Conversational Emotion Recognition through Speech and Textual Data - Final Report

Alex Gentle, Astha Modi, and Shalin Bhavsar

Problem Definition

Humans can best use various social cues in conversation to better gauge the underlying meaning behind words. Much of this nuance is lost when conversing over text, such as email, text messages, or social media. Similarly, the emotion behind the language can be lost when one only analyzes the written text. "I'm sorry" can be said mournfully, sarcastically, or with a dozen other meanings that are not immediately clear without the context that audio and video can provide. In this project, we incorporated text, audio, and video data together in a multimodal model to predict various emotions. We also tested a variety of classifiers that are able to recognize emotions from textual, audio, and video data.

Literature Review

During the course of the project, we referenced several research papers. Since we trained our model using IEMOCAP, one particularly useful paper was Multi-Modal Emotion Recognition on IEMOCAP Using Neural Networks by Tripathi et al. [5] Our first attempts at building an emotional recognition model began by learning from and implementing different ideas explored in this paper. The authors aimed to solve the problem of recognizing emotions from multiple modalities (audio, video, and text) on the IEMOCAP dataset. They used the bag-of-words model and word embeddings as features for text-based emotion recognition; mel-frequency cepstral coefficients (MFCCs) and pitch as features for audio-based emotion recognition; and the pose of the head and shoulders, facial landmarks, and facial action units as features for video-based emotion recognition. They used convolutional neural networks for audio-based emotion recognition, an LSTM network for text-based emotion recognition, and a hybrid CNN-LSTM network for video-based emotion recognition. The authors achieved state-of-the-art performance for happy, sad, angry, and neutral emotional categories with an overall accuracy of 74%.

Several other papers attempted similar models using various techniques and datasets. In the 2018 paper, Computational Modeling of Human Multimodal Language: The MOSEI Dataset and Interpretable Dynamic Fusion Liang et al. [4] worked on the MOSEI dataset and suggested a technique for combining multimodal data by building a computational model for multimodal language analysis. The paper proposed combining textual and speech data to identify conversational sarcasm. The textual features included the bag-of-words representation of the movie transcripts and the embeddings of the words using the GloVe algorithm. The visual features included facial landmarks and Action Units (AUs) extracted from the video frames using the OpenFace toolkit. The acoustic features included the prosodic and spectral features extracted from the audio using the OpenSMILE toolkit. They used a Long Short-Term Memory (LSTM) network to capture the temporal dependencies in the data and a dynamic fusion layer that allows for interpretation by assigning weights to each modality. They built a model of recurrent neural networks that could train from the changes in pitch and tone of the speech data and combine it with the textual data to increase the robustness of the model and improve the accuracy of predicting sarcasm tones in conversations.

A more recent paper, Multimodal Speech Emotion Recognition and Classification Using Convolutional Neural Network Techniques by Christy et al [2], addressed the problem of speech emotion recognition using multimodal techniques. They focused on applying convolutional neural networks to the IEMOCAP data and have collected audio and textual data from several sources, such as the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) and the Interactive Emotional Dyadic Motion Capture (IEMOCAP) database, the Semaine Database. They have used sentiment scores as linguistic features, mel-frequency cepstral coefficients (MFCC) as audio features, and facial action units (FAUs) as visual features. The authors combined these features and tested several deep learning models, including convolutional neural networks (CNNs) and long short-term memory networks (LSTMs). The paper reported around 70% accuracy in recognizing emotions from the combined modalities. The authors also conducted experiments to investigate the contribution of each modality to the overall accuracy and found that audio features were the most effective for emotion recognition.

Dataset

We used the Interactive Emotional Dyadic Motion Capture (IEMOCAP) dataset for this project. It was an ideal choice for what we wanted to accomplish with this project, with audio, video, transcripts, and associated tagged emotions. The emotions were broken down into categorical labels. There are nine emotional labels: anger, happiness, excitement, sadness, frustration, fear, surprise, other, and the neutral state. There are three-dimensional attributes: valence, activation, and dominance. Specifically, there are five male and five female actors reading from a variety of scripts as well as improvising to capture a multitude of emotions. To capture a range of different utterances, each phrase is spoken by three separate actors. Finally, the videos are tagged with motion capture information, including head movements and angles.

Method

A. **Data Preprocessing**: To stay consistent with prior work on text and audio data, data labeled as “excited” was relabeled to “happiness”. Data labeled with the “xxx” or “other” emotions were dropped. We also removed emotions such as “fear”, “disappointment”, “surprise”, and “frustration”. No oversampling was performed since there were roughly equal numbers of each remaining emotion.
    a. **Text Data:** The textual data still needed some data cleaning, performed as follows. We first used the unicodeToAscii function, which removed all diacritic marks from the text data using the normalize function from the unicodedata library. Then, converted the text to lowercase, removed trailing and leading spaces, replaced periods, exclamation marks, and question marks with a space, and removed any non-letter characters except for the specified punctuation marks.
    b. **Audio Data:** Audio noise-cleaning wasn’t needed.
    c. **Motion Capture Data**: As IEMOCAP already came with motion capture data, we opted to use this instead of recreating video data.

B. Feature Extraction

a. **Text Data**: We used TfidfVectorizer to convert the text into a numerical representation. We did this by counting the number of times a word appears in each text, and assigning a weight to each word based on how frequently it appears across all texts. The labels for each text (the emotion category it belongs to) are stored in the labels array. We also used Word Embedding, wherein the text data was first split into individual words and converted into a list of word lists. The Word2Vec model is then trained on this list of word lists, which creates a dense vector for each unique word in the corpus. The feature matrix was created by averaging the word embeddings for each transcript. For each transcript, the embeddings for each word in the transcript were added together, and then the mean of all these embeddings was taken. The features are converted into a numpy array, and the labels for each transcript are extracted. The resulting feature matrix has a shape of (number of transcripts, 2500).

b. **Audio Data**: We converted the .wav file to a numpy array, which returned the sample rate in Hz. Then, these numpy vectors were passed to calculate the short-time Fourier transform to get the magnitude of frequency content. We then obtained the mel frequency from the numpy array because these closely resemble the human auditory system, and considered mean for the same. To extract the tone of the audio file, we used the mean values obtained from chroma_stft. We used the mean of melspectogram to obtain a spectrum of audio signals which is similar to how humans perceive sound. The mean of contrast was used to determine the intensity difference between peaks and valleys. Piptrck was used to determine the mean of pitch values. We used the mean, median, max, min, standard deviation, and average values of rms to calculate the root mean square at different time frames. Zero crossing rate measures how often the amplitude of an audio signal crosses the zero point. We made an array with the mean, median, max, min, standard deviation, and average values of zcr. At last, we concatenate all the feature arrays (mfcc, chroma, mel, contrast, pitches, rms, zcr, spectrall_rolloff and spectral_flux) in a pickle file.

c. **Motion Capture Data:** For each utterance, we isolate the data based on start/end time stamps and use that to separate the data into 200 evenly split, distinct sets. Each set contains 165 data points for faces, 18 of hand movements, and 6 for head rotations. For spatial data, each x, y, and z value associated with a head location in a given moment are combined in a tuple. The same tuple grouping is also created for hand locations. Each set is then averaged along its columns (each frame) to get a 1x189 array. These 200 arrays are finally combined to achieve a 200x189 two dimensional array per utterance. This is consistent with prior research using the IEMOCAP dataset.

C. **Modeling**:

a. **Text and Audio Data:** We made a dataframe with the features obtained from word embedding and tfidf vectorization. Then, we split data and predicted the label value(emotion value) using models such as RandomForest, XGBoost, MultiLayer Perceptron, Support Vector Machines and Naive Bayes. The features were split in x_train, x_test, y_train and y_test, and the emotion label was predicted. We followed the same

methodology as in audio, using a variety of models including RandomForest, XGBoost, MultiLayer Perceptron, Support Vector Machines and Naive Bayes.

b. **Motion Capture Data:** The motion capture model is trained with all data (head, hand, and face) concatenated. Since each utterance is a 200x189 array, we opted for immediately using a convolutional neural network. We implemented five convolutions with between 32-256 filters each, some dropout, and a rectified linear unit (ReLU) activation function. After these five convolutions, a dense layer of 256 hidden units is implemented. We used Adam for our optimizer.

Results

- The individual text models had accuracies ranging from 62% to 68%. However, when these models were combined in an ensemble, the accuracy dropped to 61%. This decrease in accuracy could be attributed to factors such as lack of diversity in the individual models' predictions, which limited the overall improvement when combined. Furthermore, the ensembled model may have been affected by correlated errors or overfitting, leading to a negative impact on its performance.

Text		
Model	**Accuracy(%)**	**F1-Score**
Logistic Regression	67.91	0.66
MLP	65.97	0.64
Naïve Bayes	62.37	0.61
Random Forest	66.88	0.64
SVC	68.31	0.64

	precision	recall	f1-score	support
ang	0.6946	0.6042	0.6462	192
hap	0.6494	0.6677	0.6584	319
neu	0.5678	0.6011	0.5840	376
sad	0.5607	0.5455	0.5530	220
accuracy			0.6098	1107
macro avg	0.6181	0.6046	0.6104	1107
weighted avg	0.6119	0.6098	0.6101	1107

- The accuracy of the audio models ranged between 50-65%. We achieved slightly better accuracy when fusing the multiple audio models: 65%.

Audio		
Model	**Accuracy(%)**	**F1-Score**
Logistic Regression	64.84	0.64
MLP	65.24	0.65
Naïve Bayes	50.11	0.46
Random Forest	65.07	0.63
SVC	66.34	0.66

	precision	recall	f1-score	support
ang	0.7174	0.6346	0.6735	208
hap	0.6263	0.5868	0.6059	317
neu	0.6517	0.6287	0.6400	369
sad	0.6111	0.7746	0.6832	213
accuracy			0.6459	1107
macro avg	0.6516	0.6562	0.6506	1107
weighted avg	0.6489	0.6459	0.6448	1107

- To combine the textual and acoustic data modalities, we created a class which combined multiple speech and text models using voting to predict emotions. The results show that we achieved 76% accuracy when combining the probabilities of text and audio models (soft voting) and accuracy of 66% when selecting the majority probability (hard voting). This is roughly in-line with, but slightly worse than, prior multimodal research using the IEMOCAP dataset.

```
Accuracy:  0.7649584838223307
F1 Macro:  0.7716656797564473
Precision Macro:  0.7836522292620706
Recall Macro:  0.7639700634875147
F1 Weighted:  0.7663771159405623
Precision Weighted:  0.7719998754791252
Recall Weighted:  0.7649584838223307
```

Text Audio Combined Ensembled		
Soft	Accuracy	76.49%
	F1 Score	0.77
Hard	Accuracy	66.40%
	F1 Score	0.67

- The motion capture model was slightly less accurate. It achieved an emotional recognition accuracy of 44% with an F1 score of 0.41. This result is slightly worse than prior work, although it's difficult to make direct comparisons since most other papers use only 4-5 emotions instead of 7. The motion capture model was then combined with LSTM models using the text and speech data and achieved an accuracy of 61% with an F1 score of 0.55. It's likely that with some hyperparameter tuning, this result could be slightly improved, as this was one of our last results before finishing the project. We ran out of time to continue this work any further. Additional further work could be done in reducing the number of emotions that the model attempts to recognize. This would almost certainly increase accuracy and the F1 score, as well as allow for more direct comparisons to prior work on ensemble emotion recognition models using IEMOCAP.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 100, 95, 32)	320
dropout_5 (Dropout)	(None, 100, 95, 32)	0
activation_1 (Activation)	(None, 100, 95, 32)	0
conv2d_2 (Conv2D)	(None, 50, 48, 64)	18496
dropout_6 (Dropout)	(None, 50, 48, 64)	0
activation_2 (Activation)	(None, 50, 48, 64)	0
conv2d_3 (Conv2D)	(None, 25, 24, 64)	36928
dropout_7 (Dropout)	(None, 25, 24, 64)	0
activation_3 (Activation)	(None, 25, 24, 64)	0
conv2d_4 (Conv2D)	(None, 13, 12, 128)	73856
dropout_8 (Dropout)	(None, 13, 12, 128)	0
activation_4 (Activation)	(None, 13, 12, 128)	0
conv2d_5 (Conv2D)	(None, 7, 6, 128)	147584
dropout_9 (Dropout)	(None, 7, 6, 128)	0
activation_5 (Activation)	(None, 7, 6, 128)	0
flatten_2 (Flatten)	(None, 5376)	0
dense_3 (Dense)	(None, 256)	1376512

Training the CNN on motion capture data

Conclusions and Lessons Learned

In conclusion, we have explored the problem of speech emotion recognition using multimodal techniques, combining text, audio and motion capture data.. Our focus was on using the IEMOCAP dataset and preprocessing the data. We used various deep learning models, including CNNs and LSTMs, and achieved reasonable accuracy of around 76% in recognizing emotions from the combined modalities. Future work could include incorporating additional datasets, such as SEWA, to validate the model's performance. Overall, this project provided an exciting opportunity to develop models that better

understand human emotion and improve the communication experience, especially in situations where social cues may be limited, such as in digital communication.

By working on this emotion recognition project, we have learned about the importance of multimodal data and how it can improve NLP performance. We have also gained experience in dealing with messy real-world datasets and developing skills in data cleaning and feature engineering. Additionally, the project provides an opportunity to work with different machine learning models, including Random Forests, SVC, CNNs and LSTMs, and develop a solid understanding of machine learning fundamentals. Overall, this project was an excellent learning opportunity for us!

Team Member Contributions

- Shalin worked with the text data, trying a variety of models to see which would be the best fit.
- Astha worked on extracting features and building several models for the acoustic data.
- Alex worked on the motion capture data using IEMOCAP and a SEWA video data model (which was unable to be used in this final report). He also worked on an LSTM for speech and text data using Shalin's and Astha's text and audio features and combined it with the motion capture CNN using the keras model compilation toolkit.
- We worked together to combine our three models.
- We equally contributed to the presentation and the report.

References

[1] Carlos Busso, Murtaza Bulut, Chi Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. *Lang. Resour. Eval.* 42, 4 (2008), 335–359. DOI:https://doi.org/10.1007/s10579-008-9076-6

[2] A. Christy, S. Vaithyasubramanian, A. Jesudoss, and M. D.Anto Praveena. 2020. Multimodal speech emotion recognition and classification using convolutional neural network techniques. *Int. J. Speech Technol.* 23, 2 (2020), 381–388. DOI:https://doi.org/10.1007/s10772-020-09713-y

[3] Jean Kossaifi, Robert Walecki, Yannis Panagakis, Jie Shen, Maximilian Schmitt, Fabien Ringeval, Jing Han, Vedhas Pandit, Antoine Toisoul, Bjorn Schuller, Kam Star, Elnar Hajiyev, and Maja Pantic. 2021. SEWA DB: A Rich Database for Audio-Visual Emotion and Sentiment Research in the Wild. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 3 (2021), 1022–1040. DOI:https://doi.org/10.1109/TPAMI.2019.2944808

[4] Paul Pu Liang and Ruslan Salakhutdinov. 2018. Computational Modeling of Human Multimodal Language : The MOSEI Dataset and Interpretable Dynamic Fusion. *First Work. Gd. Chall. Comput. Model. Hum. Multimodal Lang.* (2018).

[5] Samarth Tripathi, Sarthak Tripathi, and Homayoon Beigi. 2018. Multi-Modal Emotion recognition on IEMOCAP Dataset using Deep Learning. (2018). Retrieved from http://arxiv.org/abs/1804.05788