Emotion forecasting in dyadic conversation: characterizing and predicting future emotion with audio-visual information using deep learning

Sadat Shahriar
University at Albany, State University of New York, shahriarsadat71@gmail.com

The University at Albany community has made this article openly available. Please share how this access benefits you.

Follow this and additional works at: https://scholarsarchive.library.albany.edu/legacy-etd

Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, and the Psychology Commons

Recommended Citation
https://scholarsarchive.library.albany.edu/legacy-etd/2379

This Master's Thesis is brought to you for free and open access by the The Graduate School at Scholars Archive. It has been accepted for inclusion in Legacy Theses & Dissertations (2009 - 2024) by an authorized administrator of Scholars Archive. Please see Terms of Use. For more information, please contact scholarsarchive@albany.edu.
Emotion Forecasting in Dyadic Conversation: Characterizing and Predicting Future Emotion with Audio-Visual Information Using Deep Learning

by

Sadat Shahriar

A Thesis
Submitted to the University at Albany, State University of New York
In Partial Fulfillment of
the Requirements for the Degree of
Masters of Science

University at Albany
College of Engineering and Applied Sciences
Department of Electrical and Computer Engineering

2019
Abstract

Emotion forecasting is the task of predicting the future emotion of a speaker, i.e., the emotion label of the future speaking turn–based on the speaker’s past and current audio-visual cues. Emotion forecasting systems require new problem formulations that differ from traditional emotion recognition systems. In this thesis, we first explore two types of forecasting windows (i.e., analysis windows for which the speaker’s emotion is being forecasted): utterance forecasting and time forecasting. Utterance forecasting is based on speaking turns and forecasts what the speaker’s emotion will be after one, two, or three speaking turns. Time forecasting forecasts what the speaker’s emotion will be after a certain range of time, such as 3–8, 8–13, and 13–18 seconds. We then investigate the benefit of using the past audio-visual cues in addition to the current utterance. We design emotion forecasting models using deep learning. We compare the performances of FC-DNN, D-LSTM, and D-BLSTM which allows us to examine the benefit of modeling dynamic patterns in emotion forecasting tasks. Our experimental results on the IEMOCAP benchmark dataset demonstrate that D-BLSTM and D-LSTM outperform FC-DNN by up to 2.42% in unweighted recall. When using both the current and past utterances, deep dynamic models show an improvement of up to 2.39% compared to their performance when using only the current utterance. We further analyze the benefit of using current and past utterance information compared to using the current and randomly chosen utterance information, and we find the performance improvement rises to 7.53%. The novelty in this study comes from its formulation of emotion forecasting problems and the understanding of how current and past audio-visual cues reveal future emotional information.
Acknowledgements

First and foremost, I would also like to express my gratitude to Dr. Yelin Kim, lead scientist in Amazon Lab126, for her continuous mentorship throughout the year 2018, while she was an assistant professor at UAlbany. She taught me the true science of research, also pushing me to keep the high standards of my work with insightful and constructive comments. I would like to sincerely thank my advisor, Professor Gary J Saulnier, for his continuous support and guidance throughout many difficult phases of my journey for the Master’s degree. This thesis would not have been possible without his support and I can never thank him enough for that.

I would also like to express my gratitude to my thesis committee members for their valuable time and comment on my thesis: Professor Daphney-Stavroula Zois and Professor Mohammed Agamy. My special thanks to Prof. Siwei Lyu, Prof. Feng Chen (UT, Dallas), and Prof. Kevin Knuth for their enormous inspiration which motivated me for conducting my research and reshaped my thought process.

My two years in Albany have been truly a joyful experience, and the beginning of priceless friendships with many inspiring people. I cannot thank enough my friends at UAlbany: Priti G. Pachpande, Mohammad Khaledur Rahman, Tasmia Haque, and Shihab Ahmed. I would like to thank the faculty and stuffs of department of Electrical and Computer engineering for the constant support and guidance. I owe my deepest gratitude to my wife Jannatul Ferdoush for encouraging me in every phase of the journey while tolerating separation from me for more than a year. Above all, this thesis is dedicated to my wonderful parents, Shafiul Islam and Sayeda Begum, for being my life-long mentors. I also thank my sister Shifat Sharmin Paula for her endless love and support. Lastly, I greatly thank University at Albany and Google for the financial support which made pursuing of my MS degree possible.
Contents

Abstract ii

Acknowledgements iii

Contents iv

List of Figures vi

List of Tables vii

1 Introduction 1

1.1 Problem Formulation ........................................ 3
1.2 Approach ...................................................... 5
1.3 Related Works ................................................ 7

2 Dataset Description 9

3 Feature Description 11

3.1 Speech Features ............................................... 11
  3.1.1 Pitch ..................................................... 11
  3.1.2 Intensity .................................................. 14
  3.1.3 Mel-Frequency Cepstrum ............................... 15
3.2 Facial Features ............................................... 18

4 Deep-Learning Architecture 20
4.1 Fully-Connected Neural Network ........................................... 21
4.2 Recurrent Neural Network (RNN) ........................................... 26
   4.2.1 Long Short-Term Memory Network (LSTM) ......................... 26
   4.2.2 Bidirectional Long Short-Term Memory Network (BLSTM) ....... 29

5 Methodology ................................................................. 30
  5.1 Feature Extraction and Data Processing ................................. 30
  5.2 Emotion Forecasting Methods ............................................. 31
     5.2.1 Forecasting Windows .................................................. 31
     Utterance Forecasting (UF) ............................................... 31
     Time Forecasting (TF) ..................................................... 33
  5.2.2 Absence or Presence of Previous History ......................... 34
     History-Less Emotion Forecasting ...................................... 34
     History-Added Emotion Forecasting ................................... 34
  5.3 Experimental Setup ....................................................... 35

6 Results and Discussion .................................................... 36
  6.1 History-Less Emotion Forecasting ...................................... 36
  6.2 History-Added Emotion Forecasting .................................... 37
  6.3 Further Analysis ........................................................ 39

7 Conclusion ................................................................. 42

References ................................................................. 43
# List of Figures

1.1 Overall process of emotion forecasting ................................................. 6

2.1 A recording session of IEMOCAP database ........................................... 10

3.1 Humming Window .................................................................................. 16
3.2 The facial marker regions of IEMOCAP database ..................................... 19

4.1 A simple neural network ....................................................................... 21
4.2 NN Activation functions ........................................................................ 22
4.3 A many-to-one RNN architecture .......................................................... 26
4.4 The basic cell structure of LSTM ............................................................. 27
4.5 Architecture of BLSTM ........................................................................ 28

5.1 An example of emotion forecasting without history ................................. 32
5.2 An example of emotion forecasting with added history from the previous utterance ................................................................. 35

6.1 Comparison of performance using different forecasting windows, and deep-learning framework ............................................................. 39
**List of Tables**

1.1 Experimental approaches for evaluation of $H_1$ and $H_2$. The UF-cur and TF-cur approaches are used to test $H_1$, and the remaining approaches are used to test $H_2$. .......................................................... 7

1.2 Proposed and baseline experimental models for testing $H_1$ and $H_2$. ........ 7

6.1 UAR(%) comparison between static (FC-DNN) and dynamic (D-LSTM and D-BLSTM) models for both UF-cur and TF-cur approaches. The sign “[*]” denotes that the model is statistically significant ($p<0.05$) compared to the FC-DNN .......................................................... 37

6.2 Results of emotion forecasting with added history information: performance of UF-his and TF-his with dynamic modeling. The sign “[*]” denotes that the model with history-added technique is statistically significant ($p<0.05$) compared to the history-less technique of the model. ........ 38

6.3 UAR (%) comparison of UF-his/TF-his with UF-random/TF-random. The sign “[*]” denotes the statistically significant ($p<0.05$) result of history added technique, compared to the randomly added data. ......................... 40

6.4 The ratio of identical labels of forecasting and recognition, to all the forecasting labels for different forecasting windows (i.e., same-labels). ........ 40

6.5 An example for the per-class recall score (%), when the forecasting label is same with current emotional label. The example is taken from UF-cur 3 D-BLSTM result. ‘GT’ refers to forecasting ground truth and ‘Predict’ refers to forecasting labels predicted by the network. ......................... 41
Chapter 1

Introduction

Emotion Forecasting is the task of predicting the future emotion by comprehending the current behavioral cues. The task of the traditional emotion recognition problem is to identify the current emotional state of a speaker. On the contrary, emotion forecasting predicts the future emotional state by leveraging the audio-visual behavioral analysis from the present emotional state. Emotion forecasting is an under-explored area in the field of affective computing and to the best of our knowledge, our work is the first one to formulate and characterize the emotion forecasting problem in the deep learning domain.

For an effective Human-Computer interaction, understanding emotional state is undeniably important. Using audio-visual data, researchers have explored numerous aspects of emotion recognition and stepped towards more human-centered technology[1, 2]. On that account, with the rapid growth of smart technologies, prediction of future emotion or emotion forecasting from the current emotional state can be a promising research area. For example, in the field of predicting suicidal tendencies, Glen et. al. analyzed the short-term risk and transition from suicidal thoughts to attempts [3]. Therefore, if an intelligent agent can be developed to predict the future behavior of a potential suicide victim, the suicidal approach can be detected in the thought level before it reaches to the attempt level. In support to that, the work of Korrapati et. al. on social network site based suicidal prediction suggests the need for a robust and automated process of predicting
suicides [4]. Similarly, psychiatric research on predicting a patient’s likelihood of developing anxiety or panic disorder during psychotherapy can gain several advantages from the emotion forecasting research [5, 6]. For example, during a psychotherapy session, the chances of abandoning the therapy or leaving the conversation can be predicted using emotion forecasting and thus, such situation can be prevented.

Emotion forecasting also has potential business applications. In the sector of customer care and call centers, the possibility of success in marketing and achieving high service quality immensely depends on customer satisfaction. Thus, forecasting customer emotion from conversational dialog can be tremendously handy. To illustrate, in a conversation with customer while marketing a product, if the sales agent can forecast the customer’s interest or indifference, they can decide to continue or drop the conversation before the indifference turns into boredom or disgust. Emotion forecasting can also be used effectively for automated services like chat-bots, companion robots and various intelligent agents. For instance, the user emotion forecasting capability of Siri or Alexa can help when designing more emotion specific arrangements like music or tone of conversation to comfort the user [7, 8].

Therefore, emotion forecasting is a potential research area in the field of applying artificial intelligence in affective computing. In the upcoming sections, we analyze the research questions and discuss our experimental approach and results in detail.

The following publications were a direct result of the work presented in this thesis:


1.1 Problem Formulation

Emotion forecasting requires a new problem formulation, the setting-up of hypotheses, and the investigation of different methods. Previous works have demonstrated the effectiveness of exploiting the temporal properties of a time-varying signal for recognizing speech, emotion, and human-action [10, 11, 12]. Traditional emotion recognition research showed that considering the temporal flow of emotion dynamics helps modeling the recognition system with improved accuracy [13, 14]. This produces the first research question: Can we utilize the underlying salient information in behavioral cues to predict the future emotion? In addition to that, can sequential pattern analysis produce better insight than static pattern analysis in this case? To address this question, we present a hypothesis (H1):

In a dyadic conversational dialog, considering the sequential learning of information can enhance the emotion forecasting performance than in static learning.

Previous research in the affective computing domain demonstrated that when multiple speaking utterance steps from previous steps are considered, the emotion is recognized with better performance [15]. This phenomena implies that emotion can occur in a longer range of time. Hence, we present our second research question: Can emotional history be useful for forecasting emotion? To address this question, we introduce the second hypothesis (H2):

In comparison with the history-less performance of emotion forecasting, if along with the current utterance, the previous utterance is also taken into account, emotion forecasting performance will be improved.

Previous work has explored methods to predict future events from current or previous data; however, it is not considered whether and how these methods can be used
in emotion forecasting. For example, forecasting of human activity [16], financial event prediction [17], and facial action event prediction [18]. Hoai and Torre used partial information to detect the temporal facial actions [18]. However, as the emotions do not have a well-defined boundary like facial actions, the implementation of the research cannot be applied directly to emotion forecasting.
1.2 Approach

In this thesis, as an expressive conversational dataset, we use Interactive Emotional Dyadic Motion Capture (IEMOCAP), which is segmented into variable-length utterances [19]. Table 1.1 provides a detailed overview of the experiments used to test $H_1$ and $H_2$. The four experimental approaches are described below:

1. **UF-cur**: In UF-cur approach, we forecast emotion using only the current utterance. The forecasting window is an utterance step, which means the number of speaking turns after the current speaking turn.

2. **TF-cur**: TF-cur is the time forecasting approach using only the current utterance. The forecasting window is a time range.

3. **UF-his**: In UF-his approach, emotion is forecasted by utilizing both the current and previous utterance of a speaker.

4. **TF-his**: TF-his is the same approach as UF-his but the forecasting window is time instead of utterance steps.

We test our first hypothesis by experimenting with UF-cur and TF-cur approaches. We performed preliminary analysis with static models like Support Vector Machine, Random Forest, and Fully-Connected Deep Neural Network (FC-DNN), and we found that the FC-DNN model outperforms the other static machine learning models. Thus, we decided to use FC-DNN as a baseline static model. There are several dynamic modeling methods, including recurrent neural network (RNN) and its variants. We choose to use two popular deep RNN models, namely, deep long short-term memory (D-LSTM) and deep bidirectional long short-term memory (D-BLSTM) network. The deep learning models are summarized in brief in Table 1.2.

Our second hypothesis is tested using UF-his and TF-his approaches. We explore the potential of using additional history with the current data in order to forecast emotion. To
Figure 1.1: Total process of emotion forecasting. First, in block (a), we extract the audio-visual features to build up the data for emotion forecasting both in history-less and history-added setting. Next, we use static and dynamic model architecture to forecast four emotion categories. The facial figure is generated from [19].

Further demonstrate the relevance of history information, we experiment with a randomly added utterance instead of the previous utterance. The comparison between adding random and relevant data depicts the essence of context information in forecasting emotion.

Fig. 1.1 demonstrates the emotion forecasting overview. First, audio-visual features are extracted and processed in different forecasting windows. These features are fed into different deep learning models. Our analysis indicates that the temporal analysis of behavioral cues with the history context can provide better insights for future emotion prediction.
Table 1.1: Experimental approaches for evaluation of H1 and H2. The UF-cur and TF-cur approaches are used to test H1, and the remaining approaches are used to test H2.

<table>
<thead>
<tr>
<th>Forecasting Window</th>
<th>History</th>
<th>History-less (cur)</th>
<th>History-added (his)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utterance Forecasting (UF)</td>
<td>UF-cur</td>
<td>UF-his</td>
<td></td>
</tr>
<tr>
<td>Time Forecasting (TF)</td>
<td>TF-cur</td>
<td>TF-his</td>
<td></td>
</tr>
</tbody>
</table>

Table 1.2: Proposed and baseline experimental models for testing H1 and H2.

<table>
<thead>
<tr>
<th>Static Model</th>
<th>Temporal Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC-DNN</td>
<td>D-LSTM</td>
</tr>
<tr>
<td></td>
<td>D-BLSTM</td>
</tr>
</tbody>
</table>

1.3 Related Works

Forecasting the future event has been studied in the fields related to emotions. In human action recognition, Davis and Tyagi [20] presented a comparative probabilistic inference framework for a reliable human activity classification. Their work was to address the issue of rapid detection of human actions with low error rates. They focused on low-latency recognition with the shortest video exposure. A similar work was conducted by Ryoo [16], where he developed a bag-of-words model by leveraging the sequential nature of human activities. He tried to infer ongoing activities which leads to detecting the event before it finishes using time-dependent feature distribution. Early event detection is studied by Hoai and Torre [18], where they used a structured-Output Support Vector Machine (SOSVM) to introduce Max-Margin Early Event Detectors (MMED). Using MMED, they tried to detect and localize an event before its completion. In comparison with other even detectors, such as– Support Vector Machine SVM) or Hidden Markov Model (HMM), their method was faster and more reliable. In the field of emotion recognition, Kim and
Provost [21] investigated the time and duration pattern of emotional utterances. They analyzed the performance of emotion recognition by using only a portion of the emotional utterance. Their work drives us to hypothesize that emotional effects can be foreseen from longer time distances than only a specific utterance. Wöllmer et al. [15] explored the effect of including the past and future utterances along with the current one to recognize the emotion. However, their inclusion of future utterances to boost the D-BLSTM performance contradicts our purpose of timely-manner recognition of emotion. Therefore, we focus on forecasting, where we use the present and past temporal data to predict the future emotional state, and do not use the current utterance to make the system robust and efficient.

The fields not directly related to emotion also performed the task of forecasting by leveraging the machine learning techniques. Kiratzis et. al. developed an ANN based short-term load forecasting model [22]. They predicted the daily load curve in an autonomous power system. The forecasting of future events was also investigated in financial data research. Tay and Cao presented a comparative analysis on applying SVM and NN in financial time series.

Although forecasting actions or events from previous data were analyzed in many fields, predicting future emotion has not been explored much, with the exception of [23]. The sequence of emotional states was predicted in a work by Noroozi et al. In their work, they manually concatenated four emotional speech segments annotated as boredom, fear, joy and sadness and formulated a time series. A nonlinear autoregressive model was used to predict the next emotional group from the learned sequence. However, we have not made any fixed sequence or manual concatenation of emotional signals to predict the future emotion. Rather, we retain the conversation and forecast the next step from the current and past samples of data.
Chapter 2

Dataset Description

In this thesis, we use a large emotion database developed by the SAIL lab of University of Southern California-named IEMOCAP Database [19]. IEMOCAP contains approximately twelve hours of multi-modal data, which includes video, speech, motion capture of face and text transcriptions of dyadic conversation. The database is primarily segmented in sessions where two people participate in different conversations. These conversations have improvisations or scripted scenarios, specifically selected to elicit emotional expressions.

The IEMOCAP database is annotated by multiple annotators as well as self annotators into categorical labels, such as anger, happiness, sadness, neutrality, frustration, fear, surprise, disgust, and other unspecified categories. The annotators also rated the emotional categories into dimensional labels such as valence, activation and dominance. Ten actors were recorded in dyadic sessions in order to facilitate a more natural interaction and expression of the targeted emotion. The database contains information in following modalities: motion capture face information, speech, videos, head movement and head angle information, dialog transcript, and Word level, Syllable level and Phoneme level alignment. In this thesis, we use the most widely used modalities: speech and face information. We use the speech data and 46 three-dimensional MoCap trajectories—captured over six face regions: chin, forehead, cheek, upper eyebrow, eyebrow, and mouth—similar to [14].

From the sessions, IEMOCAP is divided in dialogs, where the participants engage in
Figure 2.1: A recording session of IEMOCAP database

a specific conversation. The IEMOCAP dialogs are segmented into variable-length utterances, where each utterance has a length of $4.77 \pm 3.34$ seconds. To be consistent with previous work on IEMOCAP [14, 21], we use only four categorical emotions, namely, angry, happy, neutral, and sad. The number of utterances over 10 speakers for each emotion are: $58.00 \pm 25.41$ for angry, $115.80 \pm 27.85$ for happy, $56.10 \pm 22.03$ for neutral, and $62.50 \pm 23.28$ for sad.

Figure 2.1 depicts the recording session of IEMOCAP dataset. As exhibited in the figure, the facial marker is used by only one of the participants. Thus, in this setting, we cannot use all 10039 utterances for audio-visual forecasting.
Chapter 3

Feature Description

3.1 Speech Features

One of the important modalities for expressing emotions is speech. Although the variations in speech are subjective, this is the most important human cues for recognizing human emotion. Research is being done to demonstrate which speech features need to be taken into account for emotion recognition. Most of the existing approaches to speech emotion recognition use acoustic features for classification input based on the acoustic correlation with emotion expressions. The most widely used speech or acoustic features are prosodic features e.g., pitch-related features, energy-related features, and speech rate and spectral features e.g., Mel-Frequency Cepstral (MFCC) or filterbank features. Kwon and Lee reported that pitch and energy among these are the strongest contributors in emotion recognition [24]. Neiburg et. al. used prosodic features (MFCC and pitch) to train a Gaussian Mixture Model classifier to recognize emotion [25]. In this thesis, we use pitch, MFCC, MFB, and energy as speech features. The following sections will give a brief description of those features.

3.1.1 Pitch

The pitch of the voice is one of main characteristics. In the field of acoustic technologies it is also known as the fundamental frequency. The period is the time taken by one vibratory
cycle of the vocal folds. The frequency of vibration is, therefore, the number of periods per second, which is measured in Hertz.

\[ 1 \text{ Hertz} = 1 \text{ vibration/second} \]  

(3.1)  

Now, in the human ear, we perceive the pitch as the frequency of sound which means the higher the frequency, the higher we perceive the pitch to be and vice-versa. The fundamental frequency or pitch is directly related to the rise or fall in the voice, also known as intonation. And the intonation is associated with the expressive characteristics of the voice, which leads to expressing emotion.

However, estimation of the fundamental frequency is a complicated task. Of many different methods of detecting pitch, the autocorrelation pitch detector is one of the most robust and reliable [26]. The autocorrelation computation works directly with the speech signal. Despite the requirement of a high processing rate, a single multiplier and an accumulator is sufficient for the hardware implementation. Moreover, the autocorrelation computation is not dependent on phase, making it robust in the case of usual phase distortions. However, there are several drawbacks associated with the use of this method. One of them is to determine which of several autocorrelation peaks is due to the detailed formant structure corresponds to the pitch period. Next, the computation requires the window of computation for the short-time autocorrelation function. The window method creates two questions: first, there is the problem of choosing an optimal window size. Second, with the increase in autocorrelation index, the effect of the window is to taper the autocorrelation function smoothly to zero. This effect tends to make the above difficulties more complex. Another problem is the subjective difference in the analysis window. For high-pitched speakers the analysis frame should be short (5-20 ms), and for low-pitched speakers it should be long (20-50 ms).

Researchers demonstrated many different solutions to this problem. Most of these solutions used a sharp low-pass filter with a cutoff frequency around 900 Hz. These
solutions eliminates the effect of higher formant peaks. However, there are trade-offs with this method which include elimination of all other higher formants, and resulting in loss of information. Another set of solutions suggest the flattening of speech signals for removal the first formant [27, 28]. The flattening is done by various methods, including center clipping and spectral equalization [27], inverse filtering, spectral flattening, and a Newton transformation [29].

We discuss the autorrelation process of pitch estimation in brief. Assume that the discrete time speech signal $x(n)$ is defined for all $n$ from $-N_0N + m$. We can describe the autorrelation function as:

$$\phi_x(m) = \lim_{N \to \infty} \frac{1}{2N + 1} \sum_{n=-N}^{N} x(n)x(n+m)$$  \hspace{1cm} (3.2)

The function $\phi_x$ performs a noninvertible transformation of the signal. It is mainly useful for depicting the waveform structure. Now, if our input signal $x(n)$ is periodic with period $P$, we can write,

$$\phi_x(m) = \phi_x(m+p)$$  \hspace{1cm} (3.3)

Equation 3.3 indicates that autocorrelation is also periodic with the same period as the main signal. The result can be turned around: periodicity in the autocorrelation function can be inferred as periodicity in the signal as well. Since the speech is a non-stationary signal, the long-term autocorrelation as described in equation 3.2 is not realistic. Hence, we present a short-term autocorrelation function, defined as:
\[ \phi_l(m) = \frac{1}{N} \sum_{n=0}^{N'-1} [x(n+l)w(n)][x(n+l+m)w(n+m)], \quad 0 \leq m \leq M_0 - 1 \]  

(3.4)

Here, \( w(n) \) is the appropriate window for analysis, \( N \) is the section length, \( N' \) is the number of signal samples used in the computation of \( \phi_l(m) \), \( M_0 \) is the number of the autocorrelation, and \( l \) denotes the index of the starting sample of the frame. In most research, the \( N' \) is empirically set to \( N - m \) so that only the \( N \) samples in the analysis frame.

### 3.1.2 Intensity

Sound intensity is the power carried by sound waves per unit area in a direction perpendicular to that area. Speech intensity is one of the powerful attributes through which emotion is expressed. Sound intensity \( I \) is defined as,

\[
\vec{I} = p\vec{v} \tag{3.5}
\]

where, \( p \) is the sound pressure and \( \vec{v} \) is the velocity of sound. As velocity and intensity both are vectors, they have magnitude and direction. The direction of the sound intensity is the average direction of the energy flow. Therefore, the average sound intensity can be written as,

\[
\bar{I} = \frac{1}{T} \int_0^T p(t)\vec{V}(t)dt \tag{3.6}
\]

Now, for spherical propagation of sound at a radius \( r \), intensity equation can be written as,
\[ I(r) = \frac{P}{A(r)} = \frac{P}{4\pi r^2} \]  \hspace{1cm} (3.7)

where, \( P \) is the power of sound and \( A(r) \) is the surface area of the sphere through which the sound is propagating.

The intensity of sound is also calculated with reference to a commonly used sound intensity. The unit of such measurement is decibel (dB).

\[
\text{sound\_intensity\_level}(dB) = 10 \log \frac{I}{I_0} \]  \hspace{1cm} (3.8)

where, \( I \) is the sound intensity and \( I_0 \) is the reference sound intensity, which is 1 \( \text{pW/m}^2 \) in the air.

### 3.1.3 Mel-Frequency Cepstrum

Sound is not interpreted by human auditory system in a linear scale. As the frequency is increased, the comprehension of pitch becomes logarithmic. Therefore, in cases like emotion, where human perception is more important than actual sound properties, the mel-frequency cepstrum becomes significant. In speech signal processing, the mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound. MFCs are based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. Mel-frequency cepstral coefficients (MFCCs) are coefficients that build an MFC. In the mel-frequency cepstrum, frequency bands are equally spaced in mel-scale which has close relation to human ear perception.

There are several steps involved in the calculation the MFCC. First, we apply a pre-emphasis filter on the signal to amplify the high frequencies. A pre-emphasis filter is
useful in balancing the frequency spectrum and improving the Signal–to–Noise Ratio (SNR). After that, signals are split into frames. Speech is a non-stationary signal and thus, its properties vary with time. Therefore, we need short frames to make a reasonable Fourier transform. In this way, we can obtain a better approximation of the frequency contours of the speech signal by concatenating adjacent frames. After the framing step, we apply a Hamming window function to the frame. Using a window function reduces the spectral leakage. The Hamming window has the following form:

$$w[n] = 0.54 - .46 \cos \left( \frac{2\pi n}{N-1} \right)$$ \hspace{1cm} (3.9)

Here, N is the window length. A time and frequency domain representation is depicted in figure 3.1 where 100 samples are used.

Then, we perform a N-point FFT on each windowed frame to calculate the frequency spectrum. This process is known as the Short-Time Fourier-Transform (STFT), where N is empirically chosen as 256 or 512, and it must be greater than the number of the frames. If we have X number of frames, then N − X number of zero-padding is used after the
frames. The power spectrum is calculated using the equation:

\[ P = \frac{|FFT(x_i)|^2}{N} \]  

(3.10)

Here, \( x_i \) is the \( i^{th} \) frame for speech signal \( x \).

At the final step, we generate a bank of filters using the FFT coefficients by applying triangular filters on a mel-scale to the power spectrum to extract frequency bands. The following formula is used to convert the frequency scale to the mel-scale.

\[ M(f) = 1125 \ln \left(1 + \frac{f}{700}\right) \]  

(3.11)

To convert back to frequency scale,

\[ M^{-1}(m) = 700(\exp \frac{m}{1125} - 1) \]  

(3.12)

The first filter will start at the first point, reach its peak at the second point, then return to zero at the 3rd point. The process goes on in the same way for the following filters. The set of filters are defined as:

\[
H_m(k) = \begin{cases} 
0, & k \leq f(m-1) \\
\frac{k-f(m-1)}{f(m)-f(m-1)}, & f(m-1) \leq k \leq f(m) \\
\frac{f(m+1)-k}{f(m+1)-f(m)}, & f(m) \leq k \leq f(m+1) \\
0, & k > f(m+1) 
\end{cases}
\]
Now the filter bank coefficients will be highly correlated, and such features can create instability in the machine learning algorithm. Hence, we can apply Discrete Cosine Transform (DCT) to the filter bank coefficients to diminish the correlation effect and produce a compressed representation of the filter banks. This is how the MFCCs are obtained.

### 3.2 Facial Features

Facial expressions are the most important modality for recognizing emotion. The facial features are based on the local spatial position or displacement of the three-dimensional position of facial regions. Pantic et. al. provides a detailed review on facial features [30]. Emotion recognition from facial muscle movement was studied by Mase [31]. He used optical flow to detect the muscle movement in 11 facial regions. K-nearest neighbor rule was used as classification algorithm, with an accuracy of 80%. Similar research was conducted by Yacoob et. al. [32] where, instead of using facial muscle actions, they built a dictionary which converted motions associated with the edge of the mouth, eyes and eyebrows, into a linguistic, perframe, mid-level representation. Their classification system used a rule-based approach with an 88% accuracy score. One of the major applications of facial features involved Action Units (AU), which was developed by Ekman and Friesen [33]. For example, Tian et al. recognized the AUs using lip, nasolabial furrow and wrinkles [34]. They located the shape and placement of those features by leveraging geometrical models with 96% accuracy. Essa et al. quantified facial movements based on parametric models [35].

In the IEMOCAP database, the system is based on visual information, obtained using motion capture markers as shown in figure 3.2. The spatial 3-D data collected from the motion capture markers in each frame of the video is transformed into a 4-dimensional feature vector per sentence. Those vectors are used as input to the classifier. After the motion data are captured, they are normalized. First, all markers are translated in a way
that a nose marker is the local coordinate center of each video frame. Then, as a reference frame, one frame with neutral and closed-mouth head pose is selected. After that, three approximately rigid markers define a local coordinate origin for each frame. Next, a rotation of the coordinates are performed to align them with the reference frame. Each data frame has the following blocks of the facial region- forehead, eyebrow, low eye, right cheek and left cheek area. For each block, a data vector is produced by a concatenation of the 3D coordinate positions. Then, the Principal Component Analysis (PCA) based method is implemented to reduce the number of features per frame into a 10-dimensional vector for each area. Use of the PCA to feature reduction covers more than 99% of the variation.
Chapter 4

Deep-Learning Architecture

Deep learning (DL) has established a new era in the field of machine intelligence. With its effectiveness, DL is making tremendous impact in areas, like disease diagnosis, precision medicine, self-driving cars, predictive forecasting, facial and speech recognition, and emotion recognition. For large datasets, one of the major and painstaking tasks is building the handcrafted features in traditional machine learning algorithms. These algorithms are not scalable for large-sized data sets, resulting in poor performances. Deep learning with human-like neuron representation contains several layers of units with highly optimized algorithms and structures. The nodes or units in the layers are connected to nodes in adjacent layers with some weight value. The computed values undergo a transformation based on the activation function, which can be sigmoid or ReLU. By performing the backpropagation, the layers rectify their errors and learn imitating human brain.

The first prototype of Deep Learning was utilized by a perceptron [36]. In that work, Rosenblatt used two layers of processing units that could detect simple patterns. However, after a publication of MIT suggesting the inefficiency of such network, neural network entered a dark phase [37]. Although proposed in 1975 by Werbose, a breakthrough in deep learning occurred in 2015 through the revolutionary work by LeCun et. al. [38].

The backpropagation algorithm is the key idea behind the deep learning models. Utilizing the backpropagation, several variants of the network architecture was proposed and implemented, such as, FC, LSTM, BLSTM, CNN, GAN and many more [39, 40, 41, 42].
In this thesis, we used a FC, LSTM and BLSTM architecture. In the next subsection, these architectures will be discussed in brief.

4.1 Fully-Connected Neural Network

Fully-Connected deep learning frameworks are the most basic architecture of deep learning. By utilizing multiple connections, these networks learns the feature importance and map the output to input.

Let’s assume that we have a dataset containing $n$ data-points with each of them having three input features $x_1, x_2, x_3$, and corresponding output $y_1, y_2, y_3$. If the dataset is large or complex enough, a simple logistic regression model will not be sufficient to map the complexity. Hence, we need a neural-network mapping for the representation of the features.

In figure 4.1, a simple neural network for such a task is presented. The first layer is input layer, which is defined as $a^{(1)}$. Then, the input is mapped to a hidden layer $a^{(2)}$ by
some weights, from which we go to a final layer. The final layer simply transforms the output into our desired representation. Based on a backward propagation algorithm, the network learns from its errors. Therefore, we need to discuss the steps of backpropagation in detail.

Backpropagation is an algorithm that calculates the gradient which is needed to adjust the weights of the neural network. First introduced in 1970 [43], the backpropagation algorithm was brought into light by Rumelhart et. al. in 1986 [44]. The algorithm uses partial derivative of the objective function with different weight parameters, quantifying how fast the objective function changes with change in weight.

A Neural network also uses activation functions. There are many choices of activation function, but popular choices include sigmoid and ReLU. A detailed representation of the activation functions are presented in figure 4.2.
We can derive the equation for different units in layer 2 from figure 4.1.

\[
a_1^{(2)} = g(\theta_{10}^{(1)} x_0 + \theta_{11}^{(1)} x_1 + \theta_{12}^{(1)} x_2 + \theta_{13}^{(1)} x_3) \tag{4.1}
\]

\[
a_2^{(2)} = g(\theta_{20}^{(1)} x_0 + \theta_{21}^{(1)} x_1 + \theta_{22}^{(1)} x_2 + \theta_{23}^{(1)} x_3) \tag{4.2}
\]

\[
a_3^{(2)} = g(\theta_{30}^{(1)} x_0 + \theta_{31}^{(1)} x_1 + \theta_{32}^{(1)} x_2 + \theta_{33}^{(1)} x_3) \tag{4.3}
\]

\[
h_\theta(x) = a_1^{(3)} = g(\theta_{10}^{(2)} a_0^{(2)} + \theta_{11}^{(2)} a_1^{(2)} + \theta_{12}^{(2)} a_2^{(2)} + \theta_{13}^{(2)} a_3^{(2)}) \tag{4.4}
\]

In the above set of equations, the term \( \theta \) with different subscript and superscript denotes the weight associated from units of one layer to the units of the other. For example, \( \theta_{ij}^{(k)} \) denotes the connection weight from \( j \)th unit of \( k \)th layer to the \( i \)th unit of \( (k + 1) \)th layer. \( x_0 \) and \( a_0 \) are the bias terms. The term \( a_i^{(j)} \) is denoted as \( i \)th unit in \( j \)th layer. The function \( g(.) \) represents the activation function. This is called the forward propagation (FP).

If we consider the layers as vectors, we can write the FP equation set as follows:

\[
a^{(1)} = x \tag{4.5}
\]

\[
z^{(2)} = \theta^{(1)} a^{(1)} \tag{4.6}
\]

\[
a^{(2)} = g(z^{(2)}) \tag{4.7}
\]

\[
z^{(3)} = \theta^{(2)} a^{(2)} \tag{4.8}
\]

\[
a^{(3)} = g(z^{(3)}) \tag{4.9}
\]

\[
z^{(4)} = \theta^{(3)} a^{(3)} \tag{4.10}
\]

\[
a^{(4)} = h_\theta(x) = g(z^{(3)}) \tag{4.11}
\]

Initially, only the network architecture is defined and thus, the weights are randomized. Therefore, there will be error term and we represent the error in an objective function (also known as cost function). If the dataset has \( m \) points and \( K \) classes to classify, we
can write the objective function as:

\[ J(\theta) = -\frac{1}{m} \left[ \sum_{i=1}^{m} \sum_{k=1}^{K} y_k^{(i)} \log(h_\theta(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - h_\theta(x^{(i)}))_k \right] \]  (4.12)

As evident from equation 4.12, the cost function \( J(\theta) \) is the function of weight parameter \( \theta \). Therefore, our target is to find such a group of weights, \( \theta \), that will minimize \( J(\theta) \). We will introduce an error term \( \delta \), which we can obtain by the partial differentiation of the objective function with respect to \( \theta \). By chain differentiation, it can be shown that,

\[ \frac{\partial J(\theta)}{\partial \theta^{(l)}_{ij}} = a_j^{(l)} \delta_i^{(l+1)} \]  (4.13)
We can calculate the error term for each node or unit. $\delta_l^j$ represents the error of unit $j$ in layer $l$.

---

**Algorithm 1: Backpropagation algorithm**

**Result:** Partial derivation for each parameter. Vector $D^{(l)}$

- set $\Delta_{ij}^l = 0$;
- set $i = 1$;

**while** $i \leq m$ **do**

- Perform forward propagation to compute $a^l$ for each layer;
- Compute the error term $\delta^l$;
- **if** this is final layer **then**
  - $\delta^{L} = a^{L} - y^i$;
- **else**
  - $\delta^l = \theta^l \delta^{(l+1)} \cdot (a^l \cdot (1 - a^l))$;
- **end**

- Update $\Delta_i$;
  - $\Delta_{ij}^{(l)} := \Delta_{ij}^{(l)} + a_{j}^{(l)} \delta_{i}^{(l+1)}$;

**end**

$D^{(l)} = \frac{1}{m} \Delta^{(l)}$;

---

In our EF problem, we use an FC network with four layers and 256 units per layer, and ReLU is used as the activation function except for the final layer which uses softmax with four units.
4.2 Recurrent Neural Network (RNN)

Recurrent neural networks have the ability to comprehend sequential data. One of the major drawbacks of FC networks is that features are mapped into a representation without consideration of their interdependence. RNN provides an architecture to solve this problem.

In figure 4.3, a many-to-one RNN architecture is shown. RNN takes a sequence as features. Then inside the hidden layer, the hidden units can be thought of as multiple copies of the same network, each passing a message to a successor. Of many variants of RNN, Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BLSTM) networks are the most popular. The detail of these networks are explained in the next section.

4.2.1 Long Short-Term Memory Network (LSTM)

The LSTM network is an enhanced version of the RNN that has the added advantage of handling the vanishing gradient problem [45]. An LSTM layer is composed of recurrently
connected memory blocks, where the memory cells have three gate units: the input, output, and forget gates. Briefly, the cell input is multiplied by the activation of the input gate, the cell output by that of the output gate, and the previous cell values by the forget gate [14]. Thus, information is retained and stored over an extensive time period. For BLSTM, the network is composed of both the forward and backward directions of LSTM.

The internal cell state of an LSTM network \( (c^\tau) \) at time \( \tau \) is computed based on the current input \( (x^\tau) \) and the previous cell state \( (c^{\tau-1}) \). The combination of input \( (i^\tau) \) and forget \( (F^\tau) \) gates control how much the previous cell state \( c^{\tau-1} \) and the current input \( x^\tau \) contribute to the current cell state \( c^\tau \). The activation function for both forget and input gates is a sigmoid function \( (\sigma) \) that outputs values between 0 and 1. The current cell state

![Figure 4.4: The basic cell structure of LSTM](image-url)
$c_τ$ is determined by the following equations:

$$i_τ = \sigma(W_{xi}x_τ + W_{hi}h_τ^{-1} + W_{ci}c_τ^{-1} + b^i) \quad (4.14)$$

$$F_τ = \sigma(W_{xf}x_τ + W_{hf}h_τ^{-1} + W_{cf}c_τ^{-1} + b^f) \quad (4.15)$$

$$\tilde{c}_τ = \tanh(W_{xc}x_τ + W_{hc}h_τ^{-1} + b^c) \quad (4.16)$$

$$c_τ = F_τ \ast c_τ^{-1} + i_τ \ast \tilde{c}_τ \quad (4.17)$$

$$o_τ = \sigma(W_{xo}x_τ + W_{ho}h_τ^{-1} + W_{co}c_τ^{-1} + b^o) \quad (4.18)$$

$$h_τ = o_τ \ast \tanh(c_τ) \quad (4.19)$$

where $o_τ$ is the output gate that determines the contribution of current cell state $c_τ$ to the current output $h_τ$. We can also rewrite $h_τ$ and $c_τ$ as follows:

$$(h_τ, c_τ) = G(x_τ, h_τ^{-1}, c_τ^{-1})$$

where $G$ is the LSTM activation function.
4.2.2 Bidirectional Long Short-Term Memory Network (BLSTM)

To process the input sequence in both forward and backward directions, the bidirectional LSTM (BLSTM) is used to capture the dynamics in both past and future samples. For BLSTM, $h^\tau$ will be composed of both the forward and backward directions, $h^\tau = [\vec{h}^\tau; \overrightarrow{h}^\tau]$ and it is defined as follows:

$$
(\vec{h}^\tau, \vec{c}^\tau) = \mathcal{G}(x^\tau, \vec{h}^{\tau-1}, \vec{c}^{\tau-1}) \quad (4.20)
$$

$$
(\overrightarrow{h}^\tau, \overrightarrow{c}^\tau) = \mathcal{G}(x^\tau, \overrightarrow{h}^{\tau-1}, \overleftarrow{c}^{\tau-1}) \quad (4.21)
$$

Our main architecture will contain deep LSTM (D-LSTM) / BLSTM (D-BLSTM) networks, followed by a FC-DNN.
Chapter 5

Methodology

5.1 Feature Extraction and Data Processing

Following the work of Metallinou et al. [14], both the audio and visual data were extracted at the same frame rate of 40Hz and the overlapping window of 50 ms.

For the audio data, we extract pitch, energy, 12 Mel Frequency Cepstral Coefficients (MFCC) and 27 Mel Filter Bank (MFB) coefficients, and we use Praat for feature extraction [46]. The 46 total 3-D markers allow for 138-dimensional facial landmark representation. We use linear interpolation to remove the NaN values which occurred due to noisy sensor reading. If the total amount of NaN values in an utterance is greater than 30% of total frames, we exclude the utterance. Additionally, we exclude utterances with the audio data that has zero pitch in all frames since these utterances do not contain meaningful audio information.

We use utterance-level statistical features, namely, mean, standard deviation, first quantile, third quantile, and inter-quartile range. Therefore, building in the 179 frame-level features (41 audio and 138 facial landmark features), we have 895 statistical features in total. The features are normalized speaker-wise using z-normalization. D-LSTM and D-BLSTM use window-level features. The analyzing windows are made by taking statistical features over 30 frames with a 50% overlap. As the IEMOCAP utterances have
different length, we use zero-pad and later used a masking layer [47]. All the experiments were done using Keras [47].

5.2 Emotion Forecasting Methods

We analyze emotion forecasting based on the forecasting windows and presence of the previous utterance history in forecasting.

5.2.1 Forecasting Windows

Due to the difference in utterance length in the IEMOCAP database, we present two approaches of forecasting windows. The utterance forecasting will consider only the speaking turns after which forecasting will be done, while the time forecasting will consider the time amount for forecasting.

Utterance Forecasting (UF)

For forecasting with utterance steps, we consider the emotional flow of one participant in a dialog only. For 1 utterance forecasting (UF 1), we choose the data of the present utterance and the label for the next utterance of the same speaker in the same dialog. Similarly, for $k$-utterance forecasting (UF $k$), the data will come from current utterance and the label will come from the utterance of $n$ future steps.

To illustrate, let $L$ be the set of all utterance labels of a particular speaker $A$, and $U$ be the set of all utterance feature data of $A$. If $A$ has $n$ utterances, then $L = \{L_1, L_2, L_3...L_n\}$, and

$$U = \{Utt_1^A, Utt_2^A, Utt_3^A ...Utt_n^A\}$$

where $L_x$ is the label of utterance $Utt_x^A$, where $x = 1,2,3...n$. 
As in IEMOCAP, a dialog contains a complete scenario of emotional conversation, the forecasting has to be done within a dialog. For instance, the mapping of feature data to the labels for a $k$ utterance forecasting will be $\text{Utt}_1^A \rightarrow L_{1+k}$ and so on. If the dialog has $d$ number of utterances, then for the $d$th utterance, $\text{Utt}_d^A$, we cannot have the $L_{d+k}$ labels. If $k > 1$, we will have more utterances in every dialog which have no labels and hence, must be discarded. Hence, the number of the utterances decreases in utterance forecasting (UF) 1, 2, and 3, as stated later in this section.

The process is described in Fig. 5.1, which depicts the dataset re-processing for UF 2. If in a dialog, the conversation alternate between speaker A and speaker B: $A \rightarrow B \rightarrow A \rightarrow B \rightarrow A$, then the utterance $\text{Utt}_3^A$’s label would be used for the data of utterance $\text{Utt}_1^A$. UF 1, 2, and 3 holds 2823, 2734, and 2662 utterances consecutively. These utterances have a mean time-distance (explained in the next section) of $4.71 \pm 5.37$ seconds $12.78 \pm 8.55$ seconds and $21.48 \pm 10.78$ seconds, respectively. Note that, for utterance $\text{Utt}_1^A$, the forecasting label has to pass one utterance of the other speaker (B), while utterance $\text{Utt}_2^A$ has to pass two utterances of the other speaker. This creates inconsistency in the dataset and results in a large standard deviation of time-distance among the steps.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig5.1.png}
\caption{An example of emotion forecasting without history. Here, the data of $\text{Utt}_1^A$ will be used to forecast the emotion of the utterance $\text{Utt}_3^A$.}
\end{figure}
Time Forecasting (TF)

Due to the inconsistency explained in previous section, we introduce the *time-distance* through which future emotion will be forecasted. Time-distance is defined as the time length from the current utterance to the forecasted utterance’s emotional label for UF 1, 2 or 3. Additionally, to calculate the time-distance, we also consider the length of the other speaker’s utterance that falls in between the utterances of the same speaker. The calculation of time-distance starts from the middle point of the current utterance and ends at the middle point of the target utterance whose emotion is to be forecasted.

We divide the time-distances into several groups of 5-seconds range. TF 1 will have a forecasting time-distance of 3 second to 8 seconds, TF 2 will have a time-distance length of 8 seconds to 13 seconds, and TF 3 will have a time-distance of 13 seconds to 18 seconds. For example, in Fig. 1, the first forecasting will have a time-distance of 9.5 seconds \((2.5 + 2 + 1 + 4)\), and the second forecasting will have a time-distance of 14 seconds \((1 + 1 + 8 + 3 + 1)\). Therefore, although in terms of the forecasting window of UF, they will fall into same UF 2, in terms of TF, the first one will be part of TF 2 and the second one will be a part of TF 3. For TF 1, 2 and 3, we have 1800, 1790, and 1725 utterances. The mean time-distance for TF 1, 2 and 3 are \(5.56 \pm 1.36\) seconds, \(10.40 \pm 1.43\) seconds, and \(15.37 \pm 1.42\) seconds.

It can be mentioned that we also experiment with longer step UF and TF. As the steps increase, the number of utterances decrease and the average time-distances increase. Moreover, we also observe a gradual drop in the performance and hence, we stick to the step three for both UF and TF.
5.2.2 Absence or Presence of Previous History

History-Less Emotion Forecasting

In the history-less technique, we experiment with emotion forecasting using the current audio-visual data only with both forecasting windows, UF and TF.

History-Added Emotion Forecasting

In this technique, in addition to the present emotional data, the prior utterance’s data will also be used to forecast future emotion. An example is illustrated in Fig.5.2, where for 2 utterance forecasting (UF 2), along with the current utterance $Utt_A^2$, the previous utterance $Utt_A^1$ is also taken into account and the features are concatenated. For instance, considering the illustration of Section 5.2.1, for a $k$ utterance forecasting, the data will be,

$$U = \{Utt_1^A, Utt_{1,2}^A, Utt_{2,3}^A...Utt_{n-k-1,n-k}^A\}$$

and the labels will be $L = \{L_{1+k}, L_{2+k}...L_n\}$ where

$$Utt_{n-k-1,n-k}^A = concatenation(Utt_{n-k-1}^A, Utt_{n-k}^A)$$

We conduct the history-added emotion forecasting using UF-his and TF-his approaches.
FIGURE 5.2: An example of emotion forecasting with added history from the previous utterance. Here to forecast the emotion of $Utt_A^4$, the data of both $Utt_A^1$ and $Utt_A^2$ are used.

5.3 Experimental Setup

Our FC-DNN model contains three fully connected layers with 256 memory units and one output softmax layer. The D-LSTM uses two LSTM layers with 128 memory units each. After that, an FC layer is used with RELU as activation and, finally, a softmax output layer is stacked with the network. The D-BLSTM is built up with similar architecture with the LSTM layers having 128 memory units in both direction. For all FC-DNN, D-LSTM and D-BLSTM, a rectified linear unit (RELU) is used as the activation unit. We use a learning rate of 0.0001 with the adam [48] optimizer and a batch size of 128. We also use an early stopping criteria, where the training stops if the validation accuracy does not improve after ten consecutive epochs.

To measure the performance, we calculate the Unweighted (Average) Recall (UAR), which is defined as the mean recall of all the emotion classes over the ten test speakers. We perform leave-one-speaker-out cross-validation. The validation accuracy is used for choosing the number of epochs for early stopping. As suggested in previous research [49], we use a paired t-test over ten speakers for testing the significance level of difference between different approaches. We claim significance when $p<0.05$. 

35
Chapter 6

Results and Discussion

As stated in Section 1.2, we use the UF-cur and TF-cur approaches for addressing $H_1$ and UF-his and TF-his for addressing $H_2$.

6.1 History-Less Emotion Forecasting

Table 6.1 summarizes the results of UF-cur and TF-cur experiments. For UF-cur 1, 2, and 3, the results support $H_1$. Although not statistically significant ($p>0.05$), we achieve 0.97%, 1.28%, and 1.59% improvement using D-LSTM over FC-DNN, while we observe 0.97%, 1.91%, and 0.85% improvement using D-BLSTM. While we achieve a better accuracy with D-BLSTM than FC-DNN, it does not seem to work better than D-LSTM, particularly UF-cur 3. This may indicate that, as D-BLSTM has more complex modeling process than D-LSTM and it needs a large amount of data as well, our data quantity may be insufficient (2662 utterances for UF 3) for that purpose.

Table 6.1 also shows the performance comparison for TF-cur 1, 2, and 3. It shows that D-LSTM performs better than the FC-DNN model by 0.32%, 0.73%, and 2.42% ($p<0.05$) for TF-cur 1, 2 and 3 respectively. When we consider both forward and backward dynamics by taking into account a bidirectional temporal model (D-BLSTM), we find better accuracy than FC-DNN model by 0.76%, 1.19%, and 1.60% ($p<0.05$) respectively. Therefore, the above experiments supports our hypothesis $H_1$ that temporal model can perform
Table 6.1: UAR(%) comparison between static (FC-DNN) and dynamic (D-LSTM and D-BLSTM) models for both UF-cur and TF-cur approaches. The sign “[*]” denotes that the model is statistically significant ($p<0.05$) compared to the FC-DNN.

<table>
<thead>
<tr>
<th>Forecasting Window</th>
<th>FC-DNN</th>
<th>D-LSTM</th>
<th>D-BLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>UF-cur 1</td>
<td>57.96</td>
<td>58.93</td>
<td>58.95</td>
</tr>
<tr>
<td>UF-cur 2</td>
<td>54.45</td>
<td>55.73</td>
<td>56.36</td>
</tr>
<tr>
<td>UF-cur 3</td>
<td>51.40</td>
<td>52.99</td>
<td>52.25</td>
</tr>
<tr>
<td>TF-cur 1</td>
<td>57.05</td>
<td>57.37</td>
<td>57.81</td>
</tr>
<tr>
<td>TF-cur 2</td>
<td>55.29</td>
<td>56.02</td>
<td>56.48</td>
</tr>
<tr>
<td>TF-cur 3</td>
<td>52.93</td>
<td>55.35[*]</td>
<td>55.53[*]</td>
</tr>
</tbody>
</table>

better than the static model.

6.2 History-Added Emotion Forecasting

As stated previously, adding history to the present information incorporates more temporal contexts, which can result in an enhanced forecasting performance. Table 6.2 describes the results of both UF and TF approaches with the history of previous utterance added. Adding history improved the D-LSTM accuracy for UF-his 1, 2, and 3 by 1.75%, 0.78%, and 0.34% compared to UF-cur D-LSTM approaches. We observe even more improvement in D-BLSTM performance. D-BLSTM is improved by 2.38% ($p<0.05$), 2.01%, and 2.39% ($p<0.05$) for UF-his 1, 2, and 3.

For the TF-his approach, both the D-LSTM and D-BLSTM performance improves over the history-less performance. We observe an improvement of 1.64%, 2.25% ($p<0.05$), and 2.37% ($p<0.05$) for D-LSTM and 1.52%, 1.95%, and 2.87% ($p<0.05$) for D-BLSTM in TF-his 1, 2, and 3 than the TF-cur approaches. Moreover, Table 6.2 shows that in every case, unlike UF-cur result, the D-BLSTM outperforms D-LSTM performance. This may imply that D-BLSTM can process the information in a better way when adequate history is provided.

Furthermore, instead of previous utterance, we add a randomly chosen utterance of
Table 6.2: Results of emotion forecasting with added history information: performance of UF-his and TF-his with dynamic modeling. The sign "[*]" denotes that the model with history-added technique is statistically significant \((p<0.05)\) compared to the history-less technique of the model.

<table>
<thead>
<tr>
<th>Forecasting Window</th>
<th>D-LSTM</th>
<th>D-BLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>UF-his 1</td>
<td>60.68</td>
<td>61.33 [*]</td>
</tr>
<tr>
<td>UF-his 2</td>
<td>56.51</td>
<td>58.37</td>
</tr>
<tr>
<td>UF-his 3</td>
<td>53.33</td>
<td>54.64 [*]</td>
</tr>
<tr>
<td>TF-his 1</td>
<td>59.01</td>
<td>59.33</td>
</tr>
<tr>
<td>TF-his 2</td>
<td>58.27 [*]</td>
<td>58.43</td>
</tr>
<tr>
<td>TF-his 3</td>
<td>56.72 [*]</td>
<td>58.40 [*]</td>
</tr>
</tbody>
</table>

the same speaker (i.e., UF-random and TF-random) to compare the effect of adding random data to the current utterance rather than the history. Table 6.3 shows that for every case, compared to UF-random or TF-random, UF-his and TF-his performs significantly better \((p<0.05)\). The UF-his 1, 2, and 3 has an improvement of 5.58\%, 4.50\%, and 4.65\% respectively with D-LSTM and 4.69\%, 5.64\%, and 5.30\% respectively with D-BLSTM. Similarly, for TF-his 1, 2, and 3, the improvement is 3.97\%, 5.83\%, and 5.71\% respectively with D-LSTM and 4.31\%, 6.13\%, and 7.53\% respectively with D-BLSTM, compared to the randomly added data. As adding random utterance in the data may harm the emotional temporal flow, we observe a diminished UAR performance. It implies that our improvement in history-added D-BLSTM performance is not merely coming from adding extra information, rather, by taking the proper history context into account. Therefore, we show that adding history information from one previous utterance can add an emotional context for the network to see the temporal flow pattern of the utterances and predict the future emotion, and hence, the results support H2.

Figure 6.1 depicts the comparison of performance with different forecasting methods using deep learning. The reason for the D-BLSTM network’s superior performance in both UF-his and TF-his experiments may be that, as a more complex modeling framework compared to the D-LSTM, the D-BLSTM model can see both past and future information. Thus, by achieving the history information, it can achieve substantial information
6.3 Further Analysis

We further investigate the performance of utterances that have the same target forecasting and current emotion labels. We examine whether the performance of emotion forecasting is biased toward that of emotion recognition. We define the term same-label as, for any forecasting approach, the ratio of instances, where forecasting and recognition ground truths are same, to all the forecasting ground truth labels. We find that in all forecasting approaches, more than two-third of the forecasting utterances have the same ground truth labels with the recognition. We analyzed the emotion specific performances of such cases. For example, for UF-cur 3, we observe the highest UAR with the ‘Happy’ emotion (83.25%), and the lowest with the ‘Neutral’ emotion (36.55%). However, when the forecasting label is different from current emotion label, we achieve lower UAR as expected. For instance, UF-cur 3 D-BLSTM results achieve up to 36.72% for angry, 41.30% for happy, 17.09% for neutral, and 32.10% for sad classes. Hence, the results imply that the training
TABLE 6.3: UAR (%) comparison of UF-his/TF-his with UF-random/TF-random. The sign “[*]” denotes the statistically significant (p<0.05) result of history added technique, compared to the randomly added data.

<table>
<thead>
<tr>
<th>Forecasting Window</th>
<th>D-LSTM</th>
<th>D-BLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>UF-his 1</td>
<td>60.68 [*]</td>
<td>61.33 [*]</td>
</tr>
<tr>
<td>UF-random 1</td>
<td>55.10</td>
<td>56.62</td>
</tr>
<tr>
<td>UF-his 2</td>
<td>56.51 [*]</td>
<td>58.37 [*]</td>
</tr>
<tr>
<td>UF-random 2</td>
<td>52.01</td>
<td>52.73</td>
</tr>
<tr>
<td>UF-his 3</td>
<td>53.33 [*]</td>
<td>54.64 [*]</td>
</tr>
<tr>
<td>UF-random 3</td>
<td>48.68</td>
<td>49.34</td>
</tr>
<tr>
<td>TF-his 1</td>
<td>59.01 [*]</td>
<td>59.33 [*]</td>
</tr>
<tr>
<td>TF-random 1</td>
<td>55.04</td>
<td>55.02</td>
</tr>
<tr>
<td>TF-his 2</td>
<td>58.27 [*]</td>
<td>58.43 [*]</td>
</tr>
<tr>
<td>TF-random 2</td>
<td>52.44</td>
<td>52.30</td>
</tr>
<tr>
<td>TF-his 3</td>
<td>56.72 [*]</td>
<td>58.40 [*]</td>
</tr>
<tr>
<td>TF-random 3</td>
<td>51.01</td>
<td>51.97</td>
</tr>
</tbody>
</table>

and testing data used for the forecasting experiments may be biased towards the current emotion labels. Therefore, an extensive analysis on forecasting window is needed to develop a more robust emotion forecasting system.

TABLE 6.4: The ratio of identical labels of forecasting and recognition, to all the forecasting labels for different forecasting windows (i.e., same-labels).

<table>
<thead>
<tr>
<th>Forecasting Window</th>
<th>Same-labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>UF 1</td>
<td>0.73</td>
</tr>
<tr>
<td>UF 2</td>
<td>0.68</td>
</tr>
<tr>
<td>UF 3</td>
<td>0.64</td>
</tr>
<tr>
<td>TF 1</td>
<td>0.74</td>
</tr>
<tr>
<td>TF 2</td>
<td>0.71</td>
</tr>
<tr>
<td>TF 3</td>
<td>0.67</td>
</tr>
</tbody>
</table>
TABLE 6.5: An example for the per-class recall score (%), when the forecasting label is same with current emotional label. The example is taken from UF-cur 3 D-BLSTM result. ‘GT’ refers to forecasting ground truth and ‘Predict’ refers to forecasting labels predicted by the network.

<table>
<thead>
<tr>
<th>Predict</th>
<th>GT</th>
<th>Angry</th>
<th>Happy</th>
<th>Neutral</th>
<th>Sad</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>65.66</td>
<td>19.25</td>
<td>12.08</td>
<td>3.02</td>
<td></td>
</tr>
<tr>
<td>Happy</td>
<td>6.72</td>
<td>83.25</td>
<td>6.60</td>
<td>3.42</td>
<td></td>
</tr>
<tr>
<td>Neutral</td>
<td>28.99</td>
<td>25.63</td>
<td>36.55</td>
<td>8.82</td>
<td></td>
</tr>
<tr>
<td>Sad</td>
<td>7.44</td>
<td>23.67</td>
<td>5.85</td>
<td>63.03</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 7

Conclusion

This thesis describes the formulation of emotion forecasting techniques from current and past audio-visual data. For UF-cur and TF-cur approaches, we demonstrate the dynamic models (D-LSTM and D-BLSTM) always outperform the static model (FC-DNN). For the history added approaches (i.e., UF-his and TF-his), we use the D-LSTM and D-BLSTM models and our findings show an enhanced forecasting performance in all the experiments as compared to the history-less technique (i.e., UF-cur and TF-cur). We further experiment with an added random utterance instead of the history, and the performance declined significantly, indicating the effectiveness of adding history information in emotion forecasting. The experimental results for testing $H_1$ and $H_2$ lead us to conclude that emotion forecasting can improve considerably if we forecast with a deep bidirectional dynamic model (D-BLSTM) and take a previous utterance along with the present utterance.

One of the limitations of our work is that we have not tested if our proposed models are optimized for the emotion forecasting task. Additionally, the deep learning framework can be sensitive and unstable. An extensive hyperparameter search can make the network less sensitive. Hence, investigating an optimized model with a comparatively stable framework would be our next research step. There can be several future research directions from this work. First, we focus only on the emotional flow of one speaker and do not consider the influence of the other speaker, while previous studies showed improvement of accuracy in emotion recognition, when the mutual influence was taken into...
account [50]. Hence, forecasting performance can be enhanced considerably, if the effect of other person’s emotion is incorporated in the modeling process. Second, emotion can be highly contextual, which means emotional flow can be biased in different scenarios. Therefore, forecasting result can be improved by considering the context.
References


