Multi-Modal Approach for Sentiment Analysis

Saharsh Agarwal Siddhartha Namburu Ninaad Damis Nitika Suresh
Jaldhir Trivedi (TA) Prof. Amir B. Farimani

Abstract

Determining sentiments is one of the most important skills that separates humans from computers. In this paper, we have tried to enable computers to do an utterance level sentiment recognition in multi-party conversations using Shallow Machine Learning models. Generally, Neural Networks have proven to perform better at this task and hence, we have used state-of-the-art networks to benchmark our results in comparison. We perform sentiment analysis on both text and audio individually for the same utterance and finally, use them together to create a bi-modal approach. We have been able to achieve a significant increase in accuracy of sentiment recognition using this multi-modal approach, using very less time.

1. Introduction

With the exponential growth in social media and ease of voicing opinions in form of texts, audios and videos - industrial giants like Google, Amazon, Netflix and Facebook are leveraging this data for better customer acquisition, experience and service. Manually analysing this data is impossible because of the huge amount of it already existing and being generated each day.

Owing to the advent of Machine Learning and Neural Networks, the task of sentiment analysis has gathered significant attention from the industry as well as the research community. Identifying emotions is a fundamental step towards higher-intelligence tasks like - chat-bots [21], visual question-answering, suicide prevention [3], reducing reaction severity when angry, and many others. Conversations are further convoluted to understand as they as they are a function of more variables compared to individual sentences. Some of these variables include - the speaker, the topic, speaker's personality and viewpoint. [13].

However, most of the analysis earlier was done on textual data alone due to higher retention of information, computational limitations and availability from credible sources. With increased computational abilities, the comprehension using audio has shown similar results compared to text. Yet the availability of refined, labelled data and difference in attributes from person to person has hindered the growth in this specific domain.

Television series are one of the best sources for documented textual, auditory and visual data as they emulate emotions that produce the desired reactions from the audience. We have used the famous F.R.I.E.N.D.S. television show to procure our data. The data has 13,708 utterances from over 1430 scenes. Each of these utterances have both audio and text available for them along with the unique sentiment and emotion labelled.

In this paper, we have successfully shown that using a multi-modal approach, i.e., using both text and audio together, despite supervised shallow machine learning, a significant increase in performance of emotion recognition can be achieved.

2. Related Work

In the literature, emotion detection has mainly focused on non-conversation data, such as sentence-level text [7] and document-level text [14], and on multimodal emotion recognition using audio, visual, and text modalities [17] [16]. More recently, emotion detection in conversations has attracted increasing attention in NLP due to its applications in many emerging tasks such as social media analysis and the emergence of publicly available conversational datasets such as EmotionLines [1] and MELD [12].

Emotion Recognition in Conversation (ERC) ideally requires context modeling of the individual utterances. This context can be attributed to the preceding utterances, and relies on the temporal sequence of utterances. [6] proposes a conversational memory network (CMN) which uses a multimodal approach for emotion recognition by leveraging contextual information from the conversation history. [11] utilizes a LSTM based model that enables utterances to capture contextual information from their surroundings in the same video.The above is mostly limited to dyadic conversations, and thus not scalable to ERC with multiple interlocutors.

Emotion detection in multi-speaker conversations need to properly model the interactive influence of multiple speakers for a better performance. [8] uses hierarchical multi-stage RNN with attention mechanism to model inter-party dependencies. Graph neural networks have become very popular, and have been applied to ERC tasks in [5] and

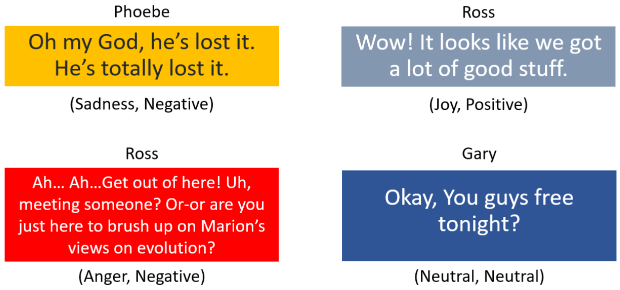

Phoebe
Oh my God, he's lost it. He's totally lost it.
(Sadness, Negative)
Ross
Wow! It looks like we got a lot of good stuff.
(Joy, Positive)
Ross
Ah... Ah...Get out of here! Uh, meeting someone? Or-or are you just here to brush up on Marion's views on evolution?
(Anger, Negative)
Gary
Okay, You guys free tonight?
(Neutral, Neutral)

Figure 1. Example of 4 utterances from the data set, each in a coloured rectangle. The speaker is at the top, followed by emotion and sentiment in brackets at the bottom, respectively.

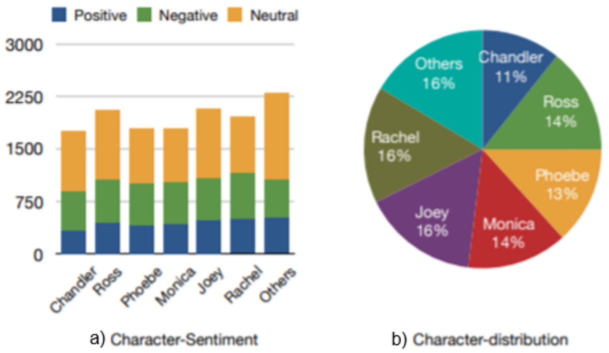


Figure 2. Character Distribution across MELD

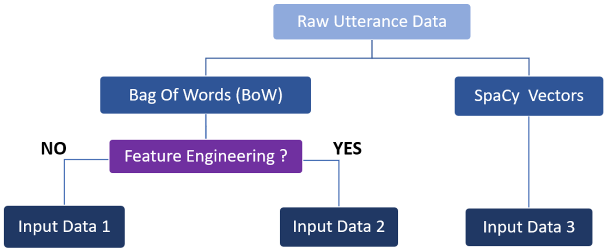


Figure 3. Two methods for feature extraction - BoW and Spacy Vectors - to generate three datasets to test for text.

[19]. Transformer networks along with an external knowledge base is used in [20] to perform emotion recognition. New frameworks such as COSMIC [4] have been proposed, which incorporates elements of commonsense such as mental states, events and causal relations, to learn interactions between interlocutors participating in a conversation.

Some of the above studied models and approaches become the benchmark for our Shallow Machine Learning approaches. Though we had expected lesser accuracy, astonishingly, we got very similar results leveraging substantially less time and complexity

3. Data

We use the Multimodal Emotion Lines Dataset (MELD), which is a large scale multi-party emotional conversational database. MELD contains about 13,708 utterances from 1433 dialogues from the TV series FRIENDS. There are 7 unique emotions for each utterance, namely - Anger, Fear, Disgust, Joy, Neutral, Sadness and Surprise. There are also 3 unique sentiments - Positive, Neutral and Negative.

Each utterance is annotated with an emotion and a sentiment label. Figure 1 illustrates 4 sample utterances from the data set selected. These dialogues are present in both audio and text forms, with a unique number to identify both as they are stored in different csv files.

Figure 2(a) illustrates the distribution across sentiment labels for each of the parties. It can be seen that the distribution of sentiment in the dataset is almost uniform, with the majority emotion being neutral. Figure 2(b) shows the overall coverage of the speakers across the dataset, which is also almost fairly balanced.

4. Methodology

4.1. Text Analysis

For analytical purposes, we first explored text individually. The flow of work is represented in fig. 3. Using the textual data, two types of feature extraction methods are deployed-

- Bag-of-Words (BoW)
- Spacy Vectors

4.1.1 Bag of Words (BoW)

Bag of Words is a simple and flexible way of for extracting features from text. It is a representation of text with numbers. It involves two steps: **i)** creating a dictionary of all the unique words in the training data, followed by **ii)** defining each sample utterance as a frequency of these known words. For example, if we have N samples of text utterances and combining them together we have M unique/non-repeated words - thus, the training data after applying Bag of Words will become a double dimensional matrix of size N rows and M columns; where the j^{th} column is the frequency of the j^{th} unique word in the BoW dictionary present in the i^{th} sample. This is represented as **Input Data 1** in the fig. 3. The model is primarily concerned with known words and their presence, not with their placing or structure.

There are some limitations to the Bag of Words approach and can eliminated to a great extend by feature engineering we performed -

- **Punctuations and Stopwords** - Textual data is filled with words like articles - $'a',' an,' the'$, connectors - $'because',' so'$, punctuations - $'.';' ,'$, and many more.

These words rarely convey any meaning and yet increase the unique word count (M), thereby increasing computational time and complexity. All such words are identified and removed using '$nltk.corpus$'.

- **Linguistic Morphology** - Stemming to reduce the inflected or derived word to their base word. Example - giving, given, gave and give - all stem from the same root word 'give' and generally denote the same emotion as well. Hence, it made sense to treat these words the same and not unique/different. This further reduced the input feature dimension without losing relevant information.

After feature engineering, Bag of Words is applied on the new and cleaner data. The double dimensional array is then fed as N training examples to the 4 Shallow Machine Learning models we chose. This is data represented as **Input Data 2** in the fig. 3

4.1.2 Spacy

Spacy vectors are obtained from a pre-trained model, trained on multitude of text corpuses including -

- **OntoNotes 5** [15] - A large corpus comprising various genres of text - news, conversational telephone speech, weblogs, usenet newsgroups, broadcast, talk shows, in three languages (English, Chinese, and Arabic). It also has the structural information (syntax and predicate argument structure) and shallow semantics (word sense linked to an ontology and co-reference).

- **ClearNLP** [2] - Constituent-to-Dependency Conversion (Emory University).

- **WordNet 3.0** [9] - Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. The structure of how the words are arranged into synsets can also be visualized (Princeton University) (http://wordnetweb.princeton.edu/perl/webwn).

- **GloVe Common Crawl** [10] - Web data from Common Crawl, trained on 840 billion tokens, with around 2.2 million unique tokens is used for training, from GloVe's repository.

Using the datasets above, an NLP pipeline is created, as shown in fig. 4, using SpaCy package to create 300 dimensional vectors for each word. That is for every word in the utterance KNN is used to find nearest word in 685k corpus of words and a float vector corresponding to that word in that corpus is assigned. Further, the whole utterance is treated as the average of the words it constitutes, making the input feature size per sample as 300. This data is stored as **Input Data 3**, represented in fig. 3.

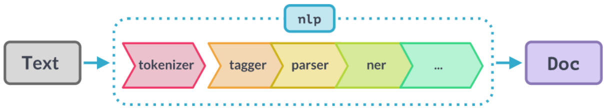
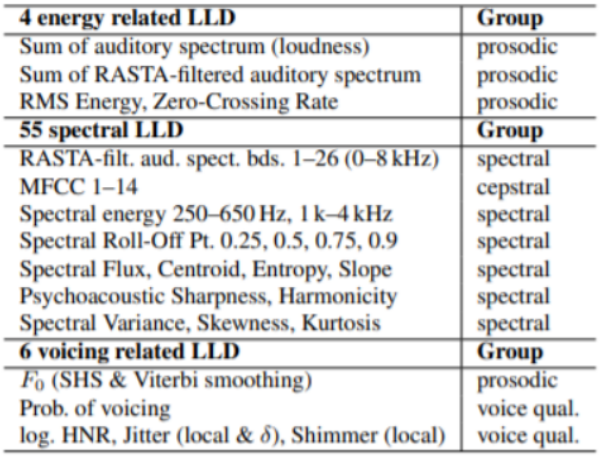


Figure 4. NLP pipeline for spacy vectors.



4 energy related LLD	**Group**
Sum of auditory spectrum (loudness)	prosodic
Sum of RASTA-filtered auditory spectrum	prosodic
RMS Energy, Zero-Crossing Rate	prosodic
55 spectral LLD	**Group**
RASTA-filt. aud. spect. bds. 1–26 (0–8 kHz)	spectral
MFCC 1–14	cepstral
Spectral energy 250–650 Hz, 1 k–4 kHz	spectral
Spectral Roll-Off Pt. 0.25, 0.5, 0.75, 0.9	spectral
Spectral Flux, Centroid, Entropy, Slope	spectral
Psychoacoustic Sharpness, Harmonicity	spectral
Spectral Variance, Skewness, Kurtosis	spectral
6 voicing related LLD	**Group**
F_0 (SHS & Viterbi smoothing)	prosodic
Prob. of voicing	voice qual.
log. HNR, Jitter (local & δ), Shimmer (local)	voice qual.

Figure 5. ComParE accoustic feature set : 65 provided low level descriptors (LLD's)

Thus, using text we generated three datasets to compare results, and ultimately chose the best feature set for the multi-modal approach - SpaCy Vectors.

4.2. Audio Analysis

To extract audio features, we use $openSMILE$ - an open-source toolkit for audio feature extraction and classification of speech and music signals. Using the $ComParE$ feature set from the INTERSPEECH 2013 Computational Paralinguistics Challenge, we obtain 6373 features. These features result from the computation of various functionals like mean, max, skewness, and kurtosis, over low-level descriptors (LLD), shown in fig. 5. The low-level descriptors cover a broad set of descriptors from the fields of speech processing, music information retrieval, and general sound analysis - including pitch, loudness, jitters, and many others. The data of each sample utterance audio gives 6373 features which are collectively stored as **Input Data 4**, as shown in fig. 6.

As this audio representation is high dimensional, we employ L2 regularization based feature selection using Support Vector Machines (SVM), to get a dense representation of the overall audio segment. This reduces our feature set from 6373 to 1422 features, without losing much of the relevant information. The new reduced data is stored as **Input Data 5**, shown in fig. 6.

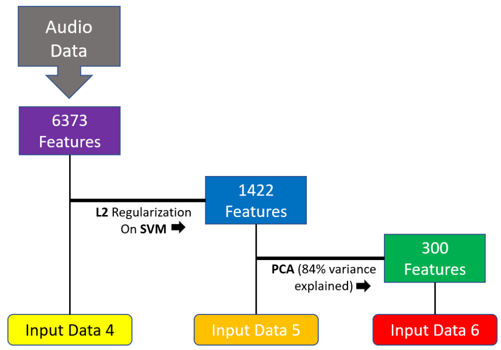


Figure 6. Feature Extraction and Reduction for Audio Input

Principal Component Analysis is further performed on the new data stored to further reduce the audio features to 300. Despite Reducing the data by 78.9%, we are still able to retain over 84% of the information in the data (,i.e., the explained variance). These 300 features for all the audio samples is saved as **Input Data 6**, shown in fig. 6.

4.3. Bi-Modal Approach

For the Bi-modal analysis, we concatenate both audio and textual features - each of dimension 300 representing their respective modes. Thus, by concatenating both the above 300 features, we obtain our bi-modal feature set of dimension 600. We have taken both audio and text features in 1:1 ratio. However, we tried for other ratios but this gave the best result in consistency with the literature.

5. Experiments

Our workflow broadly consists of feature extraction and feature engineering of text and audio data. This is further tested with shallow Machine Learning models such as Multinomial Logistic Regression, Naive Bayes, Random Forest, Support Vector Machines and a simple multi-layer perceptron (MLP), followed by comparison with the benchmark models tCNN, bcLSTM, dRNN. The results obtained have been discussed in the following sections.

5.1. Machine Learning models

Stratified K-fold cross-validation is performed on all the 4 models to ensure testing is unbiased and hence, improves the confidence on the model. The mean training accuracy, testing accuracy, their standard deviations, F1 score, and time taken was computed for both emotions and sentiments. Folds are set to 5, splitting data in 80% for training and 20% for validation for each of the 5 runs.

Table 1. MLP architecture.

Model Summary			
Layer	Input shape	Output shape	No. of parameters
Linear - 1	300	3000	903,000
Linear - 2	3000	2000	6,002,000
Linear - 3	2000	1000	2,001,000
Linear - 4	1000	500	500,500
Linear - 5	500	300	150,300
Linear - 6	300	3	903
Total trainable parameters = 9,557,703			

5.1.1 Multinomial Logistic Regression

Multinomial Logistic Regression is used on both emotion and sentiment analysis as there are multiple (more than 2) class labels for both. We use the default solver lbfgs with L2 regularization as it is observed to converge well with higher accuracy for the data.

5.1.2 Naive Bayes

Naive Bayes is another algorithm we used on the data assuming that the features are independent and contribute equally to the outcome. Scaling of data between 0 and 1 was performed to tackle negative values successfully.

5.1.3 Random Forest

Random Forest classifier is used on the data with the default number of estimators = 100. Changing the depth of the trees or the estimators did not change the accuracy significantly.

5.1.4 Support Vector Machines

SVM algorithm classifies the data by maximizing the separation between each of the classes. For our case, we used the one-vs-all approach for classification where we construct n (n = number of features) SVM's and for each SVM we separate the data into two groups and one group corresponds to one label and the other group contains the rest of the data points. The hyperparameters used are: Kernel = rbf, Kernel coefficient = 1 / number of features, Decision function shape = one-vs-all.

5.1.5 MLP (Linear Neural Network)

A multi-layer perceptron was designed with the following architecture: 6 fully connected layers with drop out in the hidden layers, ReLu activation in all the layers, SGD optimizer with learning rate = 0.005 and momentum = 0.9. The cross-entropy loss is computed. Table 1 shows the MLP architecture used.

Table 2. LSTM architecture.

Model Summary		
Layer	Output shape	No. of parameters
Bidirectional LSTM	(10966,1422)	4061232
Dropout	(10966,1422)	0
Dense	(10966,88)	125224
Dense	(10966,64)	5696
Dropout	(10966,64)	0
Dense	(10966,48)	3120
Dropout	(10966,48)	0
Dense	(10966,24)	1176
Dense	(10966,3)	75
Total trainable parameters = 9,557,703		

5.2. LSTM model

We used a bidirectional LSTM model in addition to train the dataset. The features of the model are as follows: Bidirectional LSTM followed by 5 dense layers, with dropout in each hidden layer with ReLu activation. Table 2 shows the LSTM architecture.

5.3. Baseline models

The MELD paper [12] uses three baseline models namely textCNN, bcLSTM, DialogueRNN. We use them as the baseline benchmarks for our models. textCNN applies CNN only on the text data without considering the context of the conversation. It is the most simplest baseline model. bcLSTM is a more advanced baseline model which makes use of bidirectional Recurrent Neural Network. It takes the unimodal text and extracts features taking the glove embeddings as input to a CNN and for unimodal audio it takes the audio representations as input to an LSTM model. Both these representations are input to the bimodal approach for classification. DialogueRNN is a stronger baseline model as it takes into account the individual speaker states throughout the conversastion. It uses 2 GRU's (party GRU and global GRU) to update both speaker and context states. Finally, it uses an emotion GRU to model the emotional information for classification.

6. Results

Input data sets generated in Section 4 are passed through the Machine Learning models described in section 5, using stratified k-fold. Additionally an MLP is used to estimate how shallow machine learning models compare(except in Bag of Words approach since the input size is not fixed). The Average accuracies and time taken are compared for these Input data sets to infer which approach works best to predict the emotion and sentiment.

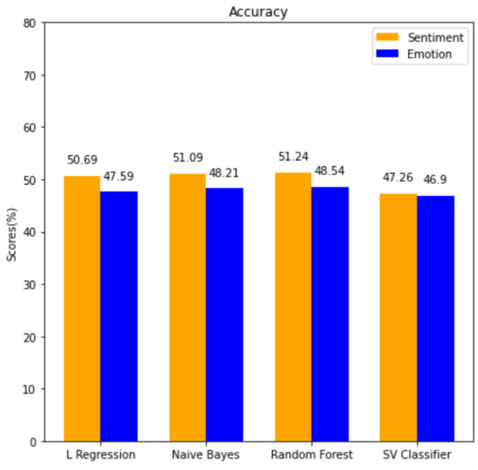


Figure 7. Bag of Words approach

6.1. Bag of Words - BoW (Input Data - 1)

Fig. 7 shows how Bag of Words approach works without feature engineering. The accuracies are comparatively less since this approach works for articles with less variation in the words used. Also, analyzing the vectors created using BoW compels for the use of feature engineering to remove unnecessary stop words, punctuation etc.

6.2. BoW + Feature Engineering (Input Data - 2)

Three feature engineering methods are used to transform input text dialogues to meaningful vectors generated from Bag of Words approach. fig. 8. shows how the shallow machine learning models compare. The accuracies are similar to raw Bag of Words approach, shows that feature engineering especially Linguistic Morphology made the predictions worse since the tense of verbs may be important in predicting the sentiment. Removing the stemming operation restores the accuracy similar to Bag of Words approach. Also, the time reduces for models like Random Forest when feature engineering reduces the size of the input data.

6.3. Spacy Vectors (Input Data - 3)

SpaCy package (en-vectors-web-lg) is used to generate 300 (hyperparameter) dimensional vectors. Since the spacy package uses its own feature engineering methods, the utterances were directly used to generate these vectors. Since the input size is always fixed and low as opposed to Bag of Words approach, an MLP (as described in Section 5.1.5) is used to compare performance of neural networks to the shallow machine learning models. Support Vector Classifier gave the best test accuracy among the five models. The accuracies are shown in fig. 9. The training time accuracies of the shallow machine learning models are less than 65 percent, hence there is no overfitting. The training accuracy

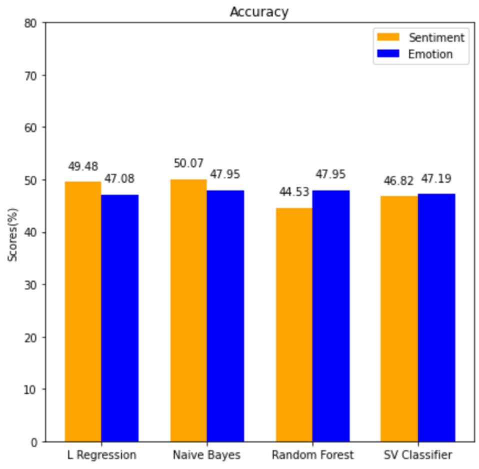


Figure 8. Bag of Words approach, with feature engineering

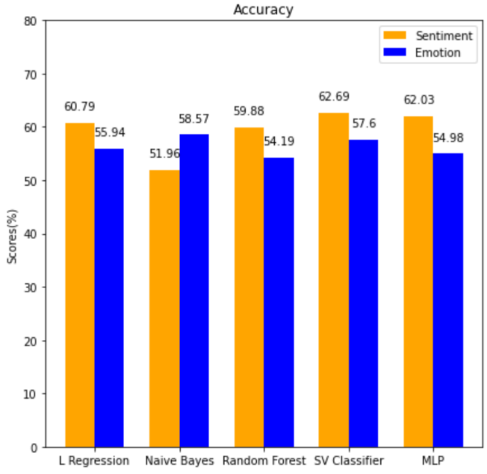


Figure 9. Spacy Vectors

of MLP corresponding to best test accuracy is 74 percent hence there is no overfitting in this case either.

6.4. Audio Analysis (Input Data - 4, 5 & 6)

Audio features are extracted using openSMILE as discussed in section 4.2. Further, three sets of input data are created by reducing the 6373 dimensional feature vectors to find which feature set is performing the best. Random forest regressor is used to compare the 6373, 1422, 300 dimensional feature vectors and the results are shown in fig. 11. The accuracy is similar for these feature sets because 84 percent of variance in the data is explained by the 300 dimensional feature set. Further the accuracies of shallow machine learning models are compared to assess their performance as shown in fig. 11. Hence it can be understood that low level features are not sufficient to capture complexity of the data validating importance of deep-learning based features to predict emotions. To estimate how neural networks are performing a Bi-Directional LSTM followed by an MLP is used to test if there is increase in performance. The LSTM described in Section 5 gives an accuracy of 47.23 percent to predict the sentiment. Hence, it can be infered that an SV Classifier is sufficient to get maximum possible accuracy from the obtained features.

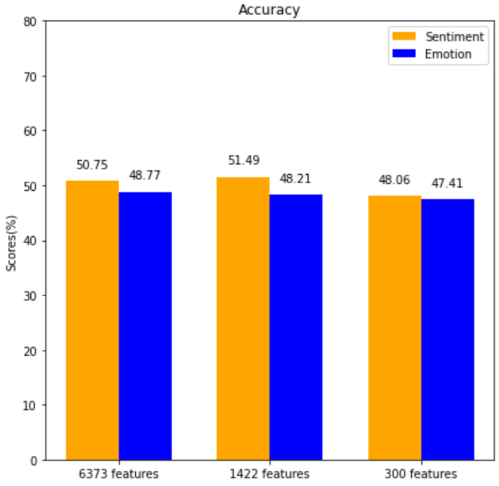


Figure 10. Audio vector size vs random forest accuracy

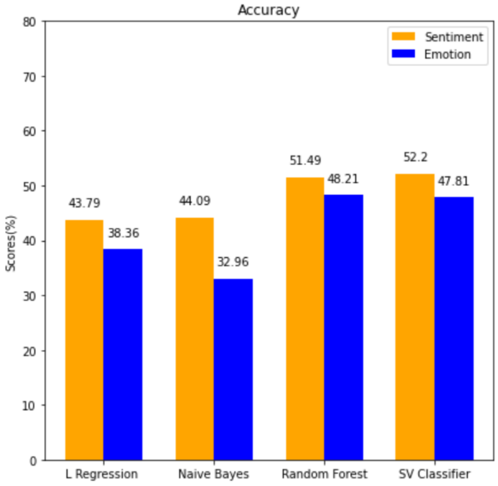


Figure 11. 1422 dimension audio features

6.5. Multi-Modal Analysis

The audio features are concatenated with spacy vectors of text to test how multi-modal approach evaluates against uni-model text and audio approaches. The Random Forest, SV Classifier and MLP are used to evaluate the multi-modal approach since the Logistic Regression and Naive Bayes performed bad in terms of accuracy and time taken.

Different combinations of audio and text vectors are

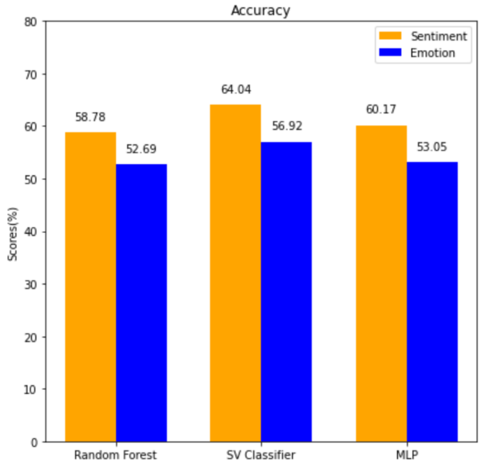


Figure 12. Accuracies in the Multi-Modal approach

Table 3. Comparison with baselines.

Models	Emotion F1 score	Sentiment F1 score
t-CNN	55.02	64.25
bcLSTM	59.25	66.68
D-RNN	60.25	67.56
SV Classifier	**56.92**	**64.02**

tested without loosing much variance in data to observe that models perform better when the text and audio is combined in 1:1 ratio. Hence, the 300 dimensional spacy vectors are combined with reduced 300 dimensional audio vectors for the prediction, and the results are shown in fig. 13.

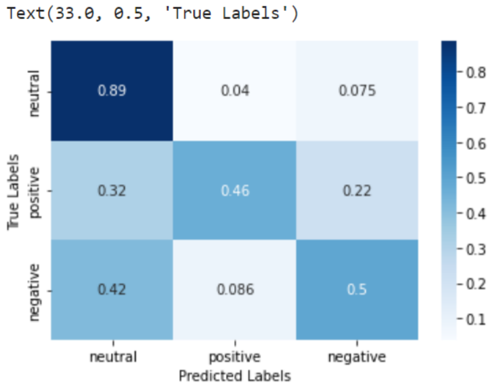
The multi-modal approach increases average accuracy by at-least 2 percent in predicting the sentiment and by at-least 1 percent in predicting the emotions. Additionally, these values are compared with text-CNN, bc-LSTM and dialogue RNN. SV Classifier gives astonishingly great results when compared to the deep-learning networks, given the training time and prediction time.

6.6. Comparison with baselines

As discussed in 5.3 the three baseline models are benchmarked against the shallow ML models used and dialogueRNN performs best while the SV classifier used in our approach performs very similar to textCNN with very less training time. Table 3 shows the comparison of the F1 scores.

6.7. Analysis

The Confusion matrix for predicting emotions and sentiments using bi-modal approach using the SV Classifier is shown in fig. 13. Since the data consists of more neutral utterances the confusion matrix depicts higher accuracy in neutral class for sentiments. Fig. 14. shows accuracy of different emotions and sentiments. Disgust and Anger are the emotion classes with least accuracy because of two probable reasons. The utterances corresponding to these emotions have varying pitch making it very difficult to predict emotion. Further, since sarcasm is often associated with these utterances, the class predicted for these emotions is often neutral.



```
Text(33.0, 0.5, 'True Labels')
```

Figure 13. Sentiment prediction confusion matrix

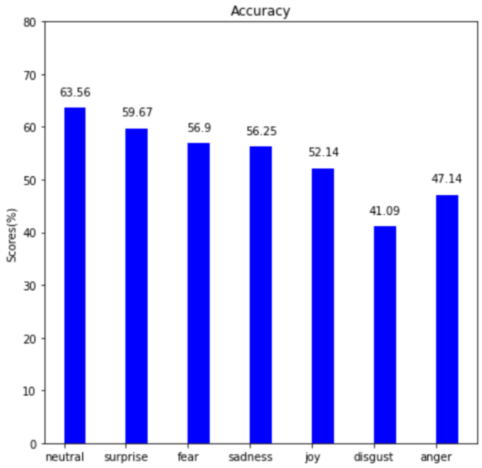


Figure 14. Accuracies corresponding to different emotions

The inference time for the test utterances are shown in fig. 15 SV classifier takes highest time to predict emotion per sample, followed by MLP, Random forest, Naive Bayes and Logistic regression. This follows the same trend in accuracies. Random forest is observed to have the best prediction time vs accuracy ratio. Appropriate model can be chosen based on corresponding NLP application. For example, to find emotion of the customer base of a product, since the inference doesn't happen live, accuracy precedes over time taken. Comparing the training time, the neural network has the highest training time of 1200 seconds compared to less than one sec for shallow machine learning models. This

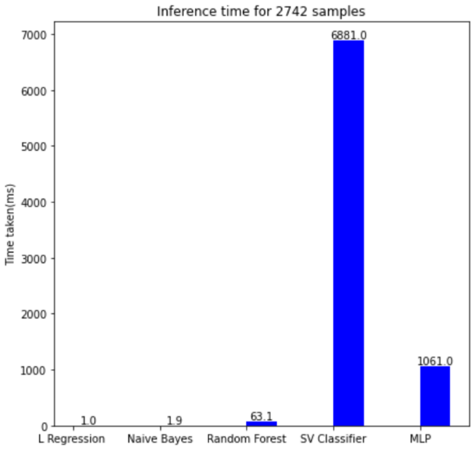


Figure 15. Inference time for different models corresponding text analysis using spacy vectors

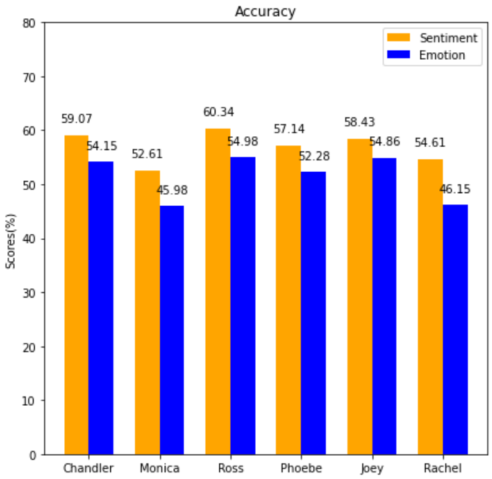


Figure 16. Accuracies corresponding to different FRIENDS characters

is a really important factor in applications of active/online learning.

The general sub-par accuracies of the above approaches can be attributed to two visible issues -

- **Sarcasm** - The multi-party conversations in the TV show FRIENDS are often sarcastic, where the underlying emotion is often difficult to predict.
- **Context** - Individual utterances often lack context due to which even positive and negative sentiments can labelled as neutral. Hence it is useful to consider all the text in a whole conversation to predict the emotion of the utterances. Hence, the deep-learning models utilizing the previous utterances leveraging consequential learning perform better.
- **Emotion shifts** - The ability to anticipate the emotion shifts within speakers throughout the course of a dialogue has synergy with better emotion classification. In our results, SV Classifier achieves a recall of 64 percent for detecting emotion shifts. However, in the ideal scenario, we would want to detect shift along with the correct emotion class.

Further, different character's utterances are used in the test data to analyze individual accuracies. Monica and Rachel have less accuracy compared to other parties as shown in fig. 16. This is because of their varying pitch in the corresponding audio making it difficult to predict. With regard to text, it is visible that some of their dialogues are incomplete and a facial expression in the original video clip is used to convey the message. Thus, a third mode using action & expression recognition based computer vision models can be beneficial in increasing the accuracy of predicting the emotion.

7. Conclusions

Natural Language Processing and Speech Recognition is one of the most researched fields in the current times. Arguably the biggest difference between humans and computers has been the ability of humans to understand and respond to emotional stimuli. As humans, we understand emotions through various channels including audio, text, visual and touch. This is what we have tried to establish in our report. Multi-modal approach for understanding emotions will be highly rewarding for computers to successfully emulate what humans have been doing for several millenniums. Adding these modes can improve the results significantly even without the use of neural networks.

8. Future Directions

Future research using this dataset should focus on improving contextual modeling. Helping models reason about their decisions, exploring emotional influences, and identifying emotion shifts are promising aspects. Another direction is to use visual information available in the raw videos. Identifying face of the speaker in a video where multiple other people are present is challenging. The advancements in computer vision can now get a label using a video using action recognition/detection networks. In our results, audio features do not help significantly. So better feature extraction modules can be adapted for these auxiliary modalities in order to improve the performance further. Also, the bimodal approach discussed in this paper just concatenates the audio and text features. A heuristic based approach can also be explored with weights learned by a neural network. Additionally, fusion methods like MARN [18] can also be used for further improvement.

References

[1] S.-Y. Chen, C.-C. Hsu, C.-C. Kuo, L.-W. Ku, et al. Emotionlines: An emotion corpus of multi-party conversations. *arXiv preprint arXiv:1802.08379*, 2018.

[2] J. D. Choi and A. McCallum. Transition-based dependency parsing with selectional branching. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1052–1062, 2013.

[3] B. Desmet and V. Hoste. Emotion detection in suicide notes. *Expert Systems with Applications*, 40(16):6351–6358, 2013.

[4] D. Ghosal, N. Majumder, A. Gelbukh, R. Mihalcea, and S. Poria. Cosmic: Commonsense knowledge for emotion identification in conversations. *arXiv preprint arXiv:2010.02795*, 2020.

[5] D. Ghosal, N. Majumder, S. Poria, N. Chhaya, and A. Gelbukh. Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation. *arXiv preprint arXiv:1908.11540*, 2019.

[6] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann. Conversational memory network for emotion recognition in dyadic dialogue videos. In *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2018, page 2122. NIH Public Access, 2018.

[7] S. Li, L. Huang, R. Wang, and G. Zhou. Sentence-level emotion classification with label and context dependence. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1045–1053, 2015.

[8] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. Gelbukh, and E. Cambria. Dialoguernn: An attentive rnn for emotion detection in conversations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6818–6825, 2019.

[9] G. A. Miller. *WordNet: An electronic lexical database*. MIT press, 1998.

[10] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[11] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, and L.-P. Morency. Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 873–883, 2017.

[12] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*, 2018.

[13] S. Poria, N. Majumder, R. Mihalcea, and E. Hovy. Emotion recognition in conversation: Research challenges, datasets, and recent advances. *IEEE Access*, 7:100943–100953, 2019.

[14] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang. Dimensional sentiment analysis using a regional cnn-lstm model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 225–230, 2016.

[15] R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini, et al. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23, 2013.

[16] M. Wöllmer, F. Weninger, T. Knaup, B. Schuller, C. Sun, K. Sagae, and L.-P. Morency. Youtube movie reviews: Sentiment analysis in an audio-visual context. *IEEE Intelligent Systems*, 28(3):46–53, 2013.

[17] A. Zadeh, Y. Chong Lim, T. Baltrusaitis, and L.-P. Morency. Convolutional experts constrained local model for 3d facial landmark detection. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2519–2528, 2017.

[18] A. B. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L.-P. Morency. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, 2018.

[19] D. Zhang, L. Wu, C. Sun, S. Li, Q. Zhu, and G. Zhou. Modeling both context-and speaker-sensitive dependence for emotion detection in multi-speaker conversations. In *IJCAI*, pages 5415–5421, 2019.

[20] P. Zhong, D. Wang, and C. Miao. Knowledge-enriched transformer for emotion detection in textual conversations. *arXiv preprint arXiv:1909.10681*, 2019.

[21] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.