460a4f82a2e.

Hakim, Zaber Ibn Abdul. “Multimodal Emotion lines Dataset (Meld).” Kaggle, 26 Feb. 2021, www.kaggle.com/datasets/zaber666/meld-dataset.

Jalammar. “A Visual Notebook to Using BERT for the First Time.Ipynb.” GitHub, 2021, github.com/jalammar/jalammar.github.io/blob/master/notebooks/bert/A_Visual_Notebook_to_Using_BERT_for_the_First_Time.ipynb.

“Sklearn.Manifold.TSNE.” Scikit, scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html. Accessed 23 Apr. 2024.