Automated Mental Disorders Assessment Using Machine Learning

Niloufar Abaei Koupaei

Thesis Submitted to the University of Ottawa

in Partial Fulfillment of the Requirements for the Degree of
Doctorate in Philosophy Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Mental and behavioural disorders such as bipolar disorder and depression are critical healthcare issues that affected approximately 45 and 264 million people around the world, respectively in 2020. Early detection and intervention are crucial for limiting the negative effects that these illnesses can have on people’s lives.

Although the symptoms for different mental disorders vary, they generally are characterized by a combination of abnormal behaviours, thoughts, and emotions. Mental disorders can affect one’s ability to relate to others and function every day. To assess symptoms, clinicians often use structured clinical interviews and standard questioners. However, there is a scarcity of automated or technology-assisted tools that can simplify the diagnostic process.

The main objective of this thesis is to investigate, develop, and propose automated methods for mental disorder detection. We focus in our research on bipolar disorder and depression as they are two of the most common and debilitating mental illnesses.

Bipolar disorder is one of the most prevalent mental illnesses in the world. Its principal indicator is the extreme swings in the mood ranging from the manic to depressive states. We propose automatic ternary classification models for the bipolar disorder manic states. We employ a dataset that uses the Young Mania Recall Scale to distinguish the manic states of patients as: Mania, Hypo-Mania, and Remission. The dataset comprises audio-visual recordings of bipolar disorder patients undergoing a structured interview.

We propose three bipolar disorder classification solutions. The first approach uses a hybrid LSTM-CNN model. We apply a CNN model to extract facial features from video signals. We supply the features’ sequence to an LSTM model to resolve the bipolar disorder state. Our solution achieved promising results on the development and test set of the Turkish Audio-Visual Bipolar Disorder Corpus with the Unweighted Average Recall of 60.67% and 57.4%, respectively.

The second solution employs additional features from the structured interview recordings. We acquire visual representations along with audio and textual cues. We capture Mel-Frequency Cepstral Coefficients and Geneva Minimalistic Acoustic Parameter Set as audio features. We compute linguistic and sentiment features for each subject’s transcript. We present a stacked ensemble classifier to classify all fused features after feature selection. A set of three homogeneous CNNs and an MLP constitute the first and second levels of the stacked ensemble classifier respectively. Moreover, we use reinforcement learning to optimize the networks and their hyperparameters. We show that our stacked ensemble solution outperforms existing models on the Turkish Audio-Visual Bipolar Disorder corpus with a 59.3% unweighted average unit on the test set. To the best of our knowledge, this is the highest performance achieved on this dataset.

The Turkish Audio-Visual Bipolar Disorder dataset comprises a relatively small number of videos. Moreover, the labels for the testing set are kept confidential by the dataset provider. Hence, this motivated us to train a classifier using a semi-supervised ladder network for the third solution. This network benefits from unlabeled data during training. Our goal was to investigate whether a bipolar disorder states classifier can be trained using a mix of labelled and unlabelled data. This would alleviate the burden of labelling all the videos in the training set. We collect informative audio, visual, and textual features from the recordings to realize a multi-model classifier of the manic states. The third proposed model achieved a 53.7% and 60.0% unweighted average unit on the test and development sets, respectively.

There is a growing demand for automated depression detection system to control the subjective bias in diagnosis. We propose an automated depression severity detection model that uses multi-modal fusion of audio and textual information. We train the model on the E-DAIC corpus, which labels the individual’s depression level with patient health questionnaire score. We use MFCCs and eGeMAPs as audio representations and Word2Vec embeddings for the textual modality. Then, we implement a stacked ensemble regressor to detect depression severity. The proposed model achieves a concordance correlation coefficient 0.49 on the test set. To the best of our knowledge, this is the highest performing model on this dataset.

Acknowledgments

I express my deepest gratitude and appreciation to my supervisor Dr. Hussein Al Osman, who is exceptionally nice human being and respectful person. He supported me throughout this endeavour with enlightening discussions and made administrative processes simpler to follow. I especially appreciate his belief and trust as I undertook research on the development of automated methods for mental disorders assessment. He was extremely patient and responsive throughout my research. It was a huge honor for me working with him and learning from him.

Nobody has been more important to me in the pursuit of this project than the members of my family. I would like to thank my mom, whose love and support motivated me to follow my dreams. Without her dedication, I could never be where I am now. She is the ultimate role model. Most importantly, I wish to thank my loving and supportive husband, Mohsen, who was always there for me in times of difficulty and frustration and eased the rough path of PhD.

Dedication

To my mom and husband

Table of Contents

List of Figures

List of Tables

Glossary of Terms

AE	Autoencoder
AI	Artificial Intelligence
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
AUC	Area Under the Curve
AVEC	Audio/Visual Emotion Challenge
BD	Bipolar Disorder
BERT	Bidirectional Encoder Representations from Transformers
BoAW	Bags-of-audio-words
BoVW	Bag-of-video-words
CapsNet	Capsule Neural Network
CBOW	Continuous Bag of Words
CCC	Concordance Correlation Coefficient
CLA	Custom List Analyzer
CNN	Convolutional Neural Network
CRAT	Constructed Response Analysis Tool
DAIC	Distress Analysis Interview Corpus
DAIC-WOZ	Distress Analysis Interview Corpus – Wizard of Oz
DBN	Deep Belief Network
DNN	Deep Neural Network
DSP	Digital Signal Processing
ECG	Electrocardiogram
E-DAIC	Extended Distress Analysis Interview Corpus
GBDT	Gradient Boosted Decision Tree
GCNN	Gated Convolutional Neural Network
GeMAPS	Geneva Minimalistic Acoustic Parameter Set
GEWELMs	Greedy Ensembles of Weighted Extreme Learning Machines
HCI	Human Computer Interaction
HDR	Histogram of Displacement Range
LBP	Local Binary Pattern
LLD	Low-Level Descriptors
LSTM	Long-Short-Term Memory
MFCCs	Mel-Frequency Cepstral Coefficients

ML	Machine Learning
MLP	Multi-Layer Perceptron
MRI	Magnetic Resonance Imaging
Multi-DDAE	Multimodal Deep Denoising Autoencoder
NLP	Natural language processing
PIORI	Predicting Individual Outcomes for Rapid Intervention
RBM	Restricted Boltzmann Machine
RF	Random Forest
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SALAT	Suite of Linguistic Analysis Tools
SÉANCE	Sentiment Analysis and Cognition Engines
SiNLP	Simple Natural Language Processing Tool
SRM	Social Rhythm Metric
SVM	Support Vector Machine
TAACO	Tool for the Automatic Analysis of Cohesion
TAALED	Tool for the Automatic Analysis of Lexical Diversity
TAALES	Tool for the Automatic Analysis of Lexical Sophistication
TAASSC	Tool for the Automatic Analysis of Syntactic Sophistication and Complexity
TAVBD	Turkish Audio-Visual Bipolar Disorder Corpus
UAR	Unweighted Average Recall
VGG	Visual Geometry Group
YMRS	Young Mania Rating Scale

Chapter 1 Introduction

There has been a 13% rise in mental disorders or neurological disorders in the past decade [1]. Mental disorders are typically characterized by developmental issues which have substantial effects on all areas of life, including school or work performance, relationships with family and friends, and ability to participate in the community. The range of developmental issues may vary for different kinds of disorders [1] .

According to the World Health Organisation report [1], the number of individuals with mental disorders is increasing worldwide. The report indicated that approximately 20% of children and adolescents have mental disorders, which can sometimes lead to suicide. The diagnosis and treatment of mental disorders mainly relies on clinical judgment and patient's self-reports [2]. Therefore, diagnosis is a challenging task especially when a patient's cognition is impaired.

Bipolar disorder (BD) is globally one of the most prevalent mental disorders. BD, previously called manic depression, is a serious mental disorder characterized by the severe swings in the mood ranging from mania to depression. Both manic and depressive episodes are separated by normal mood intervals. Clinicians often classify the patient's transient mental condition into several states. Three of the manic states recognized by clinicians are: Mania (high arousal), Hypo-Mania (less severe than Mania) and Remission. When an individual is experiencing a depressive mood, he/she may feel hopeless and lose interest in many kinds of everyday activities. When the mood shifts to Mania or Hypo-Mania, the patient may feel euphoric, full of energy or unusually irritable. These mood swings can affect sleep, energy, activity, judgment, behavior, and the ability to think clearly. Therefore, BD can continuously impair an individual’s wellbeing and ability to work.

BD is a life-long persistent condition where patients often experience episodes of improvements and setbacks [3]. Getting an accurate diagnosis is the first step towards treatment. Therefore, clinical diagnosis of BD often depends on the psychiatrist’s experience and knowledge, and may in some cases lack objectivity [4]. The BD treatment poses a serious challenge due to the potential unpredictable clinical symptoms (emotional ups and downs), duration of the mood episodes, and the heterogenous nature of clinical symptoms (Mania, Hypo-Mania, depressive and mixed) [3].

Major depressive disorder (MDD) is another serious mental illness, which based on a report in 2020, affects around 264 million people worldwide [5]. Individuals facing adversity, such as unemployment and psychological trauma, are more prone to develop depression. Depression itself may add more stress to an individual’s life and even lead one to commit suicide. MDD can change the way people think, feel, and perform daily activities such as eating, sleeping, and working [6].

Recent evidence shows that nearly two- thirds of people with mental disorders never seek help from health professionals. However, most patients seeking help receive treatment from primary care providers as opposed to mental health specialists which in some cases are difficult to access. Nonetheless, primary care providers have shown limited capacity for treatment [7]. Hence, there is a significant need for developing innovative methods to optimize care while maximizing access [8].

The shortage in clinicians in some parts of the world and stigma associated with the disorder in some contexts, highlights the need for automated mental disorder detection mechanisms.

Automated assessment might contribute to the early detection of symptoms and inform our understanding of biological markers for diagnosis. Automated detection targets indicators that belong to several modalities including: verbal (audio features) and nonverbal (facial expressions, body gestures, etc.) [9]. Hence, continuous monitoring of patients through computer interaction technologies has been increasingly receiving attention in recent years.

Machine learning algorithms can tackle expressive behaviours that are related to BD detection. As an individual may experience different emotions in each BD state, emotion detection using automated approaches has generated promising results in BD detection [10]. Recent studies illustrated that extracting speech activities and changes in speech enables us to assess symptoms severity of BD [11], [12]. In addition to verbal cues, capturing nonverbal representations, such as gestures and expressions, are indicative of BD [13].

Similar to BD, depression detection has taken advantage of automated assessment. In recent years, automatic depression screening from different verbal and non-verbal signals has been considerably investigated [14], [15]. Facial and audio features provide meaningful information in distinguishing depressed subjects and predicting the level of depression [16]. Researchers in [17] investigated obvious differences between depressed and non-depressed subjects with respect to body movements, speech, and reaction time. These differences are also supported by other studies, such as [18] and [19], which investigated how depression affects people’s emotions.

1.1 Problem Statement

Traditional methods for monitoring mental disorders rely on reports from either patients or clinicians [20]. These subjective reports are structured in the form of multiple-choice style questionnaires. Clinicians may administer these questionnaires while interviewing patients. The

Hamilton Rating Scale for Depression (HRSD) [21] and Young Mania Rating Scale (YMRS) [22] are two clinician-administered questionnaires indicating severity of depression and BD, respectively. Self-administered questionnaires are done by patients without the clinician's involvement. One example of self-administered questionnaires is the Patient Health Questionnaire (PHQ-8) [23], which indicates depression.

The main drawback of conventional clinician and self-administered methods is their inconsistency and vulnerability to bias. The reliability of self-administered questionnaires is highly dependent on the patient's honesty in providing information about their mental and behavioural situation. Studies have shown that for some participants, the self-administered questionnaire and clinical interview render conflicting results [24] . Furthermore, the variations in the training and experience of clinicians may result in inconsistencies in their diagnosis when they apply clinician-administered questionnaires [25].

Given the mentioned shortcomings, we propose an automated mental illness assessment framework. The framework can complement traditional assessment approaches and provide experimental feedback to clinicians and researchers.

1.2 Motivation

Although conventional assessments are highly dependent on the clinicians' experience and honesty of patients while reporting their behavioural status, automated computational assessments can be done remotely and consider more behavioral indicators [26]. In this thesis, we propose computer-based automated mental health detection models. These models use information recorded through a camera to classify a subject's mental health state. The advantage of automated monitoring is not only limited to its scalability but also to its capability of processing different modalities of information to increase the reliability of the detection. Hence, employing multiple modalities of information enables the system to extract more complex features and find new structures in the data.

An automated monitoring system refers to a computational framework that analyzes information that reflect human behaviour by capturing various social signals, such as speech, facial expression, and body movement. The produced results by automated systems are not susceptible to the expertise of clinicians in the diagnosis process. In addition, achieved results from an autonomous model may be reproducible. Thus, they may be later replicated for further investigation of the mental state or improvement of the assessment method.

Recently, researchers have introduced several mental disorder detection methods [27],[28], [29], [30]. For traditional diagnostic methods, doctors typically conduct consecutive face-to-face sessions with patients to assess their condition. Due to the discussed flaws of traditional methods, technological solutions can enable more reliable detection. For instance, BD and depression may be assessed through computer-based tools that do not necessitate the presence of an experienced clinician. However, more realistically, automated methods for mental disorder detection can provide experimental feedback to clinicians and research in psychiatry [9].

Automated BD detection is a new field with a very limited number of studies on the only available dataset [31], which depicts structured audio/visual recordings of interviews with BD patients. Existing automated BD detection models have shown to be promising although still far from reliable. Conversely, automated depression detection methods have received increased attention in recent years. However, there is still a potential to improve existing results. To optimize the performance of automated assessment methods, we must extract relevant and informative feature sets. Therefore, this thesis presents various automated multi-modal classification and regression models for both BD and depression using machine learning algorithms.

1.3 Contributions

In this section, we summarize the contributions of this study for both BD and depression severity detection. We develop three models for BD detection and one model for depression severity detection.

1.3.1 Automated BD severity detection

1) **Stacked ensemble model:**

    - Development of a stacked ensemble classifier: Ensemble models are one of the robust options for classifier. Instead of using a deep neural network, we define a meta learner to combine the predicted output of several shallow convolutional classifiers. The defined ensemble model allows us to achieve more accurate detection.
    - Application of a reinforcement learning algorithm to tune classifier hyper-parameters: The performance of classifiers is highly dependent to the hyper-parameters. Hyper-parameter tuning has been an active area of research in recent years. We apply a reinforcement learning algorithm to find the best architecture for the proposed stacked ensemble model, which enhances the accuracy of the classification.

2) **CNN-LSTM model:**

    - Development of a hybrid classifier including a CNN and a Long-Short-Term Memory (LSTM): As the dataset includes audio-visual recordings, we are able to capture a sequence of frames for each patient's recording files. Having video recordings allows us to employ a classifier which takes the relation between sequences into consideration.

Hence, LSTM is a fitting option. Therefore, we propose a hybrid model including a CNN to extract visual features and an LSTM classifier to detect the bipolar status.

3) **Semi-supervised model:**

- Development of a semi-supervised classifier: Giving the fact that existing BD dataset is small motivated us to implement a ladder network, which contains both supervised (using labeled data) and unsupervised (using unlabeled data) learning. The key advantage of developing a semi-supervised model is removing the barrier of manual annotation of training data, which is an expensive and time-consuming task.

This thesis has resulted the publication of three papers:

- Journal paper: **N. Abaei**, H. Al Osman " A Multimodal Ensemble Classifier for Bipolar Disorder Detection", accepted on IEEE Transaction on Affective Computing, 2020.
- Conference paper: **N. Abaei**, H. Al Osman "A Hybrid Model for Bipolar Disorder Classification from Visual Information", accepted on IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.
- Conference paper, **N. Abaei**, H. Al Osman " A Semi-supervised Classifier for Bipolar Disorder Detection using audio, visual, and text modality ", accepted to 22nd International Conference on Intelligent Data Engineering and Automated Learning (IDEAL), 2021.

1.3.2 Automated depression severity detection

1) **Stacked ensemble model:**

- Development a stacked ensemble: As the proposed stacked ensemble model performs well on BD dataset, we motivated to test the model on a depression dataset. However, we modify the architecture of the proposed model for BD and develop a stacked ensemble regression to detect the depression severity.

Our study on depression detection model resulted in the submission of one journal paper:

- **N. Abaei Koupaei**, H. Al Osman " An Automated Multimodal Classifier for Depression Detection", submitted to on IEEE Transaction on Affective Computing, 2021.

Chapter 2 Background and Related Works

In the 1930s, 1940s and early 1950s, a group of scientists from a variety of fields (mathematics, engineering, economics, and psychology) had a desire for creating machines that think. The artificial Intelligence (AI) field was founded as an academic discipline in the 1950s [32].

Human beings inevitably try to consider all their previous experiences and attempts to continuously improve their way of solving a problem. AI mimics the way the human brain solves a problem by simulating human actions. We look to intelligent software to automate routine labor, understand speech or images, make diagnoses in medicine, and support basic scientific research.

2.1 Machine Learning

Machine Learning (ML) is a subset of AI which studies computer algorithms that improve automatically through experience [33]. The machine learning framework focuses on training computers through the acquisition and handling of new knowledge to develop cognitive skills through instruction and practice. Hence, machines can learn better by analysing more data. Conversely, traditional computer programs do not consider their past decisions to build stronger models. Today, ML approaches are prevalent in different applications as they can tackle and solve problems that are intellectually complex for humans but straightforward for computers. In the following sub sections we classify machine learning approaches and introduce various types of models have been used and researched for machine learning systems.

2.1.1 Types of Machine Learning Approaches

There are different ways of classifying ML systems [34]. Typically, we can classify them according to learning strategies, such as supervised, unsupervised, semi-supervised, and reinforcement learning [35]. In the following sections, we will briefly discuss these classes.

2.1.1.1 Supervised Learning

In supervised learning, the input-output pairs are provided for the computer system. Hence, the algorithm is tasked with uncovering a pattern between the inputs and outputs that can be modeled by a function.

The main task in supervised learning is finding a deterministic function to map any input to an output while optimally minimizing the error. Based on the type of predicted output, supervised learning can be categorized as *classification* or *regression* learning.

Classification learning is a process of categorizing the given data into one of discrete outputs where each element of the output is called a class. The learning algorithm that solves the classification problem is called classifier. Some examples of classification learning are face recognition, gender prediction and many other pattern recognition tasks.

Regression learning involves an output space with continuous values such as “salary” or “weights”. The regression learning algorithm attempts to fit the best hyper-plane that goes through data points. Predicting the house pricing or the value of shares in stock exchange market are some regression-learning examples.

2.1.1.2 Unsupervised Learning

Unlike supervised learning, in unsupervised learning, the computer system is only provided with input data without any corresponding output. As data are not labeled with output classes, the system has to utilize a set of clusters or a probability density function describing how likely it is to observe a certain object in the future. There is little human supervision with unsupervised learning. The main goal of unsupervised learning is to capture the structure of the data to learn more about it. *Clustering* and *association* are two common unsupervised learning problems.

In *clustering learning* the system seeks to discover similarity or inherit grouping in data. So, the similarity of all objects that belong to one cluster is greater than the similarity of objects in different clusters. Grouping customer purchasing behaviour is an example of a clustering problem.

In *association learning*, the system discovers rules that describe large portions of the data, such as people that buy X also tend to buy Y.

2.1.1.3 Semi-supervised Learning

Semi-supervised learning is another machine learning technique that trains an algorithm on a combination of a small amount of labeled data with a large amount of unlabeled data. This learning technique combines aspects of supervised and unsupervised learning. The basic procedure involves using, first, unsupervised algorithms to discover the structure of the input data, and second, supervised algorithms attempt to label unlabelled data by leveraging the labeled data. As labeling all data is expensive and time consuming, many real-life problems fall into semi-supervised learning. Speech analysis is a common example of semi-supervised learning.

2.1.1.4 Reinforcement Learning

Reinforcement learning is based on control theory. It simulates a dynamic environment with *state*, *action*, and *reward* as data. The reinforcement algorithm learns how to map states to actions while maximizing the reward. In contrast to supervised learning, the algorithm is not informed with the best actions to take in a given state. Instead, the system needs to find the optimum actions based on immediate achieved rewards after the action is taken. Playing chess is an example of a reinforcement learning problem. Each position on the chess board is considered as one state and the actions are possible moves.

2.1.2 Artificial Neural Networks

Artificial Neural Networks (ANNs) are computing systems that are inspired from the biological brain. Each Neural Network forms weighted connections of many *nodes/neurons* to convey signals

[35]. Each neuron receives several inputs and produces only one output by considering the weighted sum of all inputs and adding the bias to the sum:

$$a_i = \sum_{k=1}^{r} w_{ik} x_i + b_i \qquad (2.1)$$

Where a_i refers to the ith neuron, w_{ik} is the corresponding weight between the ith neuron and kth input, and b_i is the bias of ith neuron.

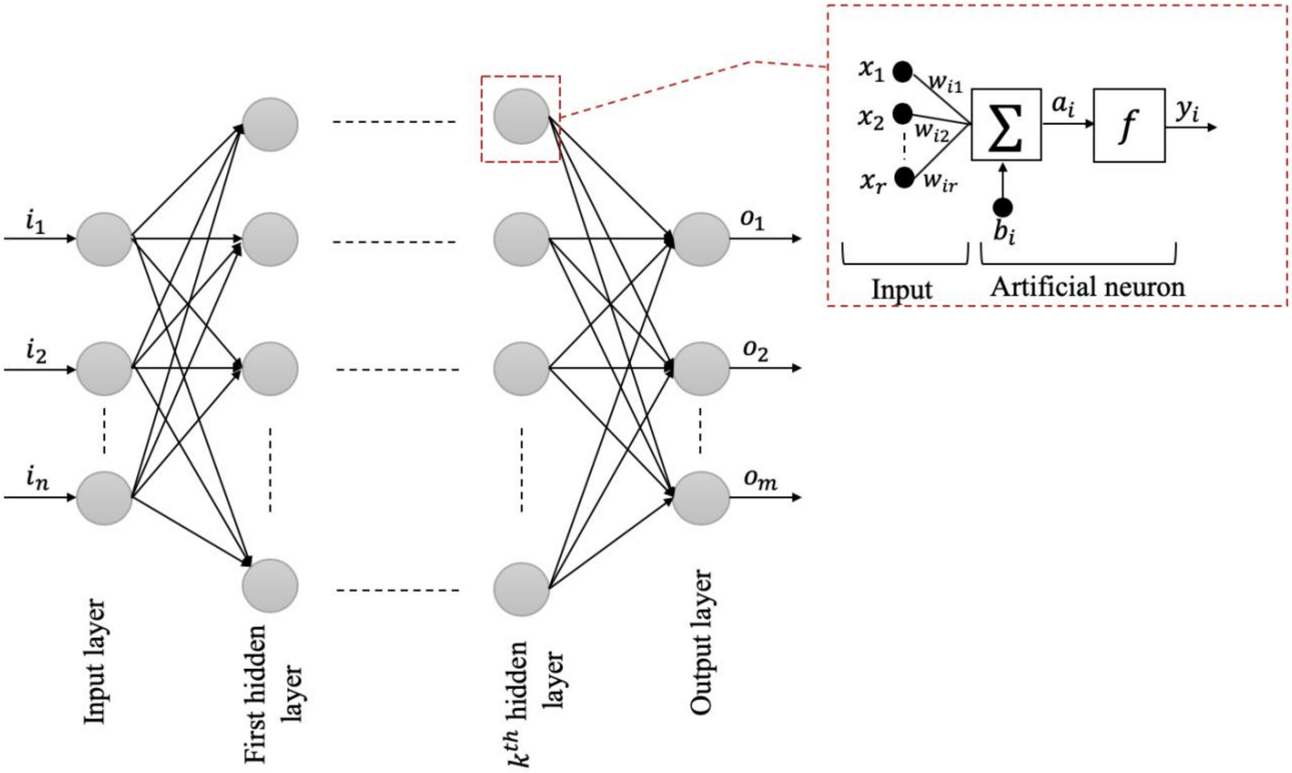


Figure 2.1 Example of an Artificial Neural Network (ANN). This figure also shows the most general form of artificial neuron.

The weighted sum is passed through an *activation* function, mostly non-linear, to produce the output. Every ANN is composed of various layers and each layer could be characterized with a different number of neurons. Figure 2.1 depicts a general form of an ANN. As the figure illustrates, the input layer receives information from external sources and the output of this layer corresponds to the input of the first hidden layer. Hence, the network connections deliver the output of one neuron as an input of another. Each neuron can have multiple input and output connections. The assigned weight for each connection shows the importance of that connection and can be adapted during the learning process.

The learning process consists of adapting connection weights based on given observations. The weight adjustment enables the model to better handle a task while reaching higher accuracy. Once considering more observations does not reduce the error rate, learning is complete. The learning rate is calculated on a defined *cost function* that is evaluated periodically during the learning. *Backpropagation* is a widely used method to readjust weight connections backwards through the network by calculating the gradient of the error function. Hence, once a prediction based on given inputs is produced, the difference between the actual output and the predicted output indicates the error. Then, the measured error is utilized to adjust connection weights starting from the final layer and moving back to the first layer. The backward flow of the error results in a more efficient computation of the gradient at each layer.

Besides choosing a learning algorithm for ANNs, there are several constant parameters that must be initialized before the learning process. These parameters include the learning rate, number of hidden layers, number of neurons in each hidden layer, batch size among others. Tuning hyper-parameters has a great impact on model accuracy. Therefore, the values of hyper-parameters are derived via learning.

2.1.3 Deep Neural Network

A Deep Neural Network (DNN) is a large neural network consisting of a hierarchical architecture with many hidden layers that is capable of solving more complex problems and extracting high-level features from the input [35]. Explicitly, an ANN that is made of more than three layers, an input layer, a hidden layer and an output layer, is typically called a DNN, although no exact definition exists.

Deep neural networks learn from data while passing through different layers. Given enough labeled training datasets and suitable models, deep learning approaches can help humans establish mapping functions for operation convenience [35]. There are several deep learning architectures as we detail below:

- **Restricted Boltzmann Machine (RBM):** RBMs are generative stochastic ANN with two shallow layers [36]. The first layer of the RBM is called visible or input layer and the second layer is called hidden layer. Each visible unit is connected to all the hidden units, this connection is undirected, so each hidden unit is also connected to all the visible units. RBM is an algorithm useful for dimensionality reduction, classification, regression, collaborative filtering, feature learning and topic modeling. Although RBMs are occasionally used, most people in the deep-learning community have started replacing their use with General Adversarial Networks or Variational Autoencoders.
- **Deep Belief Network (DBN):** The network is a stack of RBMs. Different layers of RBMs in a DBN are trained sequentially: the lower RBMs are trained first, then the higher ones [37]. DBNs have an efficient performance on unlabeled data.
- **Autoencoder (AE):** AEs contains two components including an encoder to map input data to the code and a decoder to reconstruct input data from the code. The main idea is coding input data to some representations that enables the model to

reconstruct the input data using the same extracted representations. AEs are widely used for the purpose of dimension reduction and compression tasks [38].

- **Convolutional Neural Network (CNN):** CNNs consist of input, output, and multiple hidden layers. As the name implies, CNNs utilize convolutional operations instead of general matrix multiplication for each layer. This is the most common topology for DNNs [39]. We will discuss it more in Section 2.1.4.
- **Recurrent Neural Network (RNN):** RNNs allow previous outputs to be used as the inputs while maintaining hidden states. We will provide more details on RNNs in Section 2.1.5.

2.1.4 Convolutional Neural Network (CNN)

CNNs are a type of deep neural networks that are made up of neurons that have learnable weights and biases. Each neuron in the hidden layers receives some inputs, from the previous layer, and performs convolutions.

CNNs assume that the input data is in the form of an image, which allows us to consider certain properties for the architecture. A CNN successfully captures the spatial and temporal features in an image using relevant filters. Then, CNNs can be trained to better understand the sophistication of the image. A CNN consists of three main layers:

- Convolutional (Conv) layers: The Conv layer is the key block in CNN architectures and operates all the heavy computations. Each Conv layer contains small learnable filters which move through the depth of the input image. A 2D activation map is produced while we slide each filter across the width and height volume of the input and compute the dot product between the filter and input. The stacked form of the activation maps, which are produced by each filter, corresponds to the output volume.

- Pooling layers: Typically, pooling layers are located in-between consecutive convolutional layers to reduce the spatial size of convolved features. There are two common types of pooling layers: Max pooling and Average pooling. The former one selects the maximum value from the portion of the image that is covered by the kernel, while the latter returns the average of all values from the portion of the image that is covered by the kernel.
- Fully connected layers: The convolutional layer and the pooling layer together form the i-th layer of the architecture. After building several such layers, depending on the complexity of the input, we need to flatten the captured features and feed them to the final classifier. This flattening process is done typically using fully connected layers.

In this study we will use CNN models to extract features from the input data. The features are then forward to a classification or regression model for further processing.

2.1.5 Recurrent Neural Network (RNN)

Figure 2.2 compares the architecture of RNNs [40] with feed-forward networks. As the figure depicts, in RNNs, networks neuron’s input may come from other neurons in the same hidden layer.

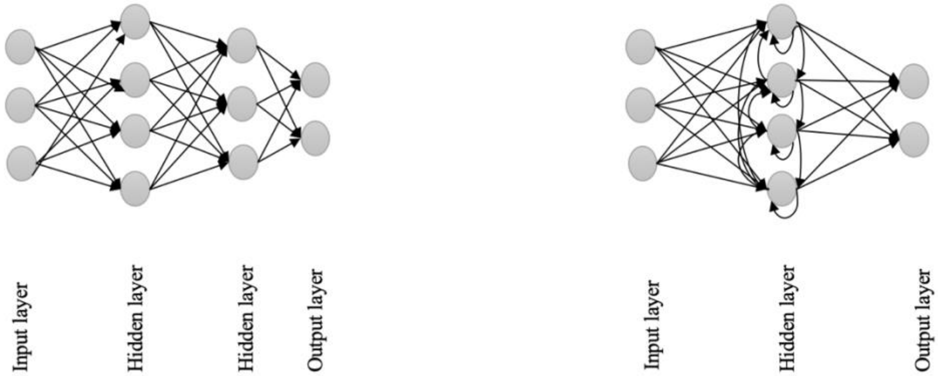


Figure 2.2 Feed-forward network (left) vs RNN network (right). RNNs allow cyclical connections.

However, in feed-forward network each neuron accepts input from neurons in the previous hidden layer.

Since RNNs provide connections between different neurons of the same hidden layer, these networks are able to accept a sequence of input data where the meaning of the data depends on the “context”. As Figure 2.3 illustrates, for a given sequence of data, $x_1, x_2, \ldots, x_n$, the output of the node with input x_k is h_k :

$$h_k = f_\theta(h_{k-1}, x_k) \qquad (2.2)$$

Where f_θ has the same parameters for all x_k and h_k.

Given the RNN’s ability to model sequential data, we will explore this architecture for our mental illness assessment tasks.

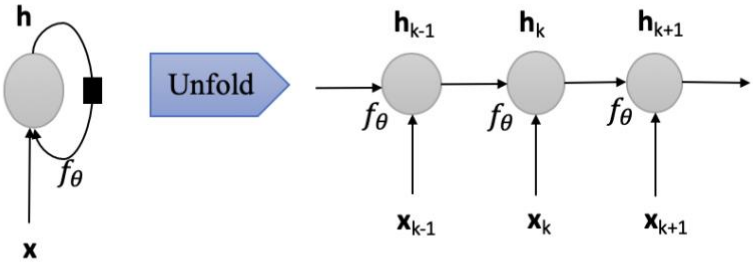


Figure 2.3 A standard RNN neuron. Weights are shared in sequential data.

2.1.6 Ensemble Neural Networks

Ensemble Neural network is a model that uses several jointly connected neural networks to solve a problem. Ensemble models combine the predicted output of different individual classifiers, where each is trained separately, to achieve a more reliable prediction.

In stochastic methods, the outputs have some degree of uncertainty. The stochastic model refers to any model involving some level of randomness and uncertainty. Many machine learning algorithms and models are trained based on stochastic learning method, as they make use of randomness during optimization and learning. Having this randomness makes the models more sensitive to their initial weights and hyper-parameters. Hence, every time we train a neural network, with a stochastic learning algorithm, it may achieve a different version of outputs because of the existing randomness.

Ensemble models were produced to overcome the high variance issue of a single neural network by decreasing the variance and the generalized error [41]. Combining the predictions from different models adds a bias which address the high variance of each neural network in the model.

There are three main features in designing an ensemble model that must be adapted to each application:

- **Training Data:** Each model in an ensemble model is trained on training data. The training data can be divided into separate sets. Then each set is used to train one of the networks in an ensemble model. However, such approach requires a large training dataset. To overcome the issue of limited training data, which is common in realistic use-cases, resampling the main training dataset typically produces promising results.
- **Choice of Sub Networks:** Neural networks in an ensemble model are either identical or different. Training the same neural networks with various initial conditions results in different models with varying performance. However, an ensemble model consisting of diverse configurations is more robust to reduce the generalization error. The choice of networks is highly dependent on the application and data structure.
- **Combinations of Sub Networks:** A straightforward linear approach to combine ensemble members is calculating the average of the predictions from all the networks. Since some members perform better than others, considering a weighted average approach gives a greater weight to those members [42]. The choice of combination strategy is not limited to linear approaches, a further complex way is defining a model to learn the best combination method of predictions from sub networks.

Given the ensemble models proven potential, we will investigate their suitability for solving the problem of mental illness assessment.

2.1.7 Machine Learning Assessment

There are several assessment performance metrics for both regression and classification problems. Hence, we list the most common classification and regression metrics as follow:

Classification

- **Confusion Matrix:** It is a table layout that captures the performance of an algorithm. As the name implies, it makes it easy to see if the system is confusing two classes. Each row of matrix shows an actual class, and each column represents the predicted class.
- **Precision:** The number of true positives (i.e., number of correctly labelled items as belonging to the positive class) divided by all positive results (i.e., sum of true positive and false negatives, which are items incorrectly labelled to the class).
- **Recall:** The number of true positives divided by total number of items that belong to the positive class (i.e., sum of true positives and false negatives, which are items that were not classified to the positive class but should have been).
- **F1-Score:** It is a harmonic mean of precision and recall score:

$$\boldsymbol{F_1 - Score = 2 \times \frac{precision \times recall}{precision + recall}}$$

High F1-Score means both precision and recall are high, which is desirable for classifiers.

Regression

- **Mean Squared Error (MSE):** MSE is one of the most prevalent metrics for regression tasks. It is simply the average of the squared difference between the target value and the value predicted by the regression model. As it squares the differences, it penalizes even a small error which leads to over-estimation of is the model’s performance shortcoming.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad (2.3)$$

Where y and $\hat{y}$ are target value and predicted value, respectively.

- **Root Mean Squared Error (RMSE):** It is the square root of the averaged squared difference between the target value and the value predicted by the model. It is preferred in some cases as the errors are first squared before averaging which poses a high penalty on large errors. This implies that RMSE is useful when large errors are undesired.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \qquad (2.4)$$

  Where y and $\hat{y}$ are target value and predicted value, respectively.

- **Mean Absolute Error (MAE):** MAE is the absolute difference between the target value and the value predicted by the model. It is a linear score which means all the individual differences are weighted equally.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad (2.5)$$

  Where y and $\hat{y}$ are target value and predicted value, respectively.

In addition to choosing a metric capable of measuring the effectiveness of the classifier, it is required to consider the standard benchmarks. Having a common metric for a specific problem, enables researchers to reliably compare the results of various systems.

2.2 Machine Learning and Applications in Mental Healthcare

The rapid developments of technology including smartphones, wearable devices, and neuroimaging has allowed researchers to capture a vast variety of data in psychology and mental health [43]. Conventional statistics are ineffective in analysing many types of clinical data due to the data's complexity and size. Hence, Machine Learning (ML) has emerged as a robust solution to tackle this issue. As we discussed in Section 2.1, ML takes advantage of some statistical techniques to automatically learn from data. Therefore, automatic improvements through experience enables ML to reliably capture existing patterns in the data and results in more accurate predictions. Based on a scoping review on ML in healthcare [44], applying ML on mental health data has led to improving patient outcomes and providing better insights on clinical and psychological conditions. Availability of automated assessment mechanisms in mental health offers clinicians powerful tools for diagnosing and monitoring mental disorders during the treatment period.

Many researchers have investigated applying ML algorithms for early diagnosis or monitoring of patients' health conditions. The authors of [45] emphasized the efficiency of utilizing wearable sensors and phones to obtain several general wellbeing factors, such as sleep quality and stress level, which helps in the assessment of the mental health status of the user. Alam et al. equipped homes with lightweight biosensors to capture emergency psychiatric symptoms [46]. All psychiatric states were processed and modeled through a set of ML techniques.

Using ML methods in healthcare is not limited to applications involving wearable sensor devices. The diagnosis of Alzheimer's disease using ML with neuroimaging (such as Magnetic Resonance Imaging (MRI)) and speech patterns in audio signals has been explored by several studies [44], [47], [48]. Automated depression detection has been another potential field of interest for researchers in recent years. Supervised ML techniques have been applied on speech signals, unstructured texts aggregated from social media, and video recordings to implement early diagnosis or distinguish between different levels of depression [49], [50]. The visual and audio modalities in particular have been utilized for the automatic detection of various mental health issues.

Deep learning feature extraction approaches have shown more efficiency than hand-crafted features on various tasks, including image classification [51] , speech recognition [52], etc. Convolutional Neural Networks (CNNs), a type of ANNs, are inspired by the brain visual cortex functioning and have been widely applied to image recognition problems. They have a capacity to learn spatial features in images, an important characteristic for assessing facial expressions. However, they are unable to capture temporal information. Since temporal properties present useful information about mental disorders, CNNs can be cascaded with other networks capable of modeling temporal dependencies, such as Recurrent Neural Networks (RNNs), and in particular LSTMs. There are several studies that used a combination of CNN and LSTM for different applications.

In [53], researchers proposed a spatial-temporal feature learning approach to model facial expression recognition. In the first step, all facial expression characteristics are learned by a CNN model from spatial images. Then, an LSTM model learns the temporal characteristics of extracted spatial features in the first step. Experimental results showed the efficiency of obtaining temporal features alongside with spatial ones. Another study on facial expression recognition, presented a hybrid CNN-LSTM model that uses facial action units (FAUs) as input [54]. Authors used a CNN to extract spatial features for each frame, which enabled them to reduce person-specific biases caused by hand-crafted descriptors (e.g., HOG and Gabor). Moreover, an LSTM network was stacked on top of the CNN regardless of the input length.

Huang et al. presented an attention-based CNN and LSTM model to detect mood disorders based on individuals' speech signals [55]. The dataset contains interview speech signals of participants who viewed six emotional video recordings. A CNN model with an attention mechanism generated an emotional profile of each speech signal. Then, an LSTM network captured the temporal characteristics of the emotions. The proposed hybrid model outperformed not only CNN but also support vector machine predictions. The implementation of hybrid models, CNN-LSTM, has proved to be effective for depression detection [56]. Ma et al. [56] used the DepAudioNet to extract depression characteristics in speech signals from the Distress Analysis Interview Corpus – Wizard of Oz (DAIC-WOZ) dataset [57] and an LSTM to classify the extracted descriptors.

Ensemble learning has achieved a great success recently in various real-world applications [58], such as object detection and tracking, image recognition, mental health disorder detection, bioinformatics, data mining, etc. In this section, we briefly outline some existing studies which employed different methods of ensemble learning for various machine learning applications.

The authors of [59] proposed an ensemble model on visual and textual features to classify social image emotions. Their objective was classifying eight image emotions including: amusement, awe, contentment, excitement, anger, disgust, fear and sadness. Then, they adapted an ensemble classifier, which outperformed conventional classifiers in the image emotion classification task. In the domain of using sensors for human emotion classification, ensemble classifiers proved notable capabilities in detecting four major human emotions : anger; sadness; joy; and pleasure integrating electrocardiogram (ECG) signals [60]. Authors in [60] defined the feature extraction module based on four different ECG signals techniques including: heart rate variability; empirical mode decomposition; with-in beat analysis; and frequency spectrum analysis. The study’s evolutions showed the superiority of the proposed ensemble learner in compare to the best performing single biosensor based models in literature.

Researchers in the field of automatic depression detection have taken advantage of ensemble models. Jiang et al. introduced an ensemble logistic regression model to detect depression of multiple speech features [61]. The data set includes 170 native Mandarin speaking subjects. Half of the subjects were healthy and the other half had depression. The reported results emphasized the superiority of ensemble models compared to other classifiers. Several studies have shown that many people tend to explicitly or implicitly convey information about their mental condition in their posts on social media [62], [63]. Therefore, processing messages and posts that users share online could help for early detection of depression [64]. Hence, Saxena and Verbeek [64] used an ensemble model to improve depression detection through sentiment and behavioral analysis of social media data.

Syed et al. [65] captured the impulsive changes in audio features using turbulence features for BD detection. They applied a Fisher Vector encoding of LLDs and proved that these features are capable of capturing BD markers from speech. They utilized Greedy Ensembles of Weighted Extreme Learning Machines (GEWELMs) for classification. Another study on the Turkish Audio-Visual Bipolar Disorder (TAVBD) corpus [31], segmented each speech file to multiple chunks where each chunk is weakly labeled [66]. Then, they proposed a bagging ensemble model to compensate for the weak labelling.

2.2.1 Ensemble Approaches in Mental Healthcare

Ensemble models have proved promising performance in accurate diagnosis, especially when there are multiple features. Kumar et al. proposed an ensemble model for anxious depression predictions in real-time tweets [67]. They extracted feature sets of 5-tuples (word, timing, frequency, sentiment, contrast). The inconsistencies in posting behaviour and irregularities were explored through analyzing the opinion polarity analytics and time/frequency of tweets, respectively. The proposed model was trained using multinomial Naïve Bayes, gradient boosting, and random forest. Then an ensemble voting classifier reached accuracy of 85.09%. Hassan et al. investigated the efficiency of an ensemble model for predicting individuals' depression by extracting their emotions from text files on different social media platforms [68]. Authors proposed an ensemble voting classification and regression on top of three classifiers including: Naïve Bayes, SVM, and Maximum Entropy. Based on the reported assessment, Naïve Bayes, SVM, and Maximum Entropy achieved accuracy of 83%, 91%, and 80%, respectively.

Al-jabery et al. researched diagnosis of autism spectrum disorder based on social communication deficits and repetitive behaviours [69]. Authors proposed an ensemble model for analyzing autism spectrum disorder based on a combination of machine learning techniques and a subspace clustering algorithm. The proposed ensemble model consists of five stages of statistical and machine learning approaches to capture a subspace clustering of autism spectrum disorder data. The reported results illustrated efficiency of the proposed model. In another study, Naghavi et al. conducted an experiment to diagnose the suicide behavior among 573 university students in Middle East and North Africa [70] through responding to several questionaries. After implementing feature selection, they applied a stacked ensemble decision tree classifier, which achieved the area under the curve (AUC) of 0.90.

2.2.2 Automated BD Detection

Since maintaining stability in daily routine can significantly reduce the risk of relapse for individuals with BD, being able to automatically assess stability without requiring active user engagement can have considerable impact on clinical care. In particular, automated detection could help overcome issues with existing paper-and-pencil based clinical tools by significantly lowering the user burden of manual tracking.

Access to rich datasets for automated assessment is one of the main challenges in this field. Some studies collected data from smartphones of individuals with BD for developing and testing their automated assessment system. However, recent studies relied on datasets depicting face-to-face interviews.

In this chapter, we provide details on both streams of research. Then we discuss the target dataset we adopt in this thesis.

2.2.2.1 Smartphone-Based Dataset

Abdullah et al. introduced an automatic smartphone sensing assessment of the Social Rhythm Metric (SRM), a clinically-approved indicator of stability and rhythmicity of BD patients [71]. Seven patients with BD used smartphones that passively collected sensor data, location, and communication information to infer behavioral (from speech, activity, SMS, and call log) and contextual (from location) patterns. All information was gathered in 4 weeks. Participants also completed SRM entries using a smartphone app. The results of this study demonstrated that automated sensing can be utilized to infer the SRM score. The study argued that the generalized model consisting of a Support Vector Machine (SVM) predicts the SRM score (0-7) with a root-mean-square error of 1.40. The reasonable performance of the model stressed the feasibility of automatic smartphone-based sensing methods to monitor the status of individuals with BD.

Although some studies utilized multiple modalities [71], other studies focused on only one informative modality. For instance, Faurholt-Jepsen et al. [72] studied the detection of manic and depressive episodes by models that use the speech modality or the combination of speech modality and behavioral data (for example, number of text messages and phone calls per day). The 28 patients were recruited from The Copenhagen Clinic for Affective Disorders, Psychiatric Center Copenhagen, Denmark. The patients were instructed to use the smartphone as they primary phone for their usual communicative purposes, and to carry it with them during the day as much as possible. Using smartphones, voice features, automatically generated objective smartphone data on behavioral activities and electronic self-monitored data were collected from outpatients with BD disorder in naturalistic settings on a daily basis during a period of 12 weeks. The self-monitored data was used as ground truth to train the ML models. To capture the self-monitored data, an app was installed on the mobile phones and an alarm once a day reminded the subjects to fill the electronic self-monitored data. The self-monitored questions included information on five parameters such as: mood (scored from depressive to manic on a scale from −3 to +3, including scores of +0.5 and − 0.5); sleep length (number of hours slept/night measured in half hours intervals); medication taken (yes/ no); medication taken with changes (yes/no); activity level (scored on a scale from − 3 to +3); alcohol consumption (number of units per day); mixed mood (yes/no); irritability (yes/no); cognitive problems (yes/no); stress level (scored on a scale from 0–2); and indication of the presence of individualized early warning signs (yes/no). They used Random Forest algorithms to implement a classification task, where the obtained BD symptoms were assessed based on the Young Mania Rating Scale (YMRS) [22]. The results of this study emphasized that voice features are accurate and sensitive for classification of BD with an AUC = 0.89. Combination of voice features and automatically generated objective smartphone data on behavioral activities and electronic self-monitored data slightly increased the classification accuracy. Therefore, accurate classification of affective states based exclusively on voice features has great potential.

Since the speech modality demonstrated a potential ability to detect depressive and manic signs of BD, Khorram et al. presented another study with a bigger dataset focused on the speech modality [12]. The researchers provided 43 participants diagnosed with BD with smartphones and collected outgoing calls through a pre-installed application. The ground truth measurements were captured based on weekly phone-based instructions, which were administered by clinician specialists. This study led to the development of a program called PRIORI (Predicting Individual Outcomes for Rapid Intervention), that examined the usefulness of acoustic features as mood predictors for BD from mobile phone data. The study reports an AUC of 0.78 for the SVM classifier.

2.2.2.2 Face-to-face interview dataset

In addition to the smartphone-based datasets, a of face-to-face interviews dataset may allow researchers to develop computing algorithms with sufficient clues to assess mental disorders. As we discussed in the Section 2.2.2.1, collecting data from patients' smartphones is limited to speech, behavioral (e.g. speech, SMS) and contextual (e.g. location) data. Speech signals convey key information on one's mental health status, however, capturing additional modalities could improve the assessment performance. Hence, recording structured face-to-face interview clips has attracted attention during the past few years.

The Audio/Visual Emotion Challenge and Workshop (AVEC), introduced in 2011, targets multi-modal solutions for emotion detection [73]. BD detection was presented as one of the challenges for the 8th AVEC challenge (i.e. AVEC 2018) [74]. The TAVBD corpus [31] was selected for that purpose. We present more details about this corpus in the next section.

As the TAVBD corpus [31] includes audio/visual recordings, the baseline system for AVEC 2018 proposed a late fusion model of audio-visual features [75]. This study targeted two modalities, audio and visual, to predict individuals' mental status. It considered the Mel-Frequency Cepstral Coefficients (MFCCs), Bags-of-audio-words (BoAW), and Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for the audio modality and Facial Action Units (FAUs) and Bag-of-video-words (BoVW) for the visual modality. The system used an SVM model to classify BD states using a late fusion of the best performing audio-visual features, which were GeMAPS and FAUs. The reported Unweighted Average Unit (UAR) on the development and test partition was 0.6032 and 0.5741, respectively. These results are considered as the baseline performance on the TAVBD corpus [31] and all the following studies have tried to make an improvement on the same dataset.

Individuals with BD experience a range of energy levels depending on their mental state. A patient during Mania or Hypo-Mania episodes will experience a higher energy level compared to periods of Remission. Due to the elevated energy that subjects in manic or hypomanic states often display, rapid changes in audio and visual features may reflect the patient's condition. Therefore, speech signals are considered as informative signals for BD assessment [11][12]. To this end, researchers in [65] captured the impulsive changes in audio features using turbulence features. They applied a Fisher Vector encoding of Low-Level Descriptors (LLDs) and proved that these features are capable of capturing BD markers from speech. They utilized GEWELMs for classification and obtained a UAR of 0.5741 on the test data of the TAVBD corpus [31]. A novel audio-based model is proposed in [76]. The authors introduced an Inception module and Long Short-Term Memory (LSTM), called IncepLSTM, to capture temporal features of speech sequences. The extracted acoustic features contained the 16-dimensional MFCCs as the LLDs. Then they trained an Inception model followed by LSTM cells to predict BD severity. To obtain a better bipolar severity classification, they proposed a severity-sensitive loss based on the triplet loss to model. This study did not report results on the test dataset. Their proposed classifier achieved a UAR of 0.65 on the development data of the TAVBD corpus.

DNN networks used for feature extraction or classification require large datasets for training. To address the limited amount of data in the TAVBD corpus [31], the authors of [66] divided speech recording into smaller weakly labelled segments. Dividing the speech signal of each subject into smaller segments results in creating a larger number of examples to train the model. They fed each segment into a DNN to extract features and employed a multi-instance ensemble classifier to predict the segment's label. The extracted segments had the same label as the source recording file, however, some of these small segments were not representative of the corresponding label. To overcome this issue, they proposed a DNN multi-instance classifier, which achieved a UAR of 0.574 on the test data. Similarly, the concept of speech segmentation has been employed in another study on BD [77]. Therefore, Amiriparian et al. [77] has recently applied a Capsule Neural Network (CapsNet), which is a neural network architecture that can capture hierarchical relationships in the input [78], for an audio-based bipolar severity classification. After speech segmentation, the researchers created a spectrogram for each segment of subjects' speech signal. They trained a CapsNet on the extracted spectrograms with 32 low-level and three high- level capsules and reported a UAR of 0.455 on test data of the TAVBD corpus [31].

Although, audio signals can reveal important insights about the individuals' mental state, adding other informative modalities may increase the capacity of the recognition system. The authors of [13] utilized both audio and visual modalities. They extracted histogram-based arousal features along with a Histogram of Displacement Range (HDR) for the upper body posture. To capture the arousal values, they trained a LSTM-RNN model with the dataset of AVEC2015 [79] and then fine-tuned the trained model with the TAVBD corpus to obtain the frame-based arousal values of BD patients. A histogram was adapted to describe the distribution of arousal values within each audio segment. In addition to the histogram of audio arousal, a set of GeMAPs and LLDs were extracted from audio files. For the visual modality, they marked the 2-D key points of the upper body and calculated the distance between the patient's right and left hand in each video frame. The hand movement could provide useful information on patients' status as it relates to the level of energy and excitement of an individual. Therefore, a histogram was presented to illustrate the average distance distribution for three different categories of BD including: Mania, Hypo-Mania, and Remission. Then, they proposed a BD classifier that uses these two sets of features. The extracted appearance descriptors and audio features were classified by a DNN and random forest algorithms. They reported a UAR of 0.78 and 0.574 on the development and test data of the TAVBD corpus, respectively.

Textual features can be extracted from the interview transcripts of the TAVBD corpus [31]. In addition to the common features such as LLDs and MFFCs for the audio and FAUs for the visual modalities, [80] also relied on automatic text analysis of transcripts of interviews. They proposed a hierarchical recall model using a Gradient Boosted Decision Tree (GBDT) on three modalities. The goal of the hierarchical model is to increase the confidence of BD status detection. Hence, the model is composed of three layers. In the first layer, the model predicts the BD status and if the prediction probability is more than a predefined threshold, the prediction is considered final for that input. Otherwise, the prediction is forwarded to the next layer for further judgment. Hence, subsequent layers are only used if predictions fail to clear the threshold. In this manner, they enhanced the classification performance and achieved promising results with a UAR of 0.5741 and 0.876 on the test and development data, respectively. Zhang et al. classified an early fusion of a set of LLDs of audio-visual and textual features by a multitask DNN classifier [9]. To this end, they proposed a multimodal deep learning model to individually process each feature set (acoustic, visual, and textual). Then a Multimodal Deep Denoising Autoencoder (multi-DDAE) was designed to encode audio-visual features on frame-level. Textual features we captured and embedded into fixed-length vectors using paragraph vector models [81]. The proposed multitask DNN mitigated the overfitting issue and achieved the UAR of 0.71 on the development data without any reported results on test partition.

2.2.2.3 BD Corpus

The BD corpus utilized in AVEC 2018 [75] was introduced in [31]. This dataset depicts 218 audio-visual recordings of structured interviews with 46 Turkish speaking BD patients who were recruited from a mental health hospital. The length of the video clips varies from 13 second to 1019 seconds. Each audio-visual recording is annotated with a BD state and a YMRS score [22] estimated by psychiatrists. In these structured interviews, each subject completed seven clinically designed tasks including:

- explaining the reason to come to hospital/participate in the activity
- describing happy and sad memories
- counting up to thirty and counting down from thirty
- explaining two emotion eliciting pictures (see Figure 2.4)

As BD patients' symptoms may vary over time, subjects participated in the interview on every follow-up with the psychiatrists (0th- 3rd-7th-14th-28th day). The dataset includes 104 recordings for the training set, 64 recordings for the development set, and 54 recordings for the test set. There are no shared patients between the training, development and test sets, while some patients have

Figure 2.4. (left) van Gogh's Depression (right) Dengel's Home Sweet Home [23].

multiple recordings in one set (although multiple recording of the same patient may have different annotations).

According to the [31], all recordings are grouped in three different levels based on the YMRS score [22]:

- Remission: YMRS ≤ 7
- Hypo-Mania: 7 < YMRS < 20
- Mania: YMRS ≥ 20

The dataset is available for research purposes. All labels for the training and development sets are available for researchers but the test labels are kept confidential. Therefore, only data the providers are able to assess the prediction accuracy for the test set. As the Figure 2.5 shows, the dataset is not balanced which renders the classification task more challenging. To overcome this issue, we apply a balancing method which we will discussed in Section 4.2.3. We refer readers to [31] for more details about the dataset

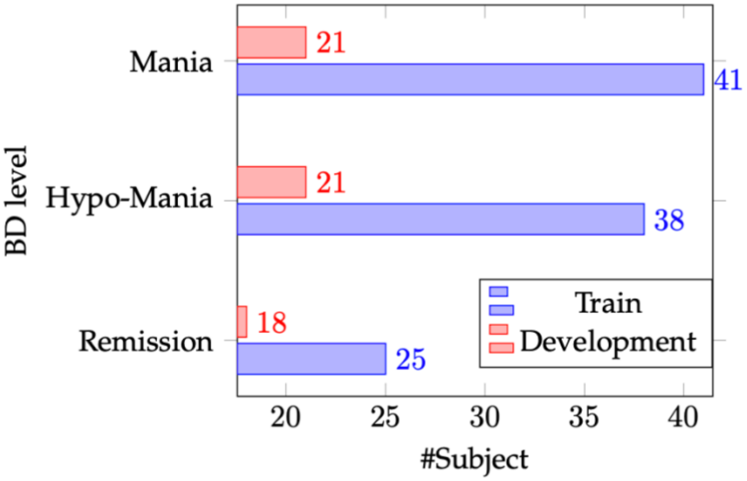


Figure 2.5. Number of patients in different bipolar disorder states for training and development partition.

2.2.3 Automated Depression Detection

Researchers developing automated depression solutions have focused on two tasks: determining whether one is depressed or detecting the level of depression using classifiers and regressors, respectively. AVEC 2014 [82] , 2016 [83], 2017 [29], and 2019 [84] targeted automated depression screening and proposed different datasets.

The dataset for AVEC 2014 [82] was a set of video recordings of subjects within a human computer interaction (HCI) environment. Subjects sat in front of a webcam and performed two tasks while they were being recorded. The first task was reading aloud a piece of a short story and the second one was answering to couple of pre-defined questions (such as “what’s your favourite dish?", "Discuss a sad childhood memory"). The objective of AVEC 2014 [82] was depression severity detection. Prior to the experiment, individuals filled a self-administered BDI-II questionnaire [85] which quantified depression severity. The baseline results reported the RMSE of 9.26 and 10.86 on development and test partition, respectively. To this end, researchers in [86] explored the relation between arousal and valence features with depression severity. They verified that reinforcing efficient dimensions can be productive for depression severity detection and their proposed model achieved RMSE of 10.82 on test partition. As Fisher Vector encoding has performed well on image classification tasks, Jain et al. investigated Fisher Vector encoding for both audio and visual features on AVEC 2014 [82]. The proposed Fisher Vector encoding model achieved RMSE of 8.17 on development data and RMSE of 10.25 on test data. Dynamic changes in audio and facial expressions provide meaningful cues for severity detection [87]. Jan et al. [87] extracted LLD audio features along with histogram of oriented gradient for visual features. Then, quantified all local dynamics of extracted features using motion history histograms. The reported RMSE is 10.26 on test data. Mitra et al. stated that considering various aspect of audio features, such as prosodic and acoustic features. They obtained novel feature sets including: estimated articulatory trajectories during speech production, acoustic characteristics, acoustic-phonetic characteristics and prosodic features [49]. Then, different approaches such as SVM, Gaussian backend, and decision tree were trained on captured features. The reported RMSE on test data is 11.10.

The target of AVEC 2016 [83] was classifying subjects to two classes, depressed and non-depressed. This challenge used the DAIC-WOZ dataset , which is part of a larger corpus i.e. the Distress Analysis Interview Corpus (DAIC) [57]. Prior to the experiment, each subject filled the assessment questionnaire based on PHQ-8 [23] which allocates a label (depressed or non-depressed) to each subject. Due to ethical concerns, the raw video recordings were not made available for researchers, however, some extracted visual feature sets are provided. The dataset providers released the raw audio recordings along with transcripts of the interviews. Assessment criteria for submitted studies on this challenge was based on average F1 score for prediction of subjects as either depressed or not-depressed for development and test partition. The baseline results are 0.50 and 0.70 of mean F1 score for development and test data, respectively. Ma et al. [56] proposed a combination of CNN and LSTM, *DepAudioNet*, to classify the audio representations. Their model achieved a mean F1 score of 0.61 on development data and they didn't provide results on test data. Another study [88] investigated the combination of the audio and visual modality. They also explored various multi-resolution windows to merge LLDs from interview sessions. The proposed model gained mean F1 score of 0.55 on test data. Due to the availability of interview transcripts, Yang et al. [89] classified textual features along with audio and visual cues using a decision tree classifier. The textual features contained related information to the PHQ-8 forms, which were extracted manually. Audio and visual features were formant features and geometric features, respectively. Their model achieved the mean F1 score of 0.72 on test data, which could beat the result of the baseline paper (mean F1 score of 0.70). Williamson et al. [90] computed the correlation between the formant of MFCC features from audio signals and FAUs from visual signals. Then they extracted textual clues similar to the approach that Yang et al. [89] used. The reported performance showed the mean F1 score of 0.70 on test data.

The AVEC 2017 [29] used the same dataset as the previous challenge, the DIAC-WOZ dataset. This challenge invited researchers to propose a model for depression severity detection as a regression task rather than the classification task of AVEC 2016. Dang et al. [91] considered MFCC features and FAUs for audio and visual modalities. For text features, they explored the idea that people with depression tend to use more negative sentiments than non-depressed ones. They extracted text-based cues using the Suite of Automatic Linguistic Analysis Tools (SALAT) [92] from the interview transcript files. The study reported a performance of RMSE = 6.02 on test data. Yang et al. [93] extracted LLD features using the open-source toolkit openSMILE [94] and fed them to a CNN to learn deep features. In their study, HDR of facial landmarks were computed as visual cues. In addition to the audio and visual modalities, they analysed subjects' answers to each question related to psychoanalytic aspects associated with depression symptoms. Their proposed model achieved RMSE of 5.79 on test data. Another study on this challenge [95] explored an ensemble model of random forest regressors to detect depression severity from the baseline audio/visual features. Researchers also manually collected some meaningful features from interview transcripts. On the test data, they achieved RMSE of 6.97.

The dataset of AVEC 2019 was the extended version of DIAC-WOZ [57] where some AI interviews were added, which was controlled by the automated SimSensei Kiosk system [96]. The SimSensi Kiosk is a virtual human interviewer designed to make users comfortable to talk and share information. The main purpose of creating this system was providing an interactional situation for automatic assessment of depression, anxiety, and post-traumatic stress disorder [96] .In the Section 2.2.3.1 will provide more details on this dataset. The main task of AVEC 2019 [84] was to predict a subject's degree of depression, formulated as a PHQ-8 score in a range $\in [0, 24]$, from vision, audio and text recordings. The dataset provider provided researchers with a set of audio and visual features including LLDs, FAUs, and deep representations feature sets (we will elaborate more in Chapter 3). The Authors of [84] used all extracted cues to train a 64-d GRU recurrent network followed by a 64-d fully connected layer regressor. This challenge adopted Concordance Correlation Coefficient (CCC) [97] and RMSE as the two metrics for ranking the performance of depression severity prediction. In the baseline study [84], fusion of the different audiovisual representations is achieved by averaging their scores. The fusion achieved the best result on development and test data. They obtained a CCC = 0.336 and RMSE = 5.03 on the development data, and CCC = 0.111 and RMSE = 6.37 on the test data. Zhang et al. [9] designed a multi-DDAE to capture audio and visual representations, which was followed by the Fisher Vector encoding to produce the session-level features. Furthermore, they collected depression related cues from transcripts using a paragraph vector model. After the early fusion of all extracted modalities, a proposed DNN regressor predicted the depression level. Based on the reported results on the development set, the textual features performed better than multimodal fusion with CCC score of 0.56.

Another study on this challenge, proposed two hierarches of bidirectional LSTM [98]. The first one extracted audio and visual features from video clips. The audio features contained two sets of deep features extracted from VGG-16 [99] and DenseNet201[100]; both sets were provided by baseline paper [84]. The second one aimed to merge all audio, visual, and textual features to regress the depression severity. The proposed model achieved a CCC score of 0.442 and RMSE of 5.50 on the test data. Fan et al. [101] investigated a multi-scale temporal dilated CNN regressor for a two task-specific feature sets. For the textual modality, they extracted sentiment scores using Bidirectional Encoder Representations from Transformers (BERT) [102]. In addition, they calculated statistical descriptors on the extracted temporal features for both the audio and visual modalities. The performance of the proposed model was reported with a CCC score of 4.30 and RMSE of 5.91 on the test data.

Makiuchi et al. [103] designed a multimodal fusion of linguistic and speech signals to detect depression severity. To extract audio representations, they converted audio files to spectrogram images followed by a DNN model, VGG16 [99], and a Gated Convolutional Neural Network (GCNN) to predict the PHQ-8 score. They employed a pre-trained BERT model to capture textual features and then fed them to a CNN-LSTM model. The proposed multimodal fusion model achieved a CCC score of 0.696 and 0.403 on development and test datasets, respectively. The RMSE was reported as 3.86 on the development data and 6.11 on the test data. Ray et al. [104] explored the efficiency of learning inter and intra modality relevance using a multi-level attention model. The multi-level attention model improves the learning process by focusing on the most meaningful features within each modality. A pre-trained Universal Sentence Encoder [105] extracted sentence embeddings as the input to the proposed regression model including 2 layers of a stacked bidirectional LSTM network. For audio representations, authors captured low-level features along with their statistical functions such as mean, variance. For the audio modality, they trained another 2 layers stacked bidirectional LSTM network where each layer had 200 hidden units. In addition to the textual and audio modalities, they used the low-level visual cues that were provided by the baseline paper. Then, a single layer bidirectional LSTM with 200 hidden units was trained for each visual feature set. This study did not report a CCC score, however, the proposed model achieved the RMSE of 4.28 on the development set.

2.2.3.1 Depression Corpus

The utilized dataset in AVEC2019 [84], E-DAIC, was introduced in [96]. E-DAIC is the extended version of DAIC-WOZ [57] which includes semi-clinical interviews of 275 US army veterans. The dataset was designed for automatic diagnosis of psychological distress such as depression and anxiety. E-DAIC dataset contains two set of interviews:

- AI interviews: which are performed by an animated virtual interviewer called Ellie. Agent Ellie acts in a fully autonomous way using different automated perception and behaviour generation modules.
- WOZ interviews: which are controlled by a human (wizard) in another room.

These structured interviews initiated with neutral questions about participants' living situations and then the interview focused on finding any related symptoms of psychological distress by asking various questions such as whether the participant has sleeping trouble or experienced any traumatic events in her/his life. Collected recordings provide audio and visual data along with automated transcripts using Google Cloud's speech recognition service. Extracted transcripts don't contain interview questions.

The 275 interviews were partitioned to the three main sets including training (163 clips), development (56 clips), and test (56 clips) while the overall diversity of the speakers - in terms of age, gender distribution, and the PHQ-8 scores - was preserved. Depression severity, PHQ-8 score, and depression binary values are provided for each participant. Training and development samples are a combination of AI and WOZ interviews, however, test set entirely contains autonomous AI interviews.

2.3 Summary

In this chapter we reviewed the basics of machine learning methods and provided its application on mental disorder detection. We specifically explained the state-of-the-art studies on automated BD and depression detection systems. We mainly focused on two challenges for the task of BD and depression detection, AVEC 2018 [74] and AVEC 2019 [84]. The AVEC 2018 invited researchers to submit their findings for BD detection on the TAVBD corpus [31], which we reviewed proposed models and performance of each. One proposed task on AVEC 2019 was depression severity detection on E-DAIC corpus [96]. We covered all the methodologies and reported results of submitted papers on this challenge. In addition, we mentioned specifications of the selected corpuses for both BD and depression.

As the next step, we need to obtain meaningful representations for subjects in each dataset. Extracting informative verbal and non-verbal features is the key part for having an accurate automated detection. In the next chapter we explain feature extraction step in details.

Chapter 3 Feature Extraction

The objective of this study is to predict the level of BD and depression from patients' video recordings. For each mental disorder, the proposed automated models include two main components:

- Feature extractor (visual, audio, and textual signals)
- Classifier for BD (LSTM, Stacked ensemble model, Semi-supervised model) and regressor for depression (Stacked ensemble model)

In this chapter, we provide more details about feature extraction. This study aims to automatically differentiate various levels of BD, including Mania, Hypo-Mania, and Remission, and detect depression severity. Automated screening methods capture verbal and non-verbal cues to identify mental states of individuals [106]. The success of these methods potentially relies on the extraction of meaningful and informative features. Hence, the first step in developing an automated method is identifying the most relevant feature sets for the task at the hand.

Since both datasets, TAVBD [31] and E-DAIC [96], collected patients' audio/visual recordings, we can investigate three main modalities: visual, audio, and textual. The rest of this chapter provides more details on each modality and the type of features we extract from it.

3.1 Visual Features

The human face conveys information about age, gender, identity along with useful clues about a person's feelings [107]. Many researchers have studied the effectiveness of using facial activities, such as head pose, eye gaze, and duration of smile, to detect mental disorders such as depression

[108], autism [109], and schizophrenia [110]. Furthermore, Derntl et al. investigated the efficiency of emotion recognition to characterize the different states of BD [111]. However, this work is still very much under development.

Visual features consist of three main types [20]:

- **Geometric features:** Each face has some nodal points, usually 80, to identify key points on a face such as nose, eyes, chin, eyebrows, and mouth. These key points are called facial landmarks. Geometric features compute the size, shape, and angle between different facial landmarks.
- **Appearance/Texture features:** Different emotions and feelings trigger various facial muscle movements. Hence, facial expressions cause changes in skin texture such as wrinkles, bulges, and furrows and appearance features depict these facial textures.
- **Deep learning features:** Deep learning models, specifically CNNs, can extract facial descriptors from face images. One may pass a face image to a pre-trained deep learning model to capture visual features or build a new deep learning model from scratch instead of a pre-trained one.

Traditional approaches for visual feature extraction are either based on geometric or appearance features. Another possibility is extracting a combination of these two feature sets, which is called a hybrid approach. Some of the generally known handcrafted features correspond to the distance and angle between landmarks (geometric features), Local Binary Pattern (LBP) histogram of different face regions (appearance features) [112]. The advantage of using conventional over deep learning approaches is that the former requires lower memory and computing power compared to the latter. However, as an expert should design the feature extractor, finding the most relevant feature set for a particular problem may be challenging.

Deep learning models have demonstrated a great potential in extracting image and video features for various computer vision tasks. Several studies showed that these models can learn pertinent features for the required task given that they are trained on a sufficiently large dataset [113],[114]. The main advantage of a CNN is to completely remove or highly reduce the dependence on physics-based models and/or other pre-processing techniques by enabling "end-to-end" learning directly from input images [115].

In Sections 3.1.1 and 3.1.2, we provide more details on the captured visual features for both tasks at the hand.

3.1.1 Visual Descriptors for BD

TAVBD [31] contains a video recording for each individual with BD. Therefore, the input to the feature extractor is the frames of each video recording. However, we need to preprocess these frames before we extract the relevant features.

3.1.1.1 Preprocessing

As each frame depicts a person with a background, we need to apply preprocessing before feeding the frame to the feature extractor. As Figure 3.1 shows, the preprocessing steps include *face detection*, *face normalization*, and *face alignment*.

To automatically detect faces in each video frame, we utilize the Histogram of Oriented Gradients (HOG) algorithm [116] implemented in Dlib toolkit [117]. We extract the face area of each patient while removing the background and hair. If a patient covers his/her face or something blocks the camera, no face can be detected and we simply skip the frame in question. After capturing face images, we apply histogram equalization [118] on each image to make it more invariant to light and increase the contrast in areas with a lower contrast.

Face alignment plays an important role in increasing the accuracy of the classification. It can tackle variations caused by different lighting conditions and/or movement of the head. A face alignment method identifies the geometric structure of human faces in digital images. Given the location and size of a face, it automatically determines the shape of the face components such as eyes and nose. We use the Dlib toolkit [117] alignment function to align the face in the processed frames.

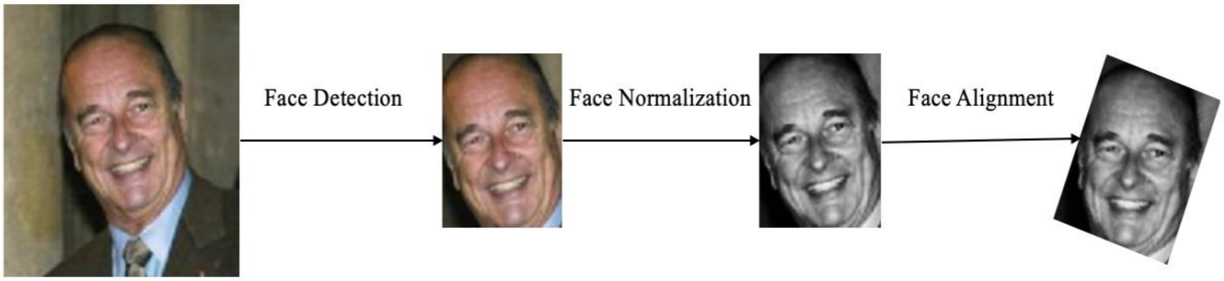


Figure 3.1 Preprocessing steps for each video frame.

3.1.1.2 **Feature Extraction**

After preprocessing, we need to extract the desired visual features from aligned face images. We discussed three different visual feature sets in Section 3.1. Due to the recent success of deep CNN approaches, we integrate a CNN model into our method. Despite several advantages of using deep CNN approaches, over-fitting is a common issue in CNNs, especially when the dataset is small. One solution is to re-use the weights from pre-trained models that were developed for standard computer vision benchmark datasets such as ImageNet and VGG-Face [119] and adopt them to the desired application setting [120]. The ImageNet was developed for object recognition while the VGG-Face was used for face recognition. Since our task is related to facial properties, using the VGG-Face model is more suitable than ImageNet.

The Visual Geometry Group (VGG) at Oxford University developed the VGG-Face model [119] as an application of the very deep ConvNet architecture VGG-16 [99]. The VGG-Face model was trained on a database of 2.6 million face images and comprised of 2622 unique identities. The database used is made of up to a thousand instances of each subject. Figure 3.2 depicts the architecture of VGG-Face which is made of a stack of 13 convolutional layers with filters having a uniform receptive field of size 3 × 3 and a fixed convolution stride of 1 pixel. As the Figure 3.2 illustrates, there are five groups of convolution layers which are separated with a max-pooling layer. The output of the last max-pooling layer is followed by three fully connected layers; FC6, FC7, and FC8. The first two have 4096 channels, while FC8 has 2622 channels which are used to classify the 2622 identities [121].

Hence, we start with fine-tuning the pre-defined VGG-Face model [119] with Facial Expression Recognition (FER) 2013 dataset [122]. The FER 2013 dataset was proposed to search for different face images with various facial emotion expressions. This corpus contains 35887 images, with 4953 *Anger* images, 547 *Disgust* images, 5121 *Fear* images, 8989 *Happiness* images.

We freeze all VGG-Face layers except the last pooling layer (pool5) and then define a classifier with 512 hidden layers on top of that. Classifier weights are updated during the training of the model with the FER2013 training set including 28,709 images. Now we can employ the fine-tuned model as a feature extractor. We collect all frame features from the CNN's layer FC6, the first layer after the last pooling layer, with 512 dimensions. We normalize the extracted features per frame and create a sequence of all frames' features for each video clip to feed to a classifier.

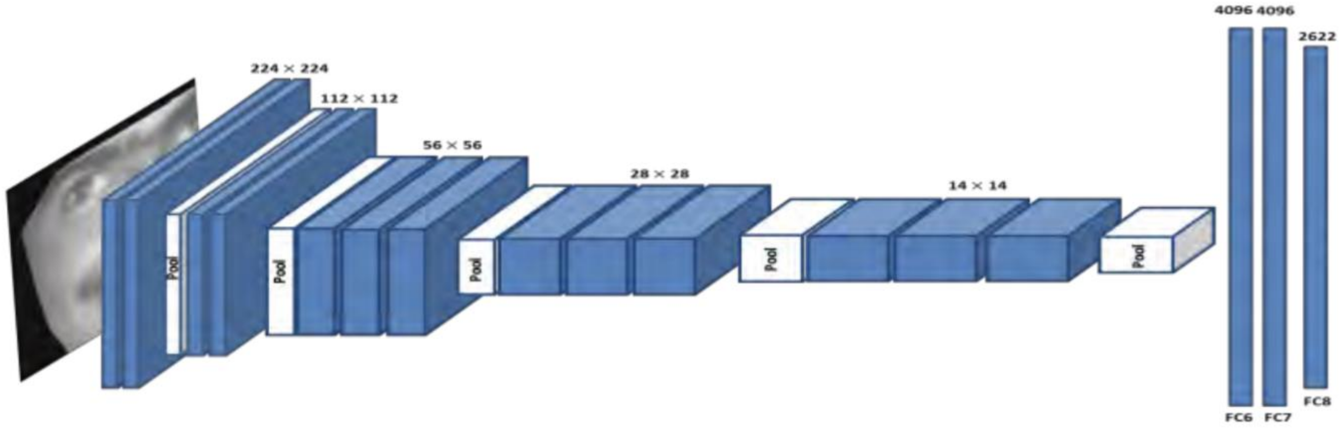


Figure 3.2 Architecture of VGG-Face model [118].

3.1.2 Visual Descriptors for Depression

As no raw video file is made publicly available for the E-DAIC WOZ corpus [96], we can only take advantage of the extracted visual features by the dataset provider. The AVEC2019 [84] provided low-level visual features along with their functionals using the OpenFace toolkit [123]. LLD visual descriptors for each video frame contain intensities of 17 FAUs, head position and pose, and gaze direction were presented as a feature vector. Each of these features was sampled at 10Hz. Furthermore, the functional of LLD visual descriptors are summarized over time by calculating the mean and standard deviation using a sliding window of 4s length and a hop size of 1s.

In addition to the LLD visual descriptor, the baseline paper [84] took advantage of two captured deep visual presentations. They employed a similar approach to the one we adopted for the BD dataset (as we explained in Section 3.1.1). They extracted the face region and performed face alignment for each frame using the OpenFace toolkit [123]. Then they individually trained two popular pre-trained models, the VGG16 [99] and ResNet-50 [124] networks, with the extracted aligned faces as inputs. While training, the original weights of two pre-trained models were frozen. To capture the deep cues for each aligned frame, they obtained the output of the first fully connected layer from the VGG16 model and the output of the global pooling average layer from the ResNet-50 model.

3.2 Audio Features

The speech is the most natural way for communication between humans. Therefore, researchers have found the speech modality as one of the most efficient methods of interaction between machines and humans. Speech signals are measurable quantities which enables machines to capture a plethora of features. Recently, speech signals have proven to contain behavioral clues for both emotional and mental states [125].

Recent studies have proven the efficiency of using acoustic features for several affective computing applications such as emotion recognition [126]. The objective of emotion recognition is to infer several feelings such as anger, fear, happiness, surprise, and sadness alongside with arousal and valance. The effectiveness of speech signals is not only limited to the emotion recognitions but also in various mental disorder monitoring, such as depression [82], Alzheimer's disease [127] , schizophrenia [128] and many other mental disorders.

Extracting suitable speech descriptors requires applying proper digital signal processing (DSP) algorithms. Given the fact that speech signals have a non-stationary nature, all DSP algorithms operate on small chunks of the speech signal instead of considering the entire signal at once. The speech signal in each chunk is assumed as a quasi-stationary signal and the length of extracted frames could vary from 15 to 40ms. These features are called LLDs and provide local information about speech recordings.

LLDs can be categorized into four main descriptors as follows:

- **Shape Descriptors:** These features provide shape characteristics for the time and frequency domain. A feature set that offers information about the shape of the signal in the time domain (its amplitude over time), is called the temporal shape. Furthermore, spectral shape features focus on the amplitude of the spectrum in the frequency domain of the speech signals.
- **Characteristic Descriptors:** Peak energy, zero crossing rate, and root-mean-square energy are examples of characteristic descriptors.
- **Perceptual Descriptors:** This set contains three main feature sets including *prosodic*, *voice quality*, and *perceptual spectral shape* features. The pattern of rhythm in speech signals is modeled by prosodic features, such as pauses, pitch, loudness, etc. Voice quality features provides information about characteristics such as tenseness and breathiness in speech signals. Perceptual spectral shape features wrap the amplitude of the spectrum signals to match the psychoacoustic frequency scales such as Mel. Hence, MFCCs are the most common feature set in this category.

Several studies have shown mental disorders affect muscle movement which causes changes in the speech production system by increasing or decreasing vocal tract muscle tension [129] [130]. In addition, recent studies on bipolar and depression detection proved the effectiveness of audio LLD features [74], [84]. Therefore, in this thesis, we mainly rely on LLD descriptors such as perceptual spectral shape features (like MFCCs). In Section 3.2.1 we discuss the details of extracted audio features for both BD and depression detection model.

3.2.1 Audio Descriptors for BD and Depression

Speech signal can enable us to monitor behavioural changes in individuals with BD or depression. People in different states of BD experience various levels of arousal and valance that can be measured through speech signals [131]. Cummins et al. [132] comprehensively reviewed studies on depression and suicide risk assessment and stated the effectiveness of analyzing speech signal for this aim.

Although speech signals are reflective of the human emotion and mental health status [126], choosing suitable speech features plays a key role in realizing an accurate system. Effectively selecting audio features is a vast area of research and undoubtedly depends on the application [126], [82]. Recent studies on mental disorder detection investigated the efficiency of spectral LLD features to capture corresponding symptoms [84], [74]. As the objective is modeling the speech characteristics of individuals with mental disorder, such as BD and depression, speech spectral modeling is a proper option for this aim. Therefore, the AVEC 2018 [74] and AVEC 2019 [84] challenges publicly published extracted LLDs from audio-visual recordings which we utilized as our audio features. Both challenges used the same approach for extracting LLD audio features. We explain this approach below.

As we described in the Section 3.2, due to the non-stationary nature of speech signals, analyzing them in short segments is preferable over considering the entire recording at once. Hence, Ringeval et al. [74] measured two main LLDs using the open-source toolkit openSMILE [94] as follows:

- **MFCCs** [133]**:** consists of 39 descriptors, including 13 Mel-frequency cepstral coefficients and 26 dynamic coefficients(delta and double-delta), at the frame level. These features are extracted with the frame length of 60ms and frame shift of 10ms.

- **GeMAPS** [134]**:** another most common audio feature set is GeMAPS or its extended version, eGeMAPS. All eGeMAPS are computed at the speaker turn level by a Long Short-Term Memory (LSTM) [135]. The 23 extracted eGeMAPS features are composed of 3 energy/amplitude related parameters, 6 frequency related parameters, and 14 spectral parameters.

The span-wise statistical audio features are generated by extracted LLD audio descriptors. As the Figure 3.3 demonstrates, the MFCCs and eGeMAPS features of each training sample are equally divided into N time spans. In each time span τ_i , four statistic functions including maximum, minimum, average and standard deviation are calculated to improve the robustness [101]. They are respectively defined as

$$f_{max}^{i} = \left\{ \max_{t\epsilon\tau_i} f_{MFCCs}(t),\ \ \max_{t\epsilon\tau_i} f_{eGeMAPs}(t) \right\} \tag{3.1}$$

$$f_{min}^{i} = \left\{ \min_{t\epsilon\tau_i} f_{MFCCs}(t),\ \ \min_{t\epsilon\tau_i} f_{eGeMAPs}(t) \right\} \tag{3.2}$$

$$f_{mean}^{i} = \left\{ \frac{1}{\tau} \sum_{t\epsilon\tau_i} f_{MFCCs}(t),\ \frac{1}{\tau} \sum_{t\epsilon\tau_i} f_{eGeMAPs}(t) \right\} \tag{3.3}$$

$$f_{var}^{i} =$$

$$\left\{ \sqrt{\frac{1}{\tau} \sum_{t\epsilon\tau_i} \left(f_{MFCCs}(t) - \frac{1}{\tau} \sum_{t\epsilon\tau_i} f_{MFCCs}(t) \right)^2},\ \sqrt{\frac{1}{\tau} \sum_{t\epsilon\tau_i} \left(f_{eGeMAPs}(t) - \frac{1}{\tau} \sum_{t\epsilon\tau_i} f_{eGeMAPs}(t) \right)^2} \right\} \tag{3.4}$$

Where $i \epsilon \{1,2,\ldots,N\}$. For each time span τi , the four extracted statistic features are concatenated into $f_{audio}^{i} = \{f_{max}^{i}, f_{min}^{i}, f_{mean}^{i}, f_{var}^{i}\} \epsilon \mathbb{R}^{(39+23)\times 4}$. Finally, for each sample, we concatenate the span features f_{audio}^{i} and yield the span-wise statistical audio features $f_{audio}^{i} \epsilon \mathbb{R}^{N\times(62\times 4)}$, which is one of the inputs for the proposed classifier.

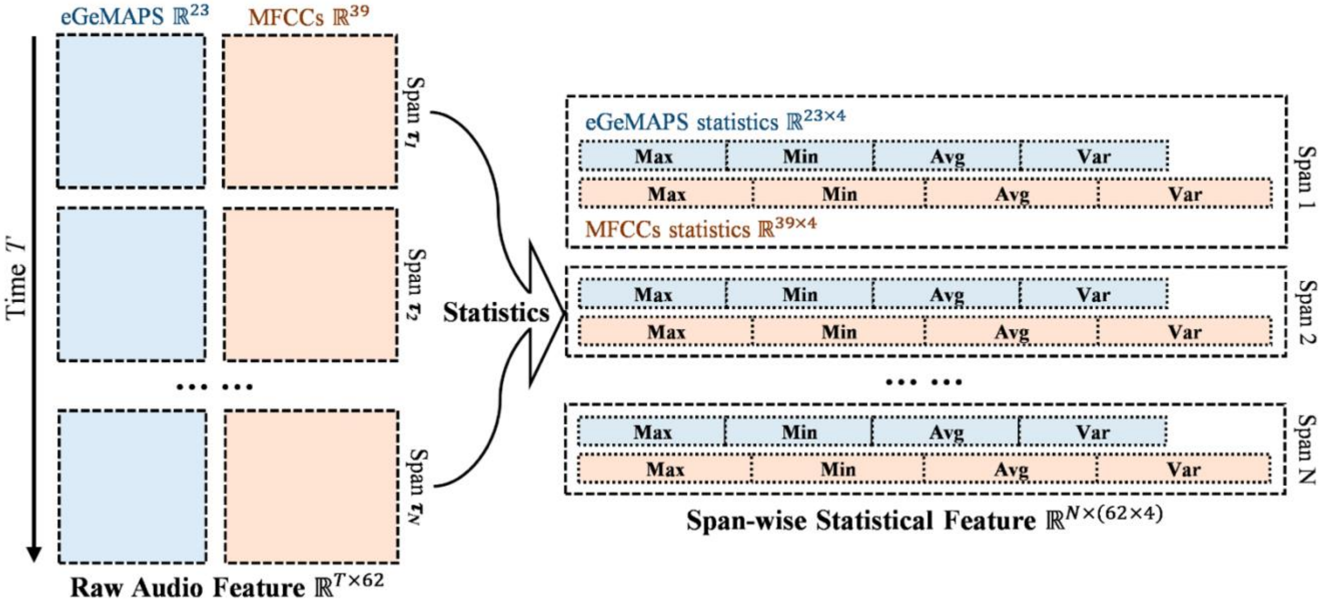
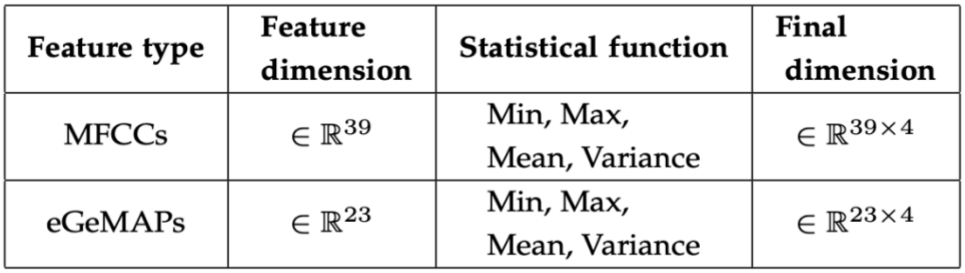

eGeMAPS $\mathbb{R}^{23}$
MFCCs $\mathbb{R}^{39}$
Time T
Span τ_1
Span τ_2
Span τ_N
Raw Audio Feature $\mathbb{R}^{T\times 62}$
Statistics
eGeMAPS statistics $\mathbb{R}^{23\times 4}$
Max
Min
Avg
Var
MFCCs statistics $\mathbb{R}^{39\times 4}$
Span 1
Span 2
Span N
Span-wise Statistical Feature $\mathbb{R}^{N\times(62\times 4)}$

Figure 3.3 The process of extracting span-wise statistical audio features [100].

Table 3.1 provides all details on the dimensions of each extracted audio feature set before and after applying statistical functions. Accordingly, we have 248 audio features for each audio sample.

Table 3.1 Audio features.



Feature type	Feature dimension	Statistical function	Final dimension
MFCCs	$\in \mathbb{R}^{39}$	Min, Max, Mean, Variance	$\in \mathbb{R}^{39\times4}$
eGeMAPs	$\in \mathbb{R}^{23}$	Min, Max, Mean, Variance	$\in \mathbb{R}^{23\times4}$

3.3 Textual Features

In addition to audio and visual descriptors, many researchers have explored textual cues to capture more features. As [136] states, syntactical elements can reflect an individual's feelings and mental status. The main objective of the study was analyzing the verbal behaviour of individuals with mood/anxiety disorders. They investigated the transcripts of cognitive therapy sessions from 85 patients using the software program Linguistic Inquiry and word Count. The results showed that patients with depressive disorders differ in their word use from those with anxiety disorders.

As there is a relation between spoken language and quantitative linguistic descriptors, Natural Language Processing (NLP) analyzes high-level language components including: grammar structure and word choice, to provide cues to textual content [137]. Researchers in [138] reviewed the state-of-the-art works that used NLP techniques to infer people's mental status based on what they posted on Facebook, Twitter, blogs, and other social media. Word-frequencies within a text and n-grams (sequence of n words, e.g. *Medium Blog* is a 2-gram) are some examples of NLP analysis. In addition to linguistic representations, conversation sentiment analysis reflects human opinions and attitudes. Sentiment analysis methods have proven to be effective for various text processing applications, such as financial forecasting, product review assessment, and social media monitoring [139].

Many studies have aimed to measure the mood valance through the sentiment analysis along with linguistic analysis [140]. However, as [138] emphasized, the main reason for ignoring the textual modality in many applications is the limited number of datasets that provide textual data. This changed with the AVEC 2017 dataset [91]. Authors in [91] investigated words affect features to classify depression severity. The dataset includes data from all patients being interviewed by a virtual agent. Therefore, recorded interview clips enabled researchers to extract transcripts of communication between the patients and virtual agent to use the textual modality alongside with the audio and visual modalities.

In recent studies, the Suite of Automatic Linguistic Analysis Tools (SALAT) [92] has been utilized for capturing textual features in depression and emotion recognition [91]. SALAT is an open-source toolkit that has been widely used for emotion and depression recognition systems[91]. This toolkit consists of eight different tools as follows [92]:

- **Custom List Analyzer (CLA):** is a simple but powerful text analysis tool that allows users to define desired list dictionaries and analyze texts. List dictionaries can be of unlimited length and can consist of words, words with wildcards, and n-grams.
- **Constructed Response Analysis Tool (CRAT):** is an easy-to-use tool that includes over 700 indices related to lexical sophistication, cohesion and source text/summary text overlap. CRAT is particularly well suited for the exploration of writing quality as it relates to summary writing.

- **Tool for the Automatic Analysis of Cohesion (TAACO):** is an easy-to-use tool that calculates 150 indices of both local and global cohesion, including a number of type-token ratio indices (including specific parts of speech, lemmas, bigrams, trigrams and more), adjacent overlap indices (at both the sentence and paragraph level), and connectives indices.
- **Tool for the Automatic Analysis of Lexical Diversity (TAALED):** is an analysis tool designed to calculate a wide variety of lexical diversity indices. Homographs are disambiguated using part of speech tags, and indices are calculated using lemma forms. Indices can also be calculated using all lemmas, content lemmas, or function lemmas. Also available is diagnostic output which allows the user to see how TAALED processed each word.
- **Tool for the Automatic Analysis of Lexical Sophistication (TAALES):** is a tool that measures over 400 classic and new indices of lexical sophistication, and includes indices related to a wide range of sub-constructs. TAALES indices have been used to inform models of second language (L2) speaking proficiency, first language (L1) and L2 writing proficiency, spoken and written lexical proficiency, genre differences, and satirical language.
- **Tool for the Automatic Analysis of Syntactic Sophistication and Complexity (TAASSC):** is an advanced syntactic analysis tool. It measures several indices related to syntactic development. Included are classic indices of syntactic complexity (e.g., mean length of T-unit) and fine-grained indices of phrasal (e.g., number of adjectives per noun phrase) and clausal (e.g., number of adverbials per clause) complexity. Also

included are indices that are grounded in usage-based perspectives to language acquisition that rely on frequency profiles of verb argument constructions.

- **Sentiment Analysis and Cognition Engines (SEACE):** is an easy-to-use tool that includes 254 core and 20 component indices based on recent advances in sentiment analysis. In addition to the core indices, SEANCE allows for several customized indices including filtering for particular parts of speech and controlling for instances of negation.
- **Simple Natural Language Processing Tool (SiNLP):** is a simple tool that allows users to analyze texts using their own custom dictionaries. In addition to analyzing custom dictionaries, SiNLP also provides the number of words, number of types, letters per word, number paragraphs, number of sentences, and number of words per sentence for each text. Included with SiNLP is a starter custom list dictionary that includes determiners, demonstratives, all pronouns, first person pronouns, second person pronouns, third person pronouns, conjuncts, connectives, negations, and future.

3.3.1 Textual Descriptors for BD

As [91] stated, syntactical elements can reflect an individual's feelings and mental status. The author reported that individuals in a manic state used more action verbs, adjectives, and concrete nouns. Moreover, according to the author, text content analysis revealed that individuals experiencing a manic episode are more likely to choose words that reflect a concern with power and achievement. Therefore, using NLP can provide rich information for mental disorder detection including BD.

As the TAVBD corpus [31] contains audio-visual recordings of individuals with BD, we can extract textual features from all interview transcripts. Sentiment analysis can reveal the sentiment polarity from each subject’s interview transcript. Polarity detection determines whether a text implies a positive/negative/neutral sentiment and indicates the intensity of the sentiment.

Similar to the method of textual feature extraction in [80], we automatically extract textual features from the BD interview transcripts using the two tools of SALAT toolkit : SiNLP [141] and SEANCE [142] tool.

The key feature of SiNLP [141] is its simplicity and power. SiNLP [141] mainly used for extracting linguistic analysis of a text file. The user should first select a list dictionary, which allow him/her to choose a custom word list to have a specific text analysis based on them. After selecting a list dictionary, users are able to upload the input text file to be processed based on chosen dictionary. We run the SiNLP [141] on each transcript and extract 14 linguistic features, such as the number of words, number of types, letters per word, number of paragraphs, number of sentences, and number of words per sentence.

For sentiment analysis, SEANCE [142] provides several useful indices. ANEW [143], EmoLex [144], SenticNet [145], and Lasswell [146] are four prominent sentiment indices that were proven to be effective for depression detection [147].

The ANEW [143] provides the affective norms for arousal, valence, pleasure, and dominance for around thousand English words including verbs, adjective, nouns, and adverbs. Each word is rated in range of [0,10], where any rate within [0,5] means a negative word token and greater than

5 indicates positive word token. As individuals with BD experience different states from manic to depression, negative and positive words are useful indicators to detect BD states.

The EmoLex [144] consists of token words that are related to eight various emotions, such as anger, fear, disgust, joy, sadness, surprise, trust, and anticipation. Hence, we extract 40 EmoLex features.

The SenticNet [145] consists of 13k token words, where each token word has evaluated perceptual polarity norms for aptitude, attention, pleasantness, and sensitivity. The main difference of SenticNet with other word affects feature is that it implements word multiple degrees of word associations. As an example, SenticNet consider that the word “grief ” is usually linked with “ cry “ , “depressed “, and “ sadness “ concepts.

The Lasswell features [146] are extracted from eight semantic characterizations such as affection, enlightenment, power, rectitude, respect, skill, wealth, and well-being. In total, we capture 146 Lasswell features for each transcript file.

Table 3.2 shows a summary of all extracted textual feature for BD corpus.

Table 3.2 Textual features.

Tool	**Features**		**Dimension**
SiNLP	Total number of words, unique words, number of paragraphs, number of sentences, letters per word, words per sentence, words type, pronounces, articles, negations, determiners		$\in \mathbb{R}^{14}$
SEANCE	**ANEW**	Indicating affective norms for arousal, valence, dominance, and pleasure for each word	$\in \mathbb{R}^{32}$
	EmoLex	Finding token words that relate to eight specific emotion types such as: anger, anticipation, disgust, fear, joy, sadness, surprise, trust	$\in \mathbb{R}^{40}$
	SenticNet	Calculating perceptual polarity norms for aptitude, attention, pleasantness, and sensitivity	$\in \mathbb{R}^{30}$

	Lasswell	Characterizing affection, enlightenment, power, rectitude, respect, skill, wealth, and well-being	$\in \mathbb{R}^{146}$

3.3.2 Textual Descriptors for Depression

Numerous studies have investigated that the verbal content of an individual's speech is important for assessing the depression level [104], [148]. The AVEC 2019 challenge obtained transcripts of all interviews using an open-source API, such as Google Cloud Platform (GCP). Therefore, it enables us to use the textual modality in our proposed depression detection solution.

To capture the semantic content of the E-DAIC corpus, we use the Word2Vec [149] neural network, which was introduced by Google. Word2Vec [149] processes text data based on two different leaning methods including Continuous Bag of Words (CBOW) [150] and Skip-gram [151] by calculating vector representation of words. CBOW predicts the word given its context, however, Skip-gram predicts the context given a word. In the training step, Word2Vec network creates a vocabulary from training text data learns the vector representation for each word. Furthermore, the superior of Word2Vec model is the ability of considering words similarities by calculating the cosine distance among each word.

In this thesis, we construct a pre-trained model on all training data with a window size of 5 words between current and predicted words in a sentence. After computing the embedding vector for each word, we obtain the sentence embedding vector by averaging the embedding of all the words in the sentence. Empirically we choose the embedding dimension of 300.

After obtaining all desired feature sets, we need to classify/regress them using a classifier/regressor. Therefore, in the next chapter we explore the proposed networks to implement this task for both BD and depression severity detection.

3.4 Summary

In this chapter, we discussed three modalities to capture characteristics for the task of BD and depression severity detection. After providing an introduction to each modality, namely; audio, visual, and textual, we introduced obtained features for each as below:

- Audio features: for both BD and depression severity detection model, we obtained MFCC and eGeMAPs audio features from each audio file. To compute the MFCC features we use the openSMILE [94] toolkit at the frame level for each normalized audio record. The eGeMAPs, however, are measured at the speaker turn level which are estimated by an LSTM network. After applying 4 statistical functions including: min, max, mean, and variance we ended up with 248 audio features, MFCCs and eGeMAPs, in total for each sample.
- Visual features: for BD detection model, we extracted face from each frame using Dlib toolkit [117]. Then, we fine-tuned the VGGFace model using FER2013 and then fed each extracted face to the model. The output is a feature vector of size 512 for each frame of video files. As we don't have access to raw video files for data in depression severity detection model, we used the provided LLD features by the dataset provider. They obtained low-level visual features along with their functionals using OpenFace toolkit [123]. LLD including 17 FAUs, head position and pose, and gaze direction were sampled at 10Hz. In addition, we have access to extracted deep visual feature obtaining through the VGG16 and ResNeT-15 models.
- Textual features: in BD detection model, we focused on linguistic and sentiment features through the Suite of Automatic Linguistic Analysis Tools (SALAT) [92]. We extracted desired textual descriptors from two main functions of this tool, The

Simple Natural Language Processing (SiNLP) [141] and Sentiment Analysis and Cognition Engine (SEANCE) [142]. For depression severity detection model, we took advantage of a pre-trained model called Word2Vec [152]. Using the pre-trained model, for each line in the transcription, we averaged the embeddings of all the words to got the sentence embedding. Then we considered the same embedding across the whole duration of this sentence

After extracting all meaningful representations, we need to implement classifier/regressor to learn them. Chapter 4 elaborates on the methodology for BD detection model, which is a classification problem. In Chapter 5, we provide details on regression method for depression severity detection.

Chapter 4 BD Classification

As we explained in Section 1.2, existing issues with conventional detection methods (clinician-based and self-based assessments) motivated us to propose automated methods for BD manic states detection. In Chapter 3, we provided details on extracted features to model the behavioral characteristics for the manic states of BD. In this chapter, we propose models to automatically classify subjects with BD into states of Mania, Hypo-Mania, and Remission.

The layout of this chapter is as follows: we start with a brief introduction on various types of learning techniques. Then, we provide details on three proposed classifiers which use different sets of extracted features from audio/visual data. Next, we discuss our experimental setting and results.

4.1 Proposed Classifiers for Automated BD Assessment

Various types of classification/regression algorithms have been used for automated mental disorders models. Some examples are Random Forest (RF) [153], decision tree [154], Support Vector Machine (SVM) [155], Extreme Learning Machine (ELM) [156], Long-Short-Term Memory (LSTM) [157], and ensemble learning models [158]. As the automated screening problems are inherently complex, there is no specific method to find the best classifier. However, we consider the following facts, to develop suitable models for BD detection:

- **Stacked ensemble model:** Ensemble models have proved superior performance when we have 1) insufficient available data, and 2) intertwined classes. We have both issues on the TAVBD corpus [31] . ML algorithms usually require large training datasets to perform well; however, collecting data, depending on the task, can be highly expensive and remains a major hurdle for many researchers. Therefore, employing ensemble models can help compensate for the limited size of available datasets, such as the TAVBD corpus [31], by training base models on overlapping random subsets drawn by resampling the original data. Moreover, separating classes may be a complex process for some tasks. For instance, there are similar symptoms for different BD states, especially for Mania and Hypo-Mania. Hence, these states might result in similar manifestations (e.g., facial expression, speech intonation). Therefore, it is more probable that a single classifier misclassifies them compared to an ensemble of classifiers. Hence, combining a set of classifiers results in more accurate definition of decision boundaries. Each base classifier solves this problem by learning a subset of the dataset.

- **LSTM model:** Since the TAVBD corpus [31] includes video recordings, consecutive frames are temporally related as they depict a continuum of patient behavior. Hence, accounting for the sequential nature of the information might help improve model performance. Therefore, LSTM is a promising option to capture temporal features by accepting a sequence of input features. LSTM is a type of RNN. We discussed RNNs in section 2.1.1.8. We will discuss the LSTM in section 4.2.2.
- **Semi-supervised ladder model:** The TAVBD corpus includes a training, development, and test set. However, the labels for the test set are kept confidential and are maintained by the dataset provider. This has motivated us to ponder the possibility of extending existing labelled datasets with unlabelled data. Labelling tends to be a laborious task and in the case of the TAVBD corpus, requires the expertise of clinical professionals. Hence, we propose a semi-supervised method that allows us to combine labelled and unlabelled data for BD assessment. In our case, we adopted the test data as the unlabeled addition to the dataset given that it was

available to us. However, in practice, additional unlabelled interviews with DB patients can be collected to supplement the training and development sets of the TAVBD.

In the following sections, we describe the proposed classifiers.

4.1.1 Ensemble Models

Using ensemble learning models is analogous to seeking multiple opinions before making decisions in our daily lives. For instance, when we need to make a crucial decision about a complex matter, we may consider various opinions. We would combine these (possibly opposing) opinions through a thought process to reach a final decision. For instance, we can possibly follow the opinion of the majority or rely more heavily on trusted advisers. Ensemble learning uses the same mechanism to improve prediction confidence.

Conventional machine learning algorithms use a single classifier/learner model for the classification task, while an ensemble method employs a set of classifiers/learners and applies a combination methods to realize the final prediction [41]. An ensemble model contains several weak classifiers, which are also referred to as base/sub learners. Recently, ensemble methods have overcome many existing limitations in single learner models [159]. The idea behind ensemble learning methods implies that considering the outputs of several classifiers renders more accurate predictions. In contrast to traditional machine learning approaches, which attempt to learn one hypothesis from the training data, ensemble methods combine a set of hypotheses during learning.

As we described in Section 2.1.6, the choice of base learners and combination approach of their predictions are two critical factors in designing an ensemble model. A base learner can be a neural network, decision tree, SVM or any other kind of machine learning algorithm. If all base learners are identical, we refer to the resulting ensemble as a *homogeneous* ensemble model. However, different base learner algorithms produce a *heterogeneous* ensemble model.

There are several effective ensemble combination methods; We introduce three popular approaches below:

- **Bagging** [160] **:** Training a set of identical machine learning algorithms on a randomly selected subset of the training set. Base learners learn independently in parallel. After training, the ensemble model makes a prediction by aggregating the outputs of all base learners using a statistical process.
- **Boosting** [161] **:** Boosting is similar to bagging as models are trained on a subset of the training set. However, as opposed to bagging, in boosting, the models are trained sequentially. Each training sample is assigned a weight that corresponds to its likelihood of being selected to train a model. At the beginning, all samples are assigned equal weights. The weights are updated as models are trained. The weight of a sample is increased if it is misclassified by a model. This increases the probability that the sample may be selected by the next created model. This may lead that model to correct the misclassification of the previous one. The process of sequentially creating models proceeds until the training set is classified perfectly, or a maximum number of models are added.
- **Stacking** [162] **:** All base learners are trained in parallel and a meta learner aggregate the output of all of the base models to predict the final class.

As Figure 4.1demonstrates, in some ensemble strategies such as boosting and bagging, the final decision is obtained by applying voting among the predictions of the base classifiers [163]. In the stacking ensemble strategy [162], instead of a voting process, a meta classifier is responsible for rendering the final prediction based on the predictions of all base classifiers. As Figure 4.1 depicts, in the bagging strategy, the classifiers are trained in parallel on a random set of training data and then a voting method produces the final prediction. However, boosting is a sequential process. The first classifier is trained on a training subset. Subsequent classifiers are more likely to train on the data points that were incorrectly classified by the previous model. Then, a voting process outputs the final prediction. The development of a stacking ensemble involves two main tasks: i) generating base learners in the first level, and ii) combining base learners using a meta learner in the second level. The meta learner is trained to optimally integrate the outputs of the base learners to improve the final predictions.

4.1.1.1 Proposed Stacked Ensemble Classifier

Due to the demonstrated efficiency of ensemble models in various applications, we propose an ensemble classifier for BD detection. We hypothesize that ensemble learning is suitable for BD classification given the limited size of the dataset and possible overlap between some of the BD classes. For example, some symptoms in Mania and Hypo-Mania are very similar which makes it hard to differentiate between them.

Hence, the idea of using only one classifier may not be sufficient. A single classifier may perform poorly as it tries to realize a best fit of the features, which may misrepresent the learning task. Therefore, developing an ensemble model, which contains various classifiers, could enhance the accuracy of the classification task.

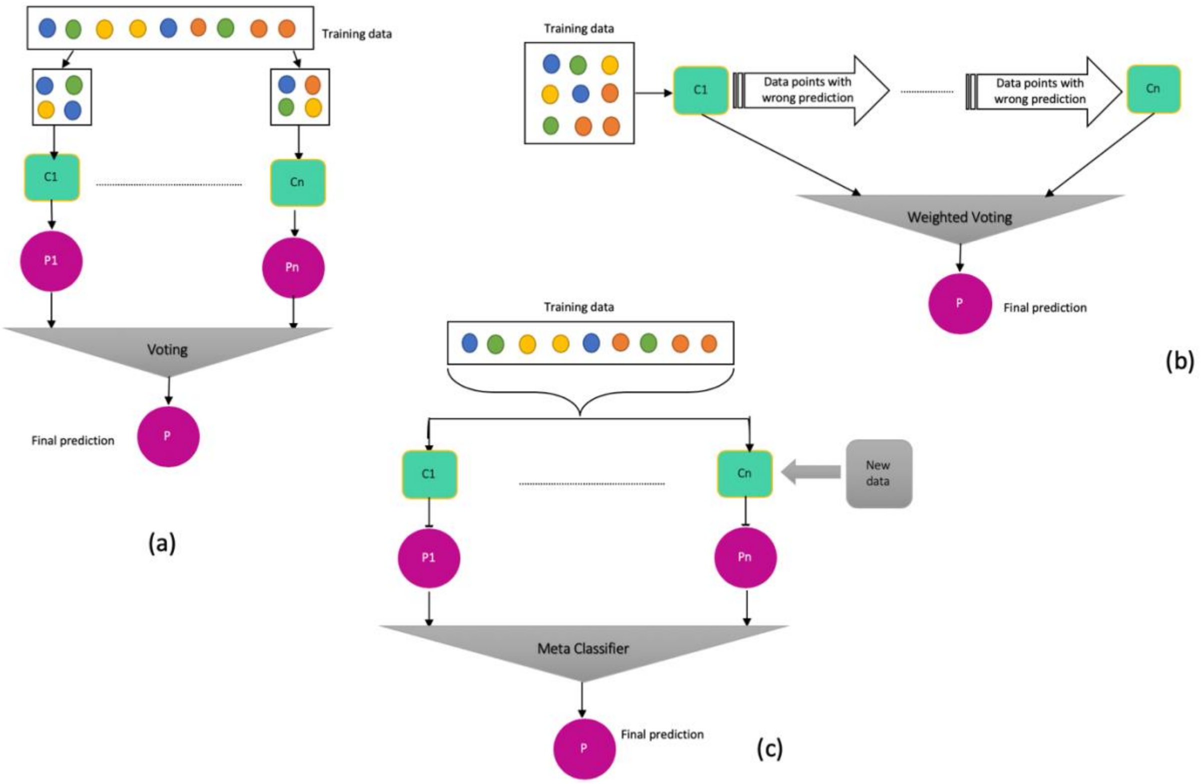


Figure 4.1 Ensemble methods: a) Bagging, b) Boosting, c) Stacking.

To implement an ensemble model, we need to decide between three common methods, bagging, boosting, and stacking. Winners of various machine learning competitions mostly used stacking ensemble models [164], [165]. Moreover, stacking models have been shown to generally perform well in classification/regression problems [166]. Hence, we propose a stacking ensemble model.

In our proposed model, we utilized a stacked ensemble model that contains three homogeneous CNN base learners. We will discuss how we selected the number of base learners in Section 4.2.2.1. The meta learner design is inspired by research on emotion recognition using speech data which indicated that a Multi-Layer Perceptron (MLP) enhanced the classification task of MFCC audio features [167]. Algorithm 1 illustrates the pseudo-code of our proposed stacking ensemble model.

Algorithm I: The Stacking Algorithm

Input: Data set $D = \{xi, yi\}_{i=1}^{m}$;

First-level learning algorithms: $\omega_1, \omega_2, \omega_3$;

Second-level learning algorithm ω.

Output: Ensemble classifier H

```
for i=1,2,3 do
  |  $h_i = \omega_i(D)$                // Training each base learner $h_i$
end for
$D' = \emptyset$                      // Constructing a new dataset
for j=l,..,m do
    for i=1,2,3 do
        $s_{ji} = h_i(x_j)$           // Classifying $x_j$ using the $h_i$
    end for
    $D' = D' \cup \{((s_{j1}, s_{j2}, s_{j3}), y_j)\}$
end for
$h' = \omega(D')$                     // Training meta learner h'
$H(x) = h'(h_1(x), h_2(x), h_3(x))$
```

Despite the success of stacked ensemble methods, choosing the best architectures for the base learners and meta learner is a critical decision. Hence, many researchers have pondered the problem of hyperparameter optimization. Hyperparameter optimization methods play a key role in machine learning, and are widely used in practice [168], [169]. Although many of these methods are effective for a wide range of problems, they mostly rely on searching for models from a fixed-sized space. Hence, the results rendered by these methods are highly dependent on starting with a good initial model. In this study, to optimize the BD model, we take advantage of a novel approach that is explained in the following.

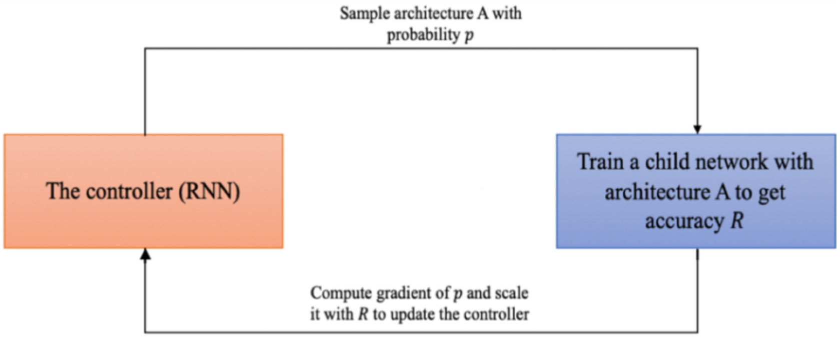

Sample architecture A with probability *p*
The controller (RNN)
Train a child network with architecture A to get accuracy *R*
Compute gradient of *p* and scale it with *R* to update the controller

Figure 4.2 Overview of Neural Architecture Search (NAS), adapted from [163].

Zoph and Le [170] proposed the Neural Architecture Search (NAS) hyperparameter optimization approach that performs model search using reinforcement learning. In NAS [170], as shown in Figure 4.2, a recurrent neural network, acting as a controller, samples several child neural networks. Training each child network yields an accuracy that is considered as a reward to update the reinforcement algorithm in the next step. Then, the controller gives higher probability to the child networks with greater accuracy in the previous step. Three elements of the reinforcement learning algorithm for this application are defined below:

- States: defined search space for each hyperparameter
- Actions: are the same as the sates
- Rewards: accuracy score of each trained child network.

So, the parameters of the controller, ξ_c, are updated to maximize the expected reward, $\upsilon(\xi_c)$, to find the optimum architecture:

$$\upsilon(\xi_c) = E_{P(a_{1:k};\,\xi_c)}[R] \qquad (4.1)$$

Where E_P is the expectation operator, $a_{1:k}$ represents all predicted architectures by the controller as a sequence of actions, and k is the number of hyperparameters that the controller tunes to achieve the best architecture. The obtained accuracy of the child networks, which corresponds to the reward for the training controller, is noted by R. To update ξ_c, we apply a policy gradient method iteratively [171] :

$$\nabla_{\xi_c}\upsilon(\xi_c) = \sum_{i=1}^{k} E_{P(a_{1:k};\,\xi_c)}[\ \nabla_{\xi_c}\ \log P(a_i|a_{(i-1):1};\xi_c)R] \qquad (4.2)$$

After simplification, it is approximately equal to:

$$\nabla_{\xi_c}\upsilon(\xi_c) = \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{k}\nabla_{\xi_c}\ \log P\big(a_i\big|a_{(i-1):1};\xi_c\big)R_j \qquad (4.3)$$

Where m refers to the number of child networks that are sampled by the controller.

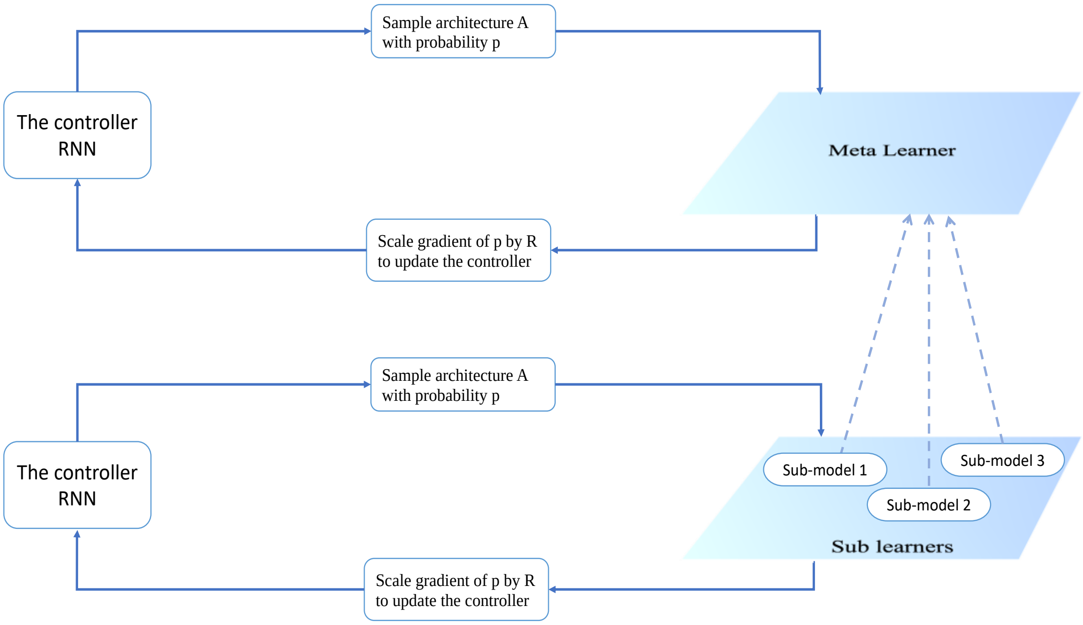

Sample architecture A with probability p
The controller RNN
Meta Learner
Scale gradient of p by R to update the controller
Sample architecture A with probability p
The controller RNN
Sub-model 1
Sub-model 2
Sub-model 3
Sub learners
Scale gradient of p by R to update the controller

Figure 4.3 Overview of Neural Architecture Search for both levels of stacked ensemble model.

We apply two RNN controllers based on the NAS [170] hyperparameter optimization method to find the best architectures for the meta and base learners in BD severity detection. Thus, NAS [170] does not only optimize the network's architecture for the base learners and meta learner, but also finds the best combination for both. As Figure 4.3 demonstrates, the base learners and meta learner are child networks of the NAS model [170] in the first and second levels of optimization, respectively.

4.1.2 LSTM Models

Long-Short-Term Memory (LSTM) networks are a special kind of RNNs that contain complex structures that permit them to handle long-term dependencies of sequential inputs. In some instances, RNNs are incapable of learning from long data sequences due to the problems of vanishing and exploding gradients. LSTM networks have built in mechanisms to address these issues. The vanishing gradients problem arises when, during backpropagation, the error signal used to train the network exponentially decreases as the algorithm travels backwards. This means that the network is unable to learn in the layers closer to the input as the gradients reaches near zero values. Conversely, exploding gradients accumulate large values during backpropagation rendering the network unable to learn. LSTM networks have a unique additive gradient structure which prevents gradients from vanishing or exploding.

As we explained in Chapter 2, all standard RNNs have a chain form of repeating modules. These modules are identical throughout the RNNs structure. However, LSTM networks form a chain of repeating modules of different structures [172]. There are three main gates in the LSTM structure including: forget gate, input gate, and output gate layer, as it is shown in Figure 4.4.

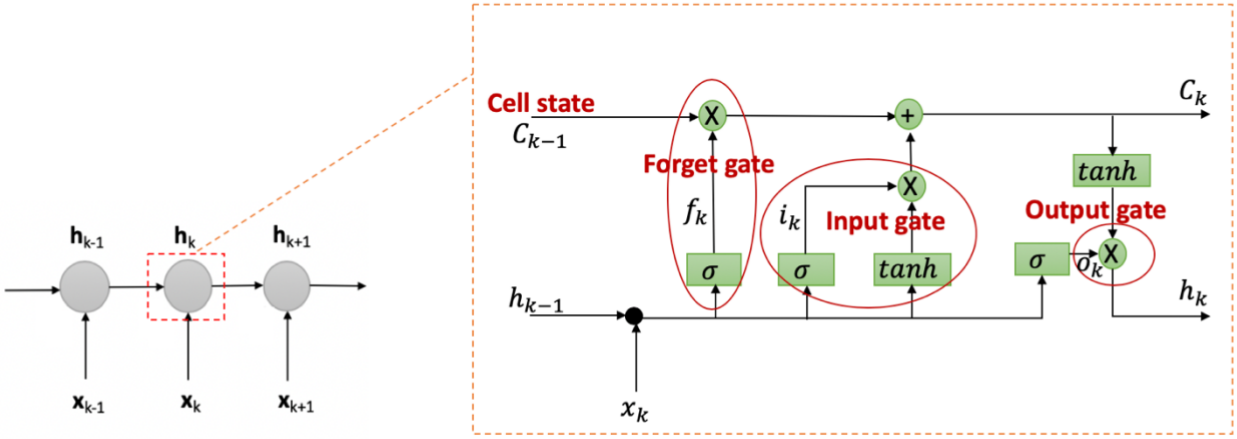


Figure 4.4 Four interactive layers: state cell, forget gate, input gate, and output gate of one LSTM cell

LSTMs are designed to remember long-term information dependency by mapping the input sequence to the output labels with hidden units. Each unit includes a built-in memory cell that stores information over time to capture long-range dynamics, with non-linear gate units controlling the information flow into and out of the cell. The main units in LSTMs structure are memory cells $\boldsymbol{c}_{\mathrm{k}}$, output gates $\boldsymbol{o}_{\mathrm{k}}$, input gates $\boldsymbol{i}_{\mathrm{k}}$, forget gates $\boldsymbol{f}_{\mathrm{k}}$ and hidden states $\boldsymbol{h}_{\mathrm{k}}$ at time k. An LSTM maps a given sequence of features ($\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_T$) to the output labels $(\boldsymbol{y}_1,\boldsymbol{y}_2, \dots ,\boldsymbol{y}_T)$ for time span of t=T by computing activations of the units using the equations below:

$$\boldsymbol{c}_k = \boldsymbol{f}_k \odot \boldsymbol{c}_{k-1} + \boldsymbol{i}_k \odot \tanh(\boldsymbol{W}_{xc}\boldsymbol{x}_k + \boldsymbol{W}_{hc}\boldsymbol{h}_{k-1} + \boldsymbol{b}_c) \tag{4.4}$$

$$\boldsymbol{o}_k = \sigma(\boldsymbol{W}_{xo}\boldsymbol{x}_k + \boldsymbol{W}_{ho}\boldsymbol{h}_{k-1} + \boldsymbol{W}_{co}\boldsymbol{c}_k + \boldsymbol{b}_o) \tag{4.5}$$

$$\boldsymbol{i}_k = \sigma(\boldsymbol{W}_{xi}\boldsymbol{x}_k + \boldsymbol{W}_{hi}\boldsymbol{h}_{k-1} + \boldsymbol{W}_{ci}\boldsymbol{c}_{k-1} + \boldsymbol{b}_i) \tag{4.6}$$

$$\boldsymbol{f}_k = \sigma\left(\boldsymbol{W}_{xf}\boldsymbol{x}_k + \boldsymbol{W}_{hf}\boldsymbol{h}_{k-1} + \boldsymbol{W}_{cf}\boldsymbol{c}_{k-1} + \boldsymbol{b}_f\right) \tag{4.7}$$

$$\boldsymbol{h}_k = \boldsymbol{o}_k \odot \tanh(\boldsymbol{c}_k) \tag{4.8}$$

Where $\boldsymbol{W}_{ab}$ refers to the weight matrix connecting two units *a* and *b*. σ is the sigmoid function, $\boldsymbol{b}_a$ is bias term and ⊙ is an element-wise product operator. As it is shown in the equations above, the content of the current memory cell is a combination of the previous memory cell information and current input cell information where the forget gate determines which part of the information should be forgotten. In addition, the output gate $\boldsymbol{o}_{\mathrm{k}}$ controls the content of the memory cell that should be passed through the hidden cells for computation in next steps.

4.1.2.1 Proposed LSTM Classifier

To the best of our knowledge, there is only one study that implemented an LSTM classifier on the TAVBD corpus. The authors of [76] considered an Inception module and Long Short-Term

Memory (LSTM), called IncepLSTM, to capture temporal features of speech sequences. However, to the best of our knowledge, there has not been an attempt to detect BD levels from video recordings using an LSTM classifier on facial features. In this section, we propose a multimodal BD detection model using an LSTM classifier. For the visual modality, we propose a hybrid model to extract non-verbal features using a CNN, and classify BD states using an LSTM network. The hybrid CNN-LSTM model can learn to distinguish dynamics in facial signals.

As we described in Section 3.1.1, the spatial feature extraction is performed by a CNN-based model fined-tuned on the FER 2013 dataset [122]. We feed the sequences of these high-level features to an LSTM, which is responsible for the classification.

As shown in Figure 4.5, which depicts the proposed LSTM classification method, all frames of a video recording are continuously captured as the input of the system. The aligned face in each frame is extracted and fed to the CNN model, VGG-Face fine-tuned with FER 2013, to form the visual spatial feature vectors. In the classification stage, the outputs of the CNN model are learnt by the LSTM which performs the prediction of the labels. Hence, the last LSTM cell outputs the predicted label of the corresponding video recording.

We will provide more details on how to implement the LSTM classifier in Section 0.

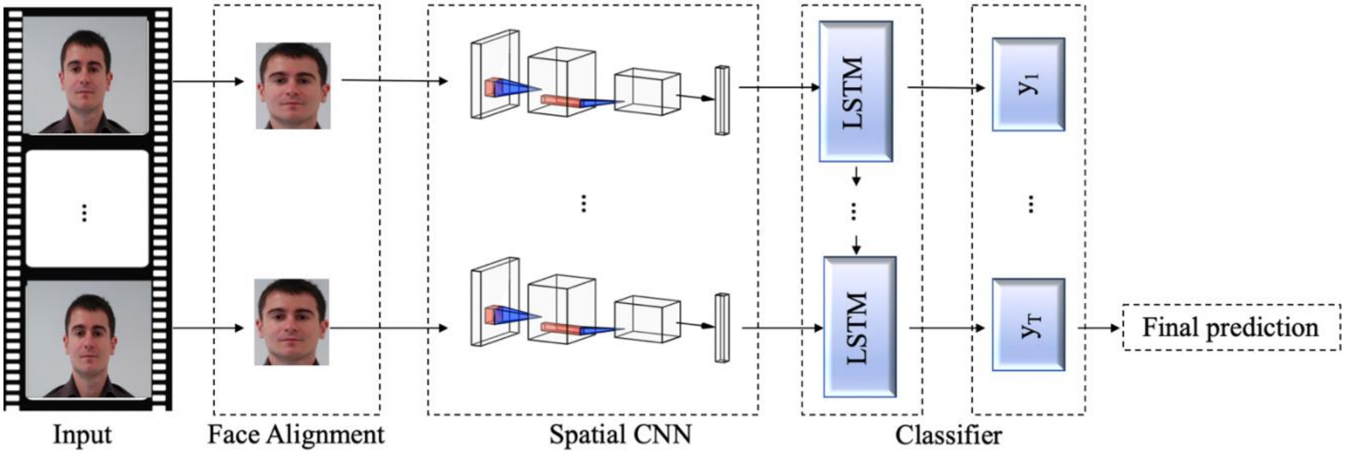


Figure 4.5 Visual processing framework of BD detection using video data.

4.1.3 Semi-Supervised Model

Many of the recent improvements in deep learning methods are associated with supervised learning. However, the effective application of such methods requires a large labeled dataset. This can present a challenge when the dataset size is limited. Semi-supervised learning enables the label prediction of a large number of unlabeled samples by training with a small number of labeled samples. Such models use a mix of labeled and unlabeled data during training.

Semi-supervised learning is halfway between supervised and unsupervised learning. In addition to some supervision information, the algorithm is provided with unlabeled data. Therefore, in this case the dataset contains both labeled and unlabeled data : $X = \{X_L, X_U\}$, where the labeled dataset $X_L = \{(x_i, y_i)\}_{i=1}^{M}$ has label set of $Y_L = \{y_i\}_{i=1}^{M}$ and unlabeled dataset $X_U = \{(x_i)\}_{i=1}^{N}$ without any provided label.

If we assume that the dataset has K different classes and the first M samples within X are labeled as $Y_L = \{y_i\}_{i=1}^{M} \in (y^1, y^2, \ldots, y^K)$, the semi-supervised learning aims to solve the following optimization problem:

$$min_\theta \underbrace{\sum_{(x,y)\in X_L} \mathcal{L}_s(x,y,\theta)}_{supervised\ loss} + \alpha \underbrace{\sum_{(x)\in X_U} \mathcal{L}_u(x,\theta)}_{unsupervised\ loss} + \beta \underbrace{\sum_{(x)\in X} \mathcal{R}(x,\theta)}_{regularization\ loss} \qquad (4.9)$$

Where θ denotes the model parameters and $\alpha, \beta \in \mathbb{R}^{>0}$ denotes the scalar weights, which balance the loss terms. $\mathcal{L}_s$ and $\mathcal{L}_u$ denote the per-example supervised and unsupervised loss, respectively. $\mathcal{R}$ denotes the designed regularization term, while normally there is no clear distinction between this term and unsupervised loss term.

The advantage of a semi-supervised model over supervised one is that the former does not require a time-consuming, expensive, and labor-intensive manual annotation task for the training data. In addition, the semi-supervised approach alleviates the need for domain experts to label the data, which is a challenging task, particularly in the healthcare domain. Semi-supervised learning has shown promising performance in many pattern recognition applications, such as emotion recognition [173], depression detection [174], and facial expression recognition [175].

The ladder network is a novel approach for semi-supervised learning, which delivered impressive results on the MNIST handwritten digit classification problem with only 100 labeled training examples [176]. This network extends the *denoising autoencoders* [177], which solely rely on unsupervised learning, by complementing them with a supervised component. Therefore, the ladder network is a deep feedforwad network that combines supervised and unsupervised learning.

4.1.3.1 Proposed Semi-Supervised Classifier

In this section, we describe the proposed ladder network classifier. We refer readers to [176], [177] for a detailed description of ladder networks. The dataset consists of M labeled data points, $\{(x_i, y_i) \mid 1 \leq i \leq M\}$, and N unlabeled data points, $\{(x_j) \mid M + 1 \leq j \leq N + (M + 1)\}$. As the ladder network is a denoising autoencoder network, we inject noise in each layer of the network to force the autoencoder to learn how to denoise the corrupted input. Thus, we have two different encoder paths with shared parameters, where one produces noisy data and the other provides noiseless data. For the encoder that produces noisy data, we combine the input vector with a noise vector, $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{noise}$, and transform the resulting vector into a latent representation, $\tilde{\boldsymbol{z}}^{(k)} \mid 1 \leq k \leq K$ where K is the number of network layers. The noiseless encoder is similar to the noisy one, except that it does not add a noise vector to the input. Hence, we denote the noisy and noiseless encoders as $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{z}}^{(1)}, \tilde{\boldsymbol{z}}^{(2)}, \dots, \tilde{\boldsymbol{z}}^{(K)} = Encoder_{noisy}(x)$ and $\boldsymbol{x}, \boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}, \dots, \boldsymbol{z}^{(K)} = Encoder_{noiseless}(x)$, respectively.

Each layer of the noisy encoder is connected through lateral connections to its corresponding layer in the decoder. This enables the higher layer features to focus on more abstract and task-specific features. Therefore, the decoder combines the two outputs, one from the layer above and one from the corresponding layer in the noisy encoder, to yield the reconstructed observation at each layer, $\hat{\boldsymbol{z}}^{(k)} \mid 1 \leq k \leq K$, where K is the number of network layers. Hence, the decoder is denoted as $\hat{\boldsymbol{x}}, \hat{\boldsymbol{z}}^{(1)}, \hat{\boldsymbol{z}}^{(2)}, \dots, \hat{\boldsymbol{z}}^{(K)} = Decoder(\tilde{\boldsymbol{z}}^{(1)},, \tilde{\boldsymbol{z}}^{(2)}, \dots,, \tilde{\boldsymbol{z}}^{(K)})$. Figure 4.6 illustrates the ladder network.

Inspired by [178], where a learnable MLP improved the performance of the ladder network, we train the ladder network with an MLP. The ladder network is trained to minimize the combination of the weighted sum of supervised cross entropy and unsupervised reconstruction cost function from the encoder and decoder paths, respectively. We formulate these cost functions in this section.

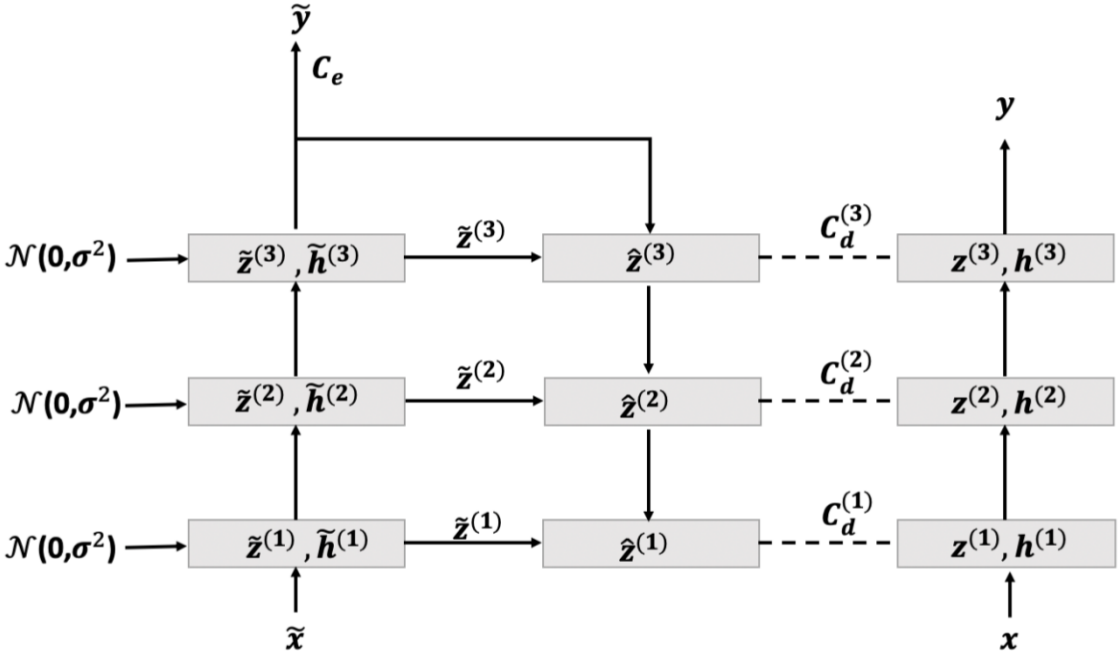


Figure 4.6 An illustration of the ladder network with two encoders on the right side (noiseless) and the left side (noisy) and one decoder in the middle. Encoder: at each layer, $\tilde{\mathbf{z}}^{(K)}$ and $\hat{\mathbf{z}}^{(K)}$ are captured by applying some linear transformation on $\tilde{\boldsymbol{h}}^{(k-1)}$ and $\boldsymbol{h}^{(k-1)}$, respectively (equations 4.10 to 4.13). Decoder: at each level it receives two sets of information, the lateral connection $\tilde{\mathbf{z}}^{(K)}$ and $\boldsymbol{v}^{(k+1)}$ to reconstruct $\hat{\mathbf{z}}^{(K)}$ (equations 4.14 to 4.16). The final objective function is a weighted sum of all cross entropy C_e and the unsupervised reconstruction cost (C_d).

Encoder: Each layer of the encoder is modeled by a linear transformation as shown below:

$$\tilde{\boldsymbol{z}}_{pre}^{(k)} = \boldsymbol{W}^{(k)} \tilde{\boldsymbol{h}}^{(k-1)} \tag{4.10}$$

Where $\boldsymbol{W}^{(k)}$ is the weight matrix between layer $(k-1)$ and layer k and $\widetilde{\boldsymbol{h}}^{(k-1)}$ is the activation at layer $(k-1)$ for $1 \leq k \leq K$. Then, we follow [179] to apply batch normalization to each layer:

$$\tilde{\boldsymbol{z}}^{(k)} = \mathbf{N}_{\boldsymbol{B}}\left(\tilde{\boldsymbol{z}}_{pre}^{(k)}\right) + \boldsymbol{noise} = \frac{\tilde{\boldsymbol{z}}_{pre}^{(k)} - \boldsymbol{\mu}^{(k)}}{\boldsymbol{\sigma}^{(\boldsymbol{k})}} + \boldsymbol{\mathcal{N}}(\mathbf{0}, \boldsymbol{\sigma}^{\mathbf{2}}) \qquad (4.11)$$

The batch normalized form of $\tilde{\boldsymbol{z}}_{pre}^{(k)}$ is calculated by the mean and standard variance from min-batch, $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\sigma}^{(\boldsymbol{k})}$, respectively. Then, we obtain $\tilde{\boldsymbol{z}}^{(k)}$ by adding a Standard Gaussian noise with mean $\mathbf{0}$ and variance. We calculate the activation function of layer k:

$$\widetilde{\boldsymbol{h}}^{(k)} = \phi\left(\boldsymbol{\gamma}^{(k)}\left(\tilde{\boldsymbol{z}}^{(k)} + \boldsymbol{\beta}^{(k)}\right)\right) \qquad (4.12)$$

Where $\boldsymbol{\gamma}^{(k)}$ and $\boldsymbol{\beta}^{(k)}$ denote the trainable scaling and biasing parameters of the nonlinear activation function $\phi(.)$. All mentioned equations formulate the noisy encoder. However, one can find all equations for the noiseless encoder by inserting the noiseless components instead of the noisy ones ($\tilde{\boldsymbol{z}}_{pre}^{(k)} \rightarrow \boldsymbol{z}_{pre}^{(k)}, \tilde{\boldsymbol{z}}^{(k)} \rightarrow \boldsymbol{z}^{(k)},\ \widetilde{\boldsymbol{h}}^{(k)} \rightarrow \boldsymbol{h}^{(k)}$) into the equations and removing the added Standard Gaussian noise from equation (4.11).

The objective of the encoder is to minimize the weighted sum of the supervised cross entropy function. Therefore, for the given $\boldsymbol{x}(m)$ inputs, the cost of matching the noisy output $\widetilde{\boldsymbol{y}}(m)$ to the true target vector $\widetilde{\boldsymbol{y}}_{true}(m)$, is formulated below:

$$C_e = -\frac{1}{M}\sum_{m=1}^{M} logP(\widetilde{\boldsymbol{y}}(m) = \widetilde{\boldsymbol{y}}_{true}(m) |\ \boldsymbol{x}(m)) \qquad (4.13)$$

Decoder: Each layer of the decoder at layer k is responsible for combining the output of the preceding layer, $\hat{\boldsymbol{z}}^{(k+1)}$, and the corresponding output from the noisy encoder, $\tilde{\boldsymbol{z}}^{(k)}$. Then, the reconstructed signal $\hat{\boldsymbol{z}}^{(k)}$ is obtained based on the following equations:

$$\boldsymbol{v}_{pre}^{(k+1)} = \boldsymbol{\Gamma}^{(k)}\hat{\boldsymbol{z}}^{(k+1)} \qquad (4.14)$$

$$\boldsymbol{v}^{(k+1)} = \mathbf{N}_{\boldsymbol{B}}\left(\tilde{\boldsymbol{v}}_{pre}^{(k+1)}\right) = \frac{\tilde{\boldsymbol{v}}_{pre}^{(k+1)} - \boldsymbol{\mu}^{(k+1)}}{\boldsymbol{\sigma}^{(\boldsymbol{k}+\mathbf{1})}} \qquad (4.15)$$

$$\hat{\boldsymbol{z}}^{(k)} = g\left(\tilde{\boldsymbol{z}}^{(k)}, \boldsymbol{v}^{(k+1)}\right) \qquad (4.16)$$

Where $\boldsymbol{\Gamma}^{(k)}$, with the same dimension of $\boldsymbol{W}^{(k)}$on the encoder side, is the weight matrix between layer $(k\ +\ 1)$ and layer k. $\boldsymbol{v}^{(k+1)}$denotes the batch normalized version of the $(k+1)$ projection vector $\boldsymbol{v}_{pre}$. Lastly, a mapping function, $g(.,.)$, is applied on $\tilde{\boldsymbol{z}}(k)$ and $\boldsymbol{v}^{(k)}$ to reconstruct the observation.

The decoder's objective is mitigating the unsupervised reconstruction cost, which is formulated as:

$$C_d = \sum_{k=1}^{K} \lambda_k\, C_d^{(k)} \qquad (4.17)$$

where λ_k is a denoising cost multiplier and

$$C_d^{(k)} = \sum_{m=M+1}^{N} \left\| \frac{\hat{\boldsymbol{z}}(k) - \boldsymbol{\mu}^{(k)}}{\boldsymbol{\sigma}^{(\boldsymbol{k})}} - \boldsymbol{z}(k) \right\|^2 \qquad (4.18)$$

where $\boldsymbol{z}(k)$ is an observation at layer k from the noiseless encoder path.

The two main components of each proposed solution, feature extractor and classifier, have been described in the Chapter 3 and Section 4.1, respectively. To evaluate the performance of each introduced model, we need to provide experimental results.

4.2 Experimental Results

We conduct various experiments to validate the performance of the proposed classifiers for automated BD assessment. We aim to implement a solution to automatically assess a serious mental disorder, BD. Therefore, we propose three BD classification models using the TAVBD corpus [31]. This is a ternary classification task considering three different states of BD including: Mania, Hypo-Mania, and Remission.

Each proposed solution is composed of two major modules: feature extractor and classifier. Hence, we must find the optimum setting for each. First, we start by explaining the experimental setting for the various components of proposed models. Second, we provide experimental results and compare the performance of the proposed models to state-of-the-art approaches.

4.2.1 Feature Extraction

The TAVBD corpus [31] contains 218 video recordings, which enables us to extract three modalities of information including: audio, visual, and textual. We discuss the details of capturing each modality below:

- **Visual representations:** The frames of each video recording are extracted at the frame rate of 30 Hz. Approximately two million frames are collected in total. To focus on facial cues, the faces are cropped, aligned, and saved in *224 × 224* pixel RGB images using the Dlib face detector [117]. Then, to extract visual features of each saved face image, we fine-tuned a VGG-Face model with the FER 2013 dataset. VGG-Face was trained on 2.6 million face images to classify 2622 identities. Therefore, we fine-tuned the pretrained VGG-Face model with FER 2013 dataset, which includes 35887 facial images to recognize various facial expressions. We freeze all VGG-Face layers except the last pooling layer (pool5) and then define a classifier with 512 hidden layers on top of that. Classifier weights are updated during the training of the model with the FER2013 training set including 28,709 images. Then, we employ the fine-tuned model as a feature extractor. We collect all the frame features from the CNN's layer FC6, the first layer after the last pooling layer, with 512 dimensions. We normalize the extracted features per frame and create a sequence of the features for all the frames in each video clip. We feed the sequence to a classifier.

- **Audio representations:** We extract MFCC and eGeMAPs audio features from each audio file. To compute the MFCC features, as we describe in Section 3.2, we use the openSMILE [94] toolkit at the frame level for each normalized audio record. The frame length is set to 60ms with a frame shift of 10ms. Therefore, our LLDs include 13 Mel-frequency cepstral coefficients and 26 dynamic coefficients (delta and double-delta). The eGeMAPs, however, are measured at the speaker turn level which are estimated by an LSTM network [135]. To obtain more meaningful information, we consider three useful parameter sets of eGeMAPs such as energy/amplitude, frequency, and spectral parameter. We collect 3, 6, and 14 descriptors for energy/amplitude, frequency, and spectral parameters, respectively. Then, after applying 4 statistical functions including: min, max, mean, and variance, we end up with 248 audio features, MFCCs and eGeMAPs, in total for each sample.

- **Textual representations :** For the textual features, we capture linguistic and sentiment features through the Suite of Automatic Linguistic Analysis Tools (SALAT) [92]. We pass each subject's transcript to The Simple Natural Language Processing (SiNLP) [141], one of the provided functions by SALAT [92], and it outputs 14 linguistic features based on the content of the text. In addition, the Sentiment Analysis and Cognition Engine (SEANCE) [142] provides sentiment features that are related to four key sentiment indices. Table 3.2 lists all the extracted textual features and their dimensions.

We employ a data-level fusion technique, which means that we fuse all the obtained data from the various modalities before conducting any analysis. After capturing the features and applying the proper pre-processing techniques for each feature set, we implement early-stage data fusion which involves the concatenation of the features in a multimodal stream.

4.2.2 Experimental Setup

In the following sub-sections, we separately provide more details on the classifier design of each proposed solution. All experiments are implemented on NVIDIA GeForce RTX 2080 Ti GPU and our model is implemented on Keras with a Tensorflow backend.

We propose three multi-modal models for the BD classification task. As we are provided with audio/visual recordings along with textual transcripts, we extract features for the audio, visual, and textual modalities as mentioned in Section 4.2.1. In Section 4.2.3, we discuss which feature sets work best for the proposed models. Table 4.1 shows the number of obtained features for each sample in the dataset.

Table 4.1 Number of extracted features for each sample.

Stacked ensemble model for	MFCCs-eGeMAPs	Textual	Visual
BD severity detection	248	262	512

4.2.2.1 Multi-Modal Stacked Ensemble Model

The obtained feature sets contain redundant features, which might: 1) increase the risk of overfitting during training and 2) increase training time exponentially. To avoid these drawbacks, we apply a feature selection method before training the classifier. Researchers often employ two strategies for feature selection: filter-based and wrapper-based method. In the former, features are selected based on their scores in various statistical tests for their correlation with the outcome variable. However, in the latter, we use a subset of features to train a model. We add or remove features from our subset based on the inferences that we draw from the model before we train our next model. We continue this iterative process until we resolve the features that maximize performance. Given that wrapper methods may render the model more prone to overfitting, we decided to explore filter-based methods. As recommended in [80], we use the Analysis of Variance (ANOVA) [180] algorithm for feature selection. ANOVA [180] offers a set of statistical functions to measure the ANOVA F-value between features. Therefore, for each sample feature set, we find the most efficient features based on the importance order of the ANOVA F-value.

As we stated in sub-section 4.1.1.1, the stacked ensemble classifier consists of base learners and a meta learner, which we need to design. Each base learner is a shallow CNN that includes a one-dimensional convolutional layer that is followed with dropout and dense layers. To decide on the number n of base learners, we examine an interval of $\mathrm{n} \in [2, 15]$. Focusing on the three modalities, Table 4.2 presents the accuracy on the development set for the range of base learners.

By considering the trade-off between performance and the time cost, we choose $n = 3$. The meta learner is an MLP network consisting of three dense layers with four dropout layers. Figure 4.7 shows each layer of the proposed classifier.

Table 4.2 BD severity detection: development accuracy for different number of base learners for 50 epochs.

Number of Base Learners	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Accuracy of Dev. (%)	60.5	**63.0**	62.1	58.0	62.0	61.2	56.1	60.2	57.0	61.72	59.1	59.7	62.4	59.8

In sub-section 4.1.1.1, we mentioned that NAS [170] performs hyperparameter optimization using a reinforcement learning algorithm for the base learners and meta learner. As in [170], the controller is a two-layer LSTM, however, we set 64 hidden units on each layer. We use the Adam optimizer [181] with a learning rate of 6×10^{-4} to train the LSTM. The weights are initialized uniformly between -0.08 and 0.08. We generate 200 controller replicas and 3 child replicas to optimize the base learners and meta learner. As the controller samples an architecture and creates a child network, we train the child network for 50 epochs with a batch size of 128.

The hyperparameters tuned by the controller at both levels of the classifier are:

- number of filters $\in$ $[32, 64, 128, 256]$
- dropout rate $\in [0.1, 0.2, 0.3, 0.4, 0.5]$
- activation function $\in [\mathrm{relu}, \mathrm{linear}, \mathrm{tanh}]$

The optimum architecture found by the NAS [170] approach for the base learners and meta learner is shown in Table 4.3. As Table 4.3 demonstrates, there is no batch-normalization unit after the Conv1D layers and the main difference between all Dense layers is the number of hidden units.

The output size of the classifier for both base and meta learner is 3, as we have three different BD states. Figure 4.7 illustrates the architecture of all three base learners and meta learner.

The proposed stacked ensemble model uses the NAS reinforcement learning strategy for hyperparameters tuning [170] and the ANOVA based feature selection scheme [180]. To assess the contribution of each of these techniques to the overall performance, we evaluate the stacked ensemble model without the application of each.

The Unweighted Average Recall (UAR), which was the AVEC 2018 [74] performance metric, and accuracy are the two indicators we use to assess the proposed classification models for BD severity detection. The results in Table 4.4 are achieved on the development dataset. We are unable to assess on the test set as the dataset provider limits the number of assessments on this set. As Table 4.4 depicts, both components contribute significantly to the performance. Based on the results, missing any of the components would degrade the performance of the model. For example, if we remove the ANOVA feature selection and NAS, we obtain a UAR of 53.6%. This weak performance was predictable as removing these two components means that we do not address the issue of redundant features and properly perform hyper-parameter tuning. Adding only one component, either the ANOVA or NAS component, increases the UAR to around 60%, which emphasizes the contribution of each component to the solution’s performance. Finally, employing both ANOVA and NAS boosts the UAR to 64% on the development data.

We implement a classifier based on the discovered architecture shown in Table 4.3. To train the stacked ensemble classifier, the training data of the meta learner must be different from that of the base learners. Accordingly, we use a holdout method to train the base learners on part of the dataset and then train the meta learner on the rest of the data which is not seen before by the base learners.

Table 4.3 Predicted architecture of classifier by NAS.

	Layer	**Number of hidden units**	**Activation function**	**Dropout rate**
Base learner	Conv1D	128	linear	-
	Dropout	-	-	20%
	Dense	256	relu	-
Meta learner	Dense	256	linear	-
	Dropout	-	-	30%
	Dense_1	100	linear	-
	Dense_2	100	linear	-

The base learners are trained with the Adam optimizer [181] with a learning rate of 8×10^{-4} for 50 epochs. The meta learner is trained with a Stochastic Gradient Descent (SGD) optimizer [182] with the learning rate of 2×10^{-3} for 200 epochs. The batch size for both levels of classifiers is set to 64.

Table 4.4 Contribution of two key components of the proposed method on development set.

	NAS	**ANOVA based feature selection**	**UAR (%)**
Ensemble Stacked Model	✗	✓	60.0
	✓	✗	59.3
	✓	✓	64.0
	✗	✗	53.6

4.2.2.2 Multi-Modal Hybrid Model

For the multi-modal hybrid model, the extracted features are forwarded to an LSTM classifier. We design a one-layer LSTM with 100 units. The input is a sequence with a length of 500. As overfitting is a common issue when using an LSTM classifier, we apply a weight regularization technique, L_2 regularization, on the weights of the LSTM nodes. L_2 regularization pushes weights towards zero but does not render them equal to zero. L_2 regularization may deduct a small percentage from the weights at each iteration. However, weights will never reach zero. Therefore, we implement a bias L_2 regularization with a factor of 0.001.

We train and validate the LSTM classifier on the original training and development partitions of the TAVBD corpus [31], which contains 104 and 64 video recording files, respectively. To train the LSTM, we adopt the Adam optimizer [181] with a 0.01 learning rate for 20 epochs. The batch size is set to 128.

4.2.2.3 Multi-Modal Semi-Supervised Model

We model a ternary BD classification task using a semi-supervised model. A defined ladder-based classifier predicts the individuals' mental states using three features. We train and test our model on the TAVBD corpus [31] using the same three extracted feature sets. The feature sets were mentioned in Section 4.2.1.

Designing the ladder network includes two main steps:

- Supervised learning: implementation of an MLP network as the encoder
- Unsupervised learning: implementation of decoder to invert the mappings for each layer of the encoder

As we mentioned in Section 4.1.3.1, both decoder and encoder are MLP networks. To choose the number of hidden units for the dense layer, we follow the suggestion of [176]. Therefore, we design an MLP network with the layer size of $[1000, 500, 250, 250, 250, 10]$. After each dense layer, we apply a batch normalization layer followed with a dropout of 0.4 rate. Then we inject noise to each layer.

The level of the added noise at each layer, which is the standard deviation (σ^2) of the standard gaussian noise, and the denoising cost value, λ, are the two hyperparameters of the proposed model. We optimize them using a grid search approach on the following search space:

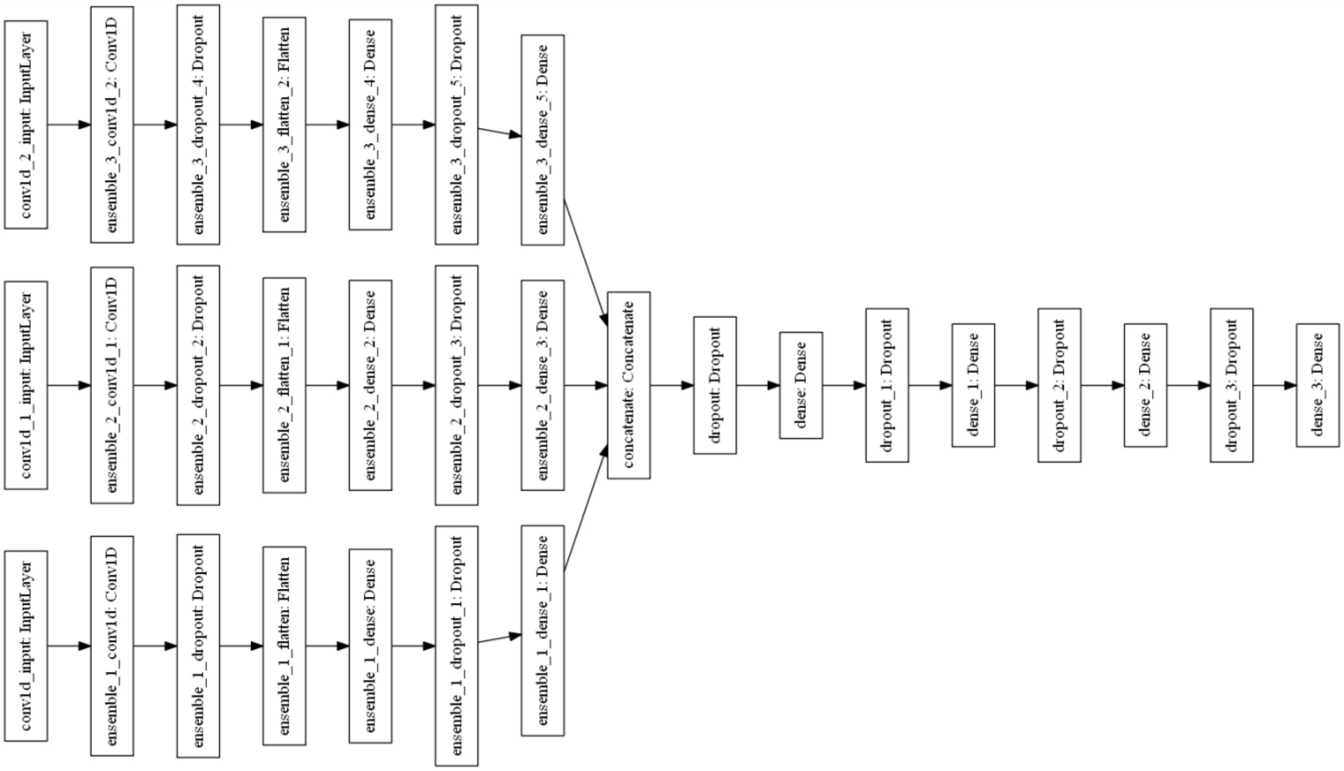

conv1d_2_input: InputLayer
ensemble_3_conv1d_2: Conv1D
ensemble_3_dropout_4: Dropout
ensemble_3_flatten_2: Flatten
ensemble_3_dense_4: Dense
ensemble_3_dropout_5: Dropout
ensemble_3_dense_5: Dense
conv1d_1_input: InputLayer
ensemble_2_conv1d_1: Conv1D
ensemble_2_dropout_2: Dropout
ensemble_2_flatten_1: Flatten
ensemble_2_dense_2: Dense
ensemble_2_dropout_3: Dropout
ensemble_2_dense_3: Dense
concatenate: Concatenate
dropout: Dropout
dense: Dense
dropout_1: Dropout
dense_1: Dense
dropout_2: Dropout
dense_2: Dense
dropout_3: Dropout
dense_3: Dense
conv1d_input: InputLayer
ensemble_1_conv1d: Conv1D
ensemble_1_dropout: Dropout
ensemble_1_flatten: Flatten
ensemble_1_dense: Dense
ensemble_1_dropout_1: Dropout
ensemble_1_dense_1: Dense

Figure 4.7 Architecture of the stacked ensemble classifier for BD severity detection. The predictions of three identical CNN base learners are fed as input to an MLP meta learner

- $\sigma^2 \in \{0.00001, 0.0001, 0.001, 0.01, 0.1\}$
- $\lambda \in \{0.001, 0.01, 0.1, 0.2, 0.5, 1, 2, 5, 10\}$

Based on the result of the grid search, the optimal values are $\sigma^2 = 0.0001$ and $\lambda = 0.1$.

We train the network with the Adam optimizer [181] with a learning rate of 1×10^{-3} for 500 epochs. The batch size is set to 64.

4.2.3 Results

As we mentioned earlier, only the training and development labels are provided for the TAVBD dataset. The labels for the test partition are kept confidential by the dataset provider. Hence, to obtain the results on the test partition, researchers must send their predicted labels to the dataset provider for assessment. Hence, the results on the test partition renders far more objective insights on the performance of the classifier compared to the development partition.

For the stacked ensemble model, we must manipulate the data partitions to train the base learners and meta learner. To report a reliable performance, the training data that is used for the base learners should be different from the training data employed for the meta learner. Therefore, to test the performance of the proposed model on the development data of the TAVBD dataset, we consider 0.6 of the training data, $0.6 \times 104 = 62$ subjects, for the base learners training. The rest is reserved for the meta learner training. Then, the development set, which comprises the data of 64 subjects, is used to validate the performance of the BD classifier. Figure 4.8 illustrates the learning curves for the proposed stacked ensemble classifier. As the figure shows, accuracy for both training and validation sets reached a plateau after 50 epochs. Therefore, we trained the model only for 50 epochs to reduce computational time.

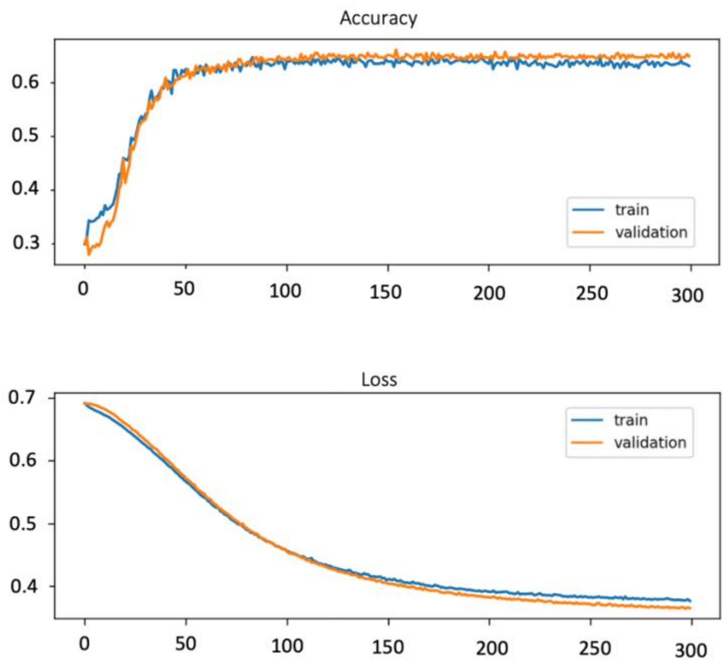


Figure 4.8 Learning curves for the proposed stacked ensemble model. Accuracy and loss curves for both training and validation sets.

To assess the BD classifier on the test partition, the base learners and meta learner are trained on the original training and development sets, respectively.

The datasets for the TAVBD corpus is imbalanced for the training and development sets. Therefore, we used oversampling to address the class imbalance problem by up-sampling the minority class/classes to match the numbers of the majority class/classes. To this end, we utilized a random resampling with replacement strategy.

For the LSTM classifier, we utilized the exact form of the original dataset for each partition. Hence, we train the model on the training dataset which includes 104 video recordings. To validate the performance of proposed model, we used the development set, which includes 60 video recording. The performance of the model is reported on the test set containing 54 recording files.

For the semi-supervised model, we have two sets of labeled and unlabeled data. As we consider the performance of the model for both the development and test sets, there is a need to precisely define a splitting scenario for each partition. After dividing the data into labeled and unlabeled partitions, we apply the proposed model for each set. The setting for each assessment is as follows:

- *Assessment of performance on the development set:* in this scenario, we consider the original development set as our unlabeled set (N = 60). Then, we use the original training samples as our labeled set (M = 104) to train the model.

- *Assessment of performance on the test set:* this time we consider the test set as our unlabeled partition (N = 54) and we train the classifier with the original training samples (M = 104). The original development set, with 60 samples, is used for evaluating the model structure and hyperparameters tuning.

For the TAVBD corpus [31], we are granted a limited number of opportunities (three times) to send the predicted labels of the test partition to the dataset provider for assessment. Hence, we select the best feature set by comparing the performance of the model on the development data. We extract various combinations of audio, visual, and textual features and use them with the proposed models. As the Table 4.5 depicts, the audio and textual modalities produce the highest UAR on the development set for the proposed stacked ensemble and semi-supervised model. However, the visual modality provides the best feature set for the LSTM model.

We make the following observations on the results of Table 4.5:

- **Proposed stacked ensemble model:** The visual or textual modalities alone produce the worst performance (UAR of around 40%). However, the audio modality alone achieves a UAR of 61.23%. The combination of the visual and textual modalities could not improve the performance considerably. Any combination that includes the audio modality increases the UAR on the development data. Considering all three modalities produces a UAR of 62.3%. However, the combination of audio and textual features achieve a UAR of 64.0%. Hence, for the proposed stacked ensemble model, we continue the testing with audio and textual representations. As we found that audio and textual modalities are the best, we need to recalculate the optimum number of base learners for the proposed approach. Table 4.6 shows the results of BD detection using these two modalities on development data. Same as having three modalities, the optimum number of base learners is 3.

- **Proposed LSTM model:** The obtained results demonstrate the superiority of the visual modality over the two others for this model. The textual features alone were unable to capture meaningful cues for the proposed LSTM model (UAR of 45.92%). The audio modality performs better than the textual modality but still worse than the visual modality. Despite the results on the proposed stacked ensemble model, the combination of the audio and textual features surpasses the performance achieved by the visual modality. Therefore, for the proposed LSTM model, we pursue the testing by only using the visual representations.
- **Proposed semi-supervised model:** Similar to the proposed stacked ensemble model, visual features are unable to achieve the best performance in comparison with the other modalities. The textual modality also does not perform well. However, the combination of the audio and textual modalities achieved the best UAR (60.0%) on the development

data. Hence, like the stacked ensemble model, we choose the audio and textual modalities for the BD classification task using the semi-supervised model.

Table 4.5 Comparison of extracting various feature sets of development data for BD classification.

Method	Features			Dev.	
	Visual	Audio	Textual	UAR (%)	Accuracy (%)
Proposed Stacked Ensemble	-	✓	-	61.23	62.1
	✓	-	-	41.8	42.34
	-	-	✓	40.1	41.0
	✓	✓	✓	62.3	63.0
	-	✓	✓	**64.0**	**65.1**
	✓	-	✓	47.7	49.5
	✓	✓	-	60	64.7
Proposed LSTM	✓	-	-	**60.67**	**63.32**
	-	✓	-	57.8	59.0
	-	-	✓	45.92	47.36
	✓	✓	✓	58.43	60.65
	-	✓	✓	58.31	59.98
	✓	-	✓	57.23	58.1
	✓	✓	-	59.1	61.14
Proposed Semi-supervised	✓	-	-	50.3	54.9
	-	✓	-	59.23	62.5
	-	-	✓	49.8	51.43
	✓	✓	✓	57.45	58.0
	-	✓	✓	**60.0**	**63.24**
	✓	-	✓	48.84	52.0
	✓	✓	-	56.25	58.93

Table 4.6 BD severity detection using audio and textual modalities: development accuracy for different number of base learners for 50 epochs.

Number of Base Learners	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Accuracy of Dev. (%)	62.0	**65.1**	57.03	62.6	65.0	62.0	57.4	59.0	64.1	63.2	60.4	60.0	64.0	62.5

After finding the best feature sets for the proposed models, we assess the performance on the test set. Table 4.7 presents the comparison of the BD classification performance between the proposed models and state-of-the- art studies. We list all existing methods for BD detection on the TAVBD corpus [31]. This comparison lists the results on both the development and test sets. However, some researchers did not report performance on the test set. In addition to the UAR and accuracy (introduced metrics in baseline paper) we calculated the mean precision for all proposed models.

The proposed stacked ensemble method outperforms the existing approaches on the test set. Among the existing work, the hierarchical recall model [80] and multi-instance learning [13] have reported the highest UAR, 57.4%. However, the proposed stacked ensemble model has achieved a UAR of 59.3%.

At the present time, this is the best performance on the TAVBD corpus [31] for the test set. Some studies have achieved better performance compared to the proposed model on the development set. However, these studies either do not report test set performance [76], [9] or have achieved inferior results on the test set [13], [80] which indicates model overfitting. For example, in [80], the UAR of 86.7% on the development set is reduced to 57.4% on the test set. Nevertheless, the similar performance for the proposed model on the development and test sets shows that the model does not exhibit a major overfitting problem.

Table 4.7 Comparison of BD classification results that achieved by the proposed methods and existing studies on the same dataset.

Method	Features			Dev.			Test		
	Visual	Audio	Textual	UAR (%)	Accuracy (%)	Precision (%)	UAR (%)	Accuracy (%)	Precision (%)
Proposed Stacked Ensemble	-	✓	✓	64.0	65.1	68.5	**59.3**	**59.3**	**59.2**
Proposed LSTM	✓	-	-	60.67	63.32	66.6	57.4	57.4	57.4
Proposed Semi-supervised	-	✓	✓	60.0	63.24	55.5	53.7	56.9	53.7

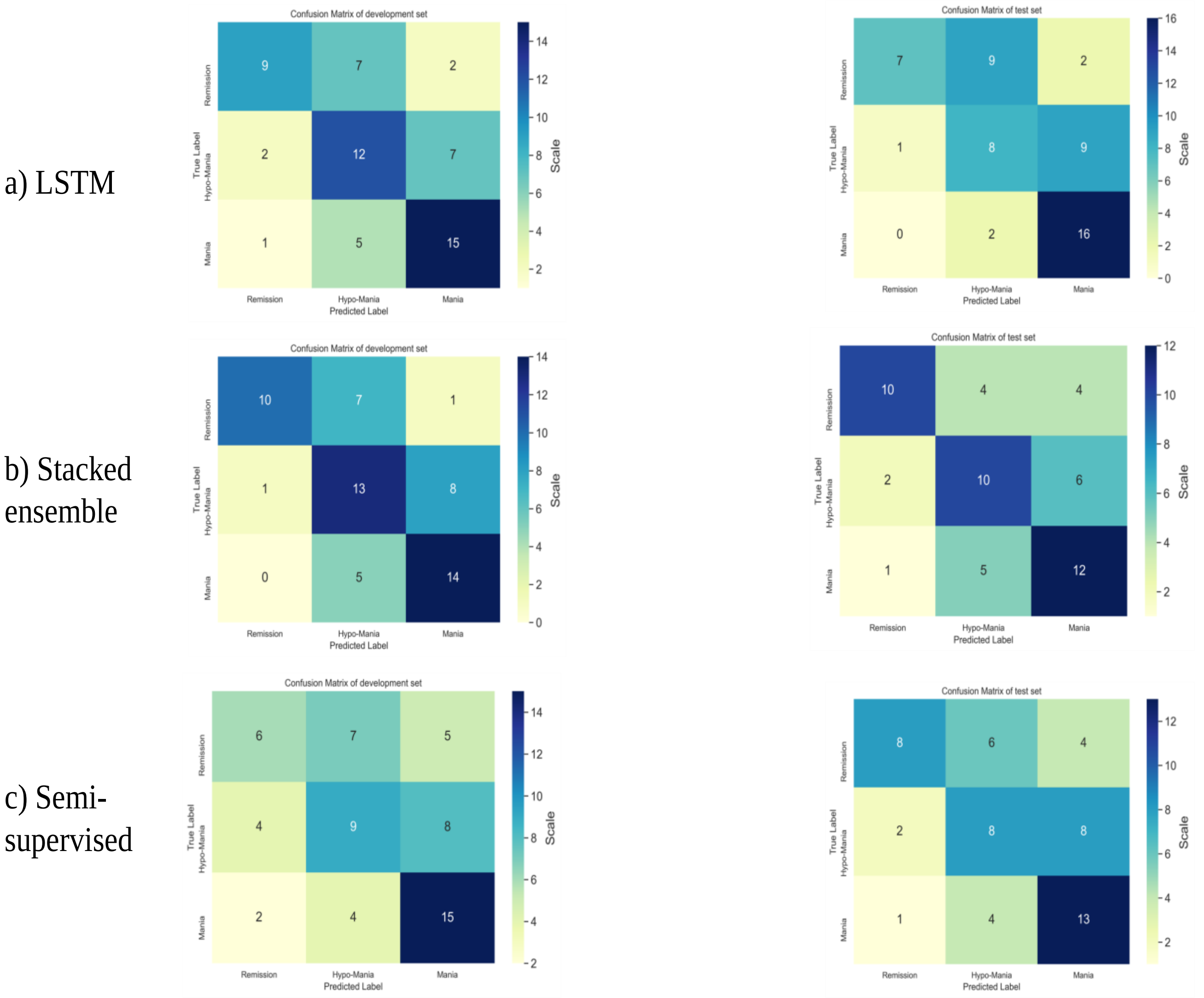
SVMs [74] (Baseline)	✓	✓	-	55.82	-	-	50.0	-	-
ELMs [31]	✓	✓	-	47.3	-	-	-	-	-
Multistream [13]	✓	✓	-	78.3	78.3	-	40.7	40.7	-
IncepLSTM [76]	-	✓	-	65.1	65.0	-	-	-	-
GEWELMs [65]	✓	✓	-	55.0	-	-	48.2	-	-
Hierarchical recall model [80]	✓	✓	✓	**86.7**	**86.7**	-	57.4	57.4	-
Multi-instance learning [66]	-	✓	-	61.6	-	-	57.4	-	-
DNN [9]	✓	✓	✓	70.9	71.07	-	-	-	-
CapsNet [77]	-	✓	-	46.2	-	-	45.5	-	-

The proposed LSTM model has superior performance in comparison to some existing studies, including baseline paper [74], ELMs [31] on both development and test portions. Although some existing models, such as the multi-stream [13] and IncepLSTM [76] reported higher UAR on development set, they either have lower UAR on the test set or did not report performance results on the test data. Therefore, the proposed LSTM model has a promising performance on the test set, however, it produces a lower UAR in comparison to another proposed classifier, the stacked ensemble.

Although its highest UAR on the test data is 59.3%, the proposed semi-supervised model achieves a comparable result to existing state-of-the-art models with a UAR of 53.7%. This work not only produces promising performance, but also establishes the possibility of employing semi-supervised methods, such as the ladder network, for BD classification, especially for datasets with a limited labelled set. Therefore, this study can inform data collectors on how to maximize the usefulness of their published datasets. The presented semi-supervised model’s result motivates investigating this model in more depth. As annotation is a time and cost consuming process, such model could highlight the potential of taking advantage of unlabeled data. Despite the lower UAR of the semi-supervised model, this study focuses on an important aspect that has been ignored by the past studies on BD assessment. A combination of labelled and unlabelled data can alleviate the need to exclusively collect a large and labelled set, given the added complexity and cost associated with this exercise, especially for mental health assessment tasks.

Figure 4.9 displays the confusion matrices for the development and test sets for the three proposed models. We observe that the classes of Hypo-Mania and Mania classified better than the Remission class. The reason could be that the training and development sets contain less samples of the Remission state compared to the two other states. Moreover, we notice from Figure 4.9 that Mania and Hypo-Mania are often confused by the classifiers, which may be attributed to their similar symptoms.

By comparing the confusion matrices for each proposed model, we notice that misclassifications are less frequent for the stacked ensemble model compared to the two other ones, which is expected due to its higher UAR. For the test data, the LSTM model correctly classified 7 samples (out of 18) for the Remission class compared to 8 samples for the semi-supervised model and 9 samples for the stacked ensemble model. We notice that the LSTM model mostly confuses the Remission and the Hypo-Mania classes, which contributed to its relatively higher misclassification rate for this class. However, the Remission samples that are misclassified by the stacked ensemble model are equally predicted to belong to the two other classes. As for the Hypo-Mania class, the number of correctly classified samples is 8, 8, 10 (out of 18) for the LSTM, semi-supervised, and stacked ensemble models, respectively. All classifiers mostly predict the Mania class when they mis-classify the Hypo-Mania samples. The LSTM model performs the best on the Mania class by classifying 16 samples correctly (out of 18) while the semi-supervised and stacked ensemble models correctly classified 13 and 12 samples, respectively. All classifiers performed relatively well for predicting this BD state. This may be due to the pronounced symptoms that are typically displayed by patients in this state which may simplify the classification task. Although the proposed stacked ensemble model performed better for the classification of the Remission and Hypo-Mania states in more misclassifications for Mania class compared to the LSTM and semi-supervised model, it results in more misclassifications for Mania class. In general, it is difficult to draw concrete conclusions based on the results in the confusion matrices due to the small size of the dataset. A larger dataset is required to assess whether our observations would stand. However, our preliminary observations may shed light on the potential strength and weaknesses of the proposed classifiers in distinguishing between the three BD states.


a) LSTM

b) Stacked ensemble

c) Semi-supervised

Figure 4.9 Confusion matrix of the proposed models for BD severity detection : a) LSTM , b) Stacked ensemble, and c) Semi-supervised on the test set (right plot) and development set (left plot).

4.3 Summary

In this chapter, we proposed three models for a BD assessment task. The first is a stacked ensemble model. After providing a brief introduction on ensemble models, we justified how the stacked ensemble model could help in this task. Then, we discussed the details of the proposed stacked ensemble model while considering three modalities. We experimentally assessed the performance of the model for all possible combinations of the three modalities. Based on the results in Table 4.5, we decided to adopt the of audio and textual features which proved to achieve the best performance. The second model is a hybrid CNN-LSTM classifier. For this model, we developed an LSTM model to classify the feature sets we extract from CNN models. The main objective of employing an LSTM model is to take advantage of temporal relations between consecutive data samples. Based on the experimental results of Table 4.5, the visual representation best optimizes the performance of the LSTM model. The last proposed model is a semi-supervised classifier. The reason for choosing a semi-supervised model is to mitigate the issue of having a small dataset and benefit from unlabeled data. This model was trained on the audio and textual modality (as the Table 4.5 demonstrated that the best performance for this model is achieved with these modalities).

After a detailed discussion of each proposed model, we concluded this chapter by providing experimental results and discussion. The achieved results illustrated the superiority of the proposed stacked ensemble model. To the best of our knowledge, this model achieved the highest recorded performance on the TAVBD corpus.

Chapter 5 Depression Regression

Similar to BD detection, conventional diagnosis assessments of depression severity are based on subjective reports from patients or clinicians. Therefore, automated detection frameworks can be beneficial for determining the severity of depression.

The similarities between BD and depression symptoms prompted us to evaluate the applicability of the proposed BD assessment model for depression severity detection. In Chapter 4 we introduced three approaches for BD detection; however, we selected the best model to test on the E-DAIC corpus [96]. The proposed stacked ensemble model achieved the best results to date on the TAVBD corpus [31], which prompted us to test the model for depression severity detection. Apart from the stacked ensemble model's superior performance for BD detection, the limited size of the E-DAIC corpus [96] indicates that an ensemble-based model may be appropriate for this application. Considering that the labels for the E-DAIC [96] are continues numbers, we use a regression model rather than a classification model. Both base and meta learners are regression models for which the optimal architecture must be determined. Figure 5.1 illustrates the proposed depression severity detection model.

We implement three homogeneous MLP networks at the first level and one at the second level in the proposed stacked ensemble model. Later in this chapter, we will discuss the experiments we conducted to determine the optimal number of base learners.

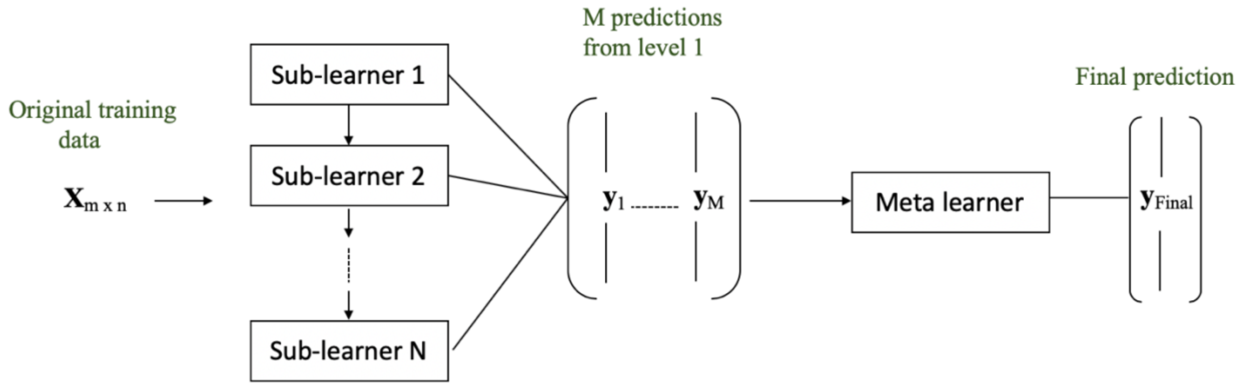


Figure 5.1 The stacked ensemble model. The original training data ($\mathbf{X}_{m \times n}$) is fed to the first level. Then the predictions of sub-learners are considered as the input for the next level. The final prediction is the output of the meta learner.

5.1 Experimental Results

We evaluate the proposed stacked ensemble model's efficacy in detecting depression severity using a publicly available dataset called the E-DAIC corpus [96]. We determine the optimal architecture for the base and meta learners through a series of experiments. Additionally, we conduct tests to determine the optimal feature set for this application. In Chapter 3, we discussed the representations obtained from the E-DAIC corpus [96]. In this chapter, we discuss the feature extraction and regression modules in detail.

5.1.1 Feature Extraction

The E-DAIC corpus [96] is an audio/visual dataset that allows for the capture of a variety of modalities. The extracted features are summarized as follows:

- **Visual representations:** As no raw video file is made publicly available for the E-DAIC [96] corpus, we must rely on the visual features provided by the data provider. The data provider used the OpenFace toolkit to obtain low-level visual features along with their functionals [123]. LLDs, including 17 FAUs, head position and pose, and gaze direction were sampled at 10Hz. The functional properties of LLD visual descriptors are summarized over time using a sliding window of 4s length and a hop size of 1s. In terms of deep visual representations, the pipeline starts by extracting the face region and aligning the face using the OpenFace toolkit [123]. The weights of the two pre-trained models, VGG16 [98] and ResNet-50 [123], were then frozen and the aligned faces were individually fed into each network. Deep cues were captured using the output of the VGG16 network's last connected layer and the ResNet-50's global average pooling layer. The former is a deep vector with 4096 dimensions, while the latter is a 2048-dimensional vector.

- **Audio representations:** For each audio file, we extract MFCC and eGeMAPs audio features. As described in Section 3.2, we compute the MFCC features at the frame level for each normalized audio record using the openSMILE [93] toolkit. The frame duration is set to 60ms with a 10ms frame shift. As a result, our LLDs contain thirteen Mel-frequency cepstral coefficients and twenty-six dynamic coefficients (delta and double-delta). However, the eGeMAPs are estimated at the speaker turn level using an LSTM network [134]. We consider three useful parameter sets of eGeMAPs to obtain more meaningful information: energy/amplitude, frequency, and spectral parameter. We collect 3, 6, and 14 descriptors for the energy/amplitude, frequency, and spectral parameters, respectively. Then, using four statistical functions (min, max, mean, and variance), we end up with a total of 248 audio features, MFCCs, and eGeMAPs for each sample.
- **Textual representations:** We remove all punctuation and non-letter characters from each transcript file during preprocessing. We employ Word2Vec to generate word

vectors using genism [182], a python library. We chose to use a pre-trained model [183] introduced by Google [184] as our dataset is small and Word2Vec models require a large amount of text for training. This model was trained using data from a subset of the Google News dataset (about 100 billion words). Three million words and phrases are represented by 300-dimensional vectors in the model. We obtain the sentence embedding by averaging the embeddings of all the words in each line of the transcription using the pre-trained model. Hence, we consider the same embedding for the entire sentence. We defined a five-word window between the current and predicted words in a sentence.

As with the BD classification approaches, we employ data-level fusion for the proposed depression regression models. All pre-processed data from the various modalities are concatenated as one vector and fed to the regressor.

5.1.2 Experimental Setup

All experiments were conducted on an NVIDIA GeForce RTX 2080 Ti GPU. Our model was developed using a Keras and Tensorflow backend.

As we stated in Section 2.2.3.1, the subject's degree of depression is quantified using a PHQ-8 score in the range [0,24]. To account for bias, we convert the PHQ-8 score labels to floating point numbers prior to training by downscaling by a factor of 25. We follow the steps detailed in Section 4.2.2.1 to design the model. To determine the optimum number n of base learners, we repeat the test using the depression corpus this time. Table 5.1 depicts the results on the development set using the three modalities, which indicates that the optimal number of base learners is $n = 3$.

Table 5.1 Depression severity detection: development CCC for different number of base learners for 50 epochs.

Number of Base Learners	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CCC score of Dev.	0.42	**0.46**	0.44	0.45	0.40	0.39	0.41	0.44	0.41	0.45	0.43	0.38	0.42	0.40

Despite the proposed model for BD detection, the NAS [170] approach performs poorly on the depression corpus. Based on the investigation of [183], when the number of layers and complexity of the architecture is limited, grid search typically performs better than the NAS approach. Given that our model is based on a relatively simple architecture that includes dense layers, NAS may not be the most effective method for hyper-parameter tunning. In fact, Table 5.2 depicts the best CCC results obtained with both hyperparameter tuning methods. These results show the superiority of the grid search method compared to NAS for the proposed model. Therefore, we chose to design the architecture using the grid search approach, which involves an exhaustive search through a manually specified subset of the hyperparameter space. We utilize the GridSearchCV method provided by the scikit-learn open-source machine learning library for Python. As illustrated in Figure 5.2, all base learners and the meta learner are made up of MLP networks. We define a search space for all hyperparameters that the grid search approach will tune:

- number of filters $\in [32, 64, 128, 256, 512]$
- activation function $\in [\mathrm{relu}, \mathrm{linear}, \mathrm{tanh}]$

Table 5.3 shows the optimum base and meta learner architectures selected by the grid search method.

Table 5.2 Comparison of hyperparameter tuning methods on the development data.

Hyperparameter tuning method	NAS	Grid Search
CCC score of Dev.	0.38	**0.46**

Table 5.3 Predicted architecture for regressor by the grid search approach.

	Layer	Number of hidden units	Activation function
Base learner	Dense_1	512	relu
	Dense_2	256	relu
Meta learner	Dense_1	512	relu
	Dense_2	256	relu
	Dense_3	128	relu

Figure 5.2 illustrates the architecture of the proposed stacked ensemble model for depression severity detection. The solution consists of three base learners, each with five layers. After the input layer, there is a dense layer with 512 hidden units and relu activation function (based on the results in Table 5.3). Then, after the flatten layer, the second dense layer with 256 hidden units and relu function is connected to the output layer which is a dense layer with one hidden unit. The architecture of all the base learners is identical. The output of the three base learners is forwarded to the meta learner. The meta learner consists of three dense layers with 512, 265, and 128 hidden units, respectively. The last dense layer of the meta learner produces the final prediction.

We train the proposed model after determining the optimal architecture for the stacked ensemble regressor. We must use entirely different data to train the meta learner than we did for the base learner. As a result, we partitioned the data into two sets, one for training base learners and another for training the meta learner. Both base learners and meta learner are trained using the

Adam optimizer [181] with a learning rate of 1×10^{-4} for 1000 and 2000 epochs, respectively. The batch size is 64 for both levels of regressors.

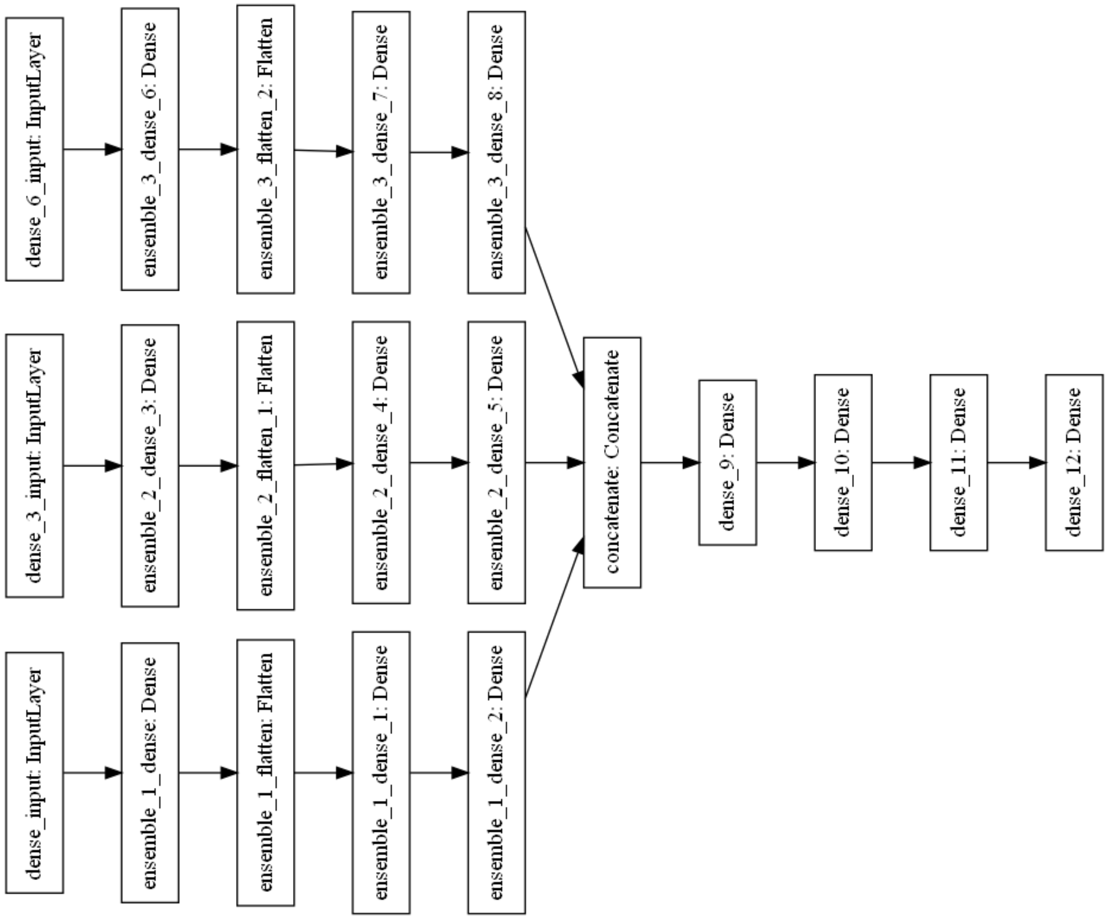

dense_6_input: InputLayer
ensemble_3_dense_6: Dense
ensemble_3_flatten_2: Flatten
ensemble_3_dense_7: Dense
ensemble_3_dense_8: Dense
dense_3_input: InputLayer
ensemble_2_dense_3: Dense
ensemble_2_flatten_1: Flatten
ensemble_2_dense_4: Dense
ensemble_2_dense_5: Dense
dense_input: InputLayer
ensemble_1_dense: Dense
ensemble_1_flatten: Flatten
ensemble_1_dense_1: Dense
ensemble_1_dense_2: Dense
concatenate: Concatenate
dense_9: Dense
dense_10: Dense
dense_11: Dense
dense_12: Dense

Figure 5.2 Architecture of tacked ensemble regressor for depression severity detection. The prediction of three identical MLP base learners are fed as input to an MLP meta learner.

5.1.3 Results

The AVEC2019, which introduced the E-DAIC corpus, recommended the Concordance Correlation Coefficient (CCC) and root mean square error (RMSE) as two metrics [83] to evaluate the models for depression severity detection. To conduct a valid comparison with existing work, researchers using the E-DAIC corpus should report on the same metrics [95].

Given that the proposed stacked ensemble model has two network levels, base and meta, we must manipulate the original data partitioning. The original dataset is divided into three sections:

training, development, and test. To ensure that the performance reported is reliable, the training data used to train base learners should be distinct from the training data used to train meta learners. As a result, we divided the training set into two parts to report on the model's performance on the development set. The base learners are trained using 0.6 of the initial training samples, which equates to $0.6 \times 163 = 98$ samples. The meta learner classifier is trained using the remaining training samples. Hence, the model is validated using 56 samples. However, to evaluate on the test partition, the base learners and meta learner are trained on the original training and development set, respectively.

The classes in the training and development sets are imbalance. Therefore, we use oversampling to correct for the class imbalance by increasing the sample size of the minority class/classes to bring them in line with the majority class/classes. As a result, we augment the minority sample size to match the size of the majority class/classes. To accomplish this, we used a random resampling strategy with replacement.

To determine the optimal feature combination among the audio, visual, and textual modalities, we conduct an experiment comparing the model's performance for each combination. The results are summarised in Table 5.4. As the table depicts, considering the visual modality features alone does not produce the best performance (CCC of 0.38). Considering the textual modality improves the performance compared to the visual features (CCC of 0.41). However, the performance of the audio modality surpasses that of the two other modalities (CCC of 0.46). Even the combination of the visual and textual modality achieves a CCC of 0.40, which is inferior to the result obtained for the audio modality alone. Any combination with the audio representations enhances the results and the highest CCC and lowest RMSE is achieved by considering the audio and textual modalities together. Since employing the audio and textual modalities performs better than considering the three modalities, we need to recalculate the optimum number of base learners for this feature set. Table 5.5 depicts the CCC values when we consider audio and textual features only. Based on the results, the optimum number of base learners is 3.

Table 5.4 Comparison of extracting various feature sets for the development data for depression regression.

Method	Features			Dev.	
	Visual	Audio	Textual	CCC	RMSE
Proposed Stacked Ensemble	-	✓	-	0.46	6.98
	✓	-	-	0.38	7.01
	-	-	✓	0.41	6.53
	✓	✓	✓	0.46	7.2
	-	✓	✓	**0.51**	**5.83**
	✓	-	✓	0.40	6.32
	✓	✓	-	0.44	6.78

The comparison between the proposed models and state-of-the-art studies is shown in Table 5.6. We list all existing methods for detecting depression severity on the E-DAIC corpus [95]. This comparison table summarises the findings from the development and test sets. Certain researchers omitted to report the performance of their approach on the test set.

Table 5.5 Depression severity detection using audio and textual modalities: development CCC for different number of base learners for 50 epochs.

Number of Base Learners	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CCC score of Dev.	0.48	**0.51**	0.49	0.43	0.48	0.45	0.41	0.47	0.43	0.39	0.44	0.41	0.40	0.33

Table 5.6 Comparison of depression regression results that achieved by the proposed methods and existing studies on the same dataset.

Method	**Features**			**Dev.**		**Test**	
	Visual	Audio	Textual	CCC	RMSE	CCC	RMSE
Proposed Stacked Ensemble	-	✓	✓	0.51	5.83	**0.49**	6.20
GRUs [84] (baseline)	✓	✓	✓	0.33	5.03	0.11	6.37
Hierarchical model [98]	✓	✓	✓	0.402	4.94	0.442	5.50
Multi-scale CNN [101]	-	✓	✓	0.466	5.07	0.430	5.91
BERT-CNN [103]	-	✓	✓	0.696	3.86	0.403	6.11
Multi-level attention [104]	✓	✓	✓	-	4.28	-	-

The proposed stacked ensemble model achieved the highest CCC on the test data of the E-DAIC corpus [96]. Despite achieving the highest CCC, the RMSE on the test data is not the lowest one in comparison with [98], [101], and [103]. To investigate the reason, we closely checked all predicted labels in the test data. Comparing them with the true labels, we noticed some outliers (three samples) with big difference with their corresponding true labels. To calculate RMSE, the error is squared and then averaged. Hence, this measure can be drastically affected by outliers, especially for a small sample size. Therefore, the presence of outliers among the samples may have caused the higher value of the RMSE on the test data. However, the CCC metric is more reliable as it incorporates information on the precision and accuracy and is unaffected by changes in location and scale [184]. Additionally, the data provider [83] introduced the CCC metric as the most appropriate metric for reporting model performance. Among the state-of-the-art studies, [97] reported the highest CCC, 0.442, on the test data, and the proposed model improves it to 0.49. Our results may indicate that the proposed model performs better than existing ones given the superior CCC. However, they also suggest that for some inputs, the proposed model produced relatively large errors which increased the RMSE beyond the results achieved by [98], [101], and [103].

5.2 Summary

This chapter discusses the model for detecting depression severity. The proposed stacked ensemble model's acceptable performance on the BD assessment task motivated us to develop one for the depression detection task as well. As a result, we proposed a stacked ensemble regressor that takes audio and textual features into account. The proposed model's objective was to predict depression severity using the E-DAIC corpus. The results of the experiments presented in this chapter established the efficacy of the proposed model. The obtained performance reports the best available CCC on the E-DAIC corpus.

Chapter 6 Conclusion and Future Work

The purpose of this thesis is to propose new models for automatically detecting bipolar and depression severity. We addressed a ternary classification problem using the TAVBD corpus [31] by proposing three multi-modal approaches: the hybrid CNN-LSTM, the stacked ensemble, and the semi-supervised model. Due to the confidentiality of the test labels in this dataset, researchers must submit predicted labels to the data provider for evaluation. We had only three opportunities to submit our results, and three of them yielded promising results. We also investigated an automated depression detection model. Hence, we proposed a stacked ensemble regressor that detects depression severity automatically using the E-DAIC corpus [95].

We employed visual cues obtained from video signals as input for the CNN-LSTM model. As a result, we used the VGG-Face network fine-tuned with the FER2013 to extract spatial features. The proposed hybrid model enables the system to capture both spatial and temporal relations between consecutive frames through CNN and LSTM model. The proposed CNN-LSTM model produced a UAR of 60.67% and 57.4% on development and test datasets respectively.

We used both audio and textual descriptors as input features for the stacked ensemble model for BD detection. For the audio modality, we used MFCCs and eGeMAPs features, while for the textual modality, we used a variety of linguistic and sentiment descriptors. These textual characteristics include the number of unique words, paragraphs, and sentences, the affective norms of arousal and valence, and emotional token words. After selecting features, the extracted set of features is fed into the proposed stacked ensemble model. The classifier is composed of two distinct layers of neural networks: base learners and meta learners. We created three homogeneous CNN networks as our base classifiers and a meta classifier using an MLP network. We optimised all networks using the NAS reinforcement learning strategy. The performance comparison of our novel model to all previous studies on the TAVBD corpus [31] demonstrates the proposed approach's superiority on the test set. This model achieved a UAR of 59.3%, compared to the best reported value of 57.4%. Not only is the proposed approach highly effective, but it also includes useful components such as a reinforcement learning algorithm for hyper-parameter tuning.

The semi-supervised model on BD corpus employs the same feature set as the stacked ensemble classifier. To take advantage of both supervised and unsupervised learning, we designed a ladder network with an encoder and decoder. This work not only achieves comparable performance to the state-of-the-art, but also demonstrates the feasibility of using semi-supervised methods, such as the ladder network, for BD classification, particularly for datasets with a small labelled set. Hence, this study can assist data collectors in maximising the utility of their published datasets. The results of the presented semi-supervised model motivates further investigation of this model. Given that annotation is a time- and resource-intensive process, this model may provide additional opportunities to exploit unlabeled data. The proposed model produced a UAR of 60.0% and 53.7% on the development and test data. Despite the proposed model's lower UAR, this study focuses on an important aspect that has been overlooked in previous research on bipolar disorder detection. Combining labelled and unlabeled data can alleviate the need to collect a large labelled set exclusively, given the added complexity and cost of this exercise, particularly for mental health assessment tasks.

For the depression severity model, we obtained LLD audio features, MFCCs and eGeMAPs, as well as textual cues using Word2Vec model. Following feature selection, we used a stacked ensemble regressor to combine all selected features. For each subject, the proposed regressor determines the severity of depression. The model achieved a CCC of 0.49 on the test data, compared to a CCC of 0.44 on the E-DAIC corpus.

The main limitation of this research is related to the small number of available datasets. This mainly due to the novelty of the topic as limited research has been done in this area and most of the existing approaches have been published recently. Even the publicly available datasets are limited in size given the cost associated with the collection and annotation of the data. This thesis adopted two public datasets, TAVBD [31] and E-DAIC [96]. Both are relatively small.

The proposed research can be further extended as follows:

- In this thesis, we implemented data-level fusion to combine the data from the available modalities. However, as future work, we would like investigate additional fusion techniques including decision-level fusion and hybrid fusion (combination of data and decision level technique).
- We are interested in testing the proposed models with additional datasets to further assess their ability to generalize. Moreover, we would like to explore datasets that with data pertaining to subjects with different cultural backgrounds and languages. Given that such datasets do not exist, we are interested in contributing to their collection. However, such effort requires a close collaboration with clinicians.
- Given the size limitations of existing datasets, we are interested in comparing a variety of data augmentation techniques, in particular those that are based on generative adversarial network to attempt to further boost the performance of the proposed models.
- In this thesis we focused on two common mental disorders, bipolar and depression. We are interested in investigating additional mental disorders (such as autism). Automated autism detection has progressed recently, but there still a lot of potential for improving on existing methods.

- We have explored supervised and semi-supervised learning methods. In the future, we would like to investigate the performance of unsupervised methods as well. Given that we often face difficulties procuring annotated datasets for healthcare applications, implementing unsupervised methods might allow us to overcome this challenge.
- We are interested in realizing a full system that encompasses the proposed approaches. Such system would be employed by clinicians to facilitate or automate mental disorder assessment. We presented a possible vision for such system in Chapter 1.

References

[1] W. H. Organization, “Mental Health,” 2020. [Online]. Available: https://www.who.int/health-topics/mental-health#tab=tab_2.

[2] J. Lobbestael, M. Leurgans, and A. Arntz, “Inter-rater reliability of the Structured Clinical Interview for DSM-IV Axis I Disorders (SCID I) and Axis II Disorders (SCID II),” *Clin. Psychol. Psychother.*, vol. 18, no. 1, pp. 75–79, 2011.

[3] I. E. Bauer, J. C. Soares, S. Selek, and T. D. Meyer, “The Link between Refractoriness and Neuroprogression in Treatment-Resistant Bipolar Disorder,” *Mod. Trends Pharmacopsychiatry*, vol. 31, pp. 10–26, 2017.

[4] P. B. Mitchell, G. M. Goodwin, G. F. Johnson, and R. M. A. Hirschfeld, “Diagnostic guidelines for bipolar depression: A probabilistic approach,” *Bipolar Disord.*, vol. 10, no. 1 PART 2, pp. 144–152, 2008.

[5] W. H. Organization, “Depression,” 2020. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/depression.

[6] R. C. Kessler *et al.*, “The epidemiology of major depressive disorder,” *Evidence-Based Eye Care*, vol. 4, no. 4, pp. 186–187, 2003.

[7] A. M. Kilbourne, D. Goodrich, D. J. Miklowitz, K. Austin, E. P. Post, and M. S. Bauer, “Characteristics of patients with bipolar disorder managed in VA primary care or specialty mental health care settings,” *Psychiatr. Serv.*, vol. 61, no. 5, pp. 500–507, 2010.

[8] S. Reilly, C. Planner, M. Hann, D. Reeves, I. Nazareth, and H. Lester, “The role of primary care in service provision for people with severe mental illness in the United Kingdom.,” *PLoS One*, vol. 7, no. 5, 2012.

[9] Z. Zhang, W. Lin, M. Liu, and M. Mahmoud, “Multimodal Deep Learning Framework for Mental Disorder Recognition,” *IEEE Int. Conf. Autom. Face Gesture Recognit.*, pp. 1–7, 2020.

[10] Z. Zhang, F. Ringeval, B. Dong, E. Coutinho, and E. Marchi, “Enhanced semi-supervised learning for multimodal emotion recognition,” *Icassp 2016*, pp. 5185–5189, 2016.

[11] M. Faurholt-Jepsen *et al.*, “Voice analysis as an objective state marker in bipolar disorder,” *Transl. Psychiatry*, vol. 6, no. 7, p. e856, 2016.

[12] S. Khorram, J. Gideon, M. McInnis, and E. M. Provost, “Recognition of depression in bipolar disorder: Leveraging cohort and person-specific knowledge,” *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*, vol. 08-12-Sept, pp. 1215–1219, 2016.

[13] L. Yang, D. Jiang, Y. Li, M. C. Oveneke, H. Chen, and H. Sahli, "Bipolar disorder recognition with histogram features of arousal and body gestures," *AVEC 2018 - Proc. 2018 Audio/Visual Emot. Chall. Work. co-located with MM 2018*, pp. 15–21, 2018.

[14] A. Pampouchidou *et al.*, "Automatic Assessment of Depression Based on Visual Cues: A Systematic Review," *IEEE Trans. Affect. Comput.*, vol. 10, no. 4, pp. 445–470, 2019.

[15] L. S. A. Low, N. C. Maddage, M. Lech, L. Sheeber, and N. Allen, "Influence of acoustic low-level descriptors in the detection of clinical depression in adolescents," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, no. April, pp. 5154–5157, 2010.

[16] N. J. Scott, R. S. S. Kramer, A. L. Jones, and R. Ward, "Facial cues to depressive symptoms and their associated personality attributions," *Psychiatry Res.*, vol. 208, no. 1, pp. 47–53, 2013.

[17] C. Sobin and H. A. Sackeim, "Psychomotor symptoms of depression," *Am. J. Psychiatry*, vol. 154, no. 1, pp. 4–17, 1997.

[18] D. Schrijvers, W. Hulstijn, and B. G. C. Sabbe, "Psychomotor symptoms in depression: A diagnostic, pathophysiological and therapeutic tool," *J. Affect. Disord.*, vol. 109, no. 1–2, pp. 1–20, 2008.

[19] P. M. Niedenthal, L. W. Barsalou, P. Winkielman, S. Krauth-Gruber, and F. Ric, "Embodiment in attitudes, social perception, and emotion," *Personal. Soc. Psychol. Rev.*, vol. 9, no. 3, pp. 184–211, 2005.

[20] M. Zafi Sherhan Shah, "Automated Screening Methods for Mental and Neuro-developmental Disorders," no. September, 2018.

[21] F. Colom *et al.*, "A RATING SCALE FOR DEPRESSION," *Br. J. Psychiatry*, vol. 194, no. 3, pp. 260–265, 2009.

[22] R. C. Young, J. T. Biggs, V. E. Ziegler, and D. A. Meyer, "A Rating Scale for Mania," *Br. J. Psychiatry*, vol. 133, pp. 429–435, 1978.

[23] K. Kroenke, T. W. Strine, R. L. Spitzer, J. B. W. Williams, J. T. Berry, and A. H. Mokdad, "The PHQ-8 as a measure of current depression in the general population," *J. Affect. Disord.*, vol. 114, no. 1–3, pp. 163–173, 2009.

[24] Z. S. Syed, J. Schroeter, K. Sidorov, and D. Marshall, "Computational paralinguistics: Automatic assessment of emotions, mood, and behavioural state from acoustics of speech," *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*, vol. 2018-Septe, no. September 2018, pp. 511–515, 2018.

[25] L. R. Snowden, "Bias in mental health assessment and intervention: Theory and evidence," *Am. J. Public Health*, vol. 93, no. 2, pp. 239–243, 2003.

[26] D. M. Low, K. H. Bentley, and S. S. Ghosh, "Automated assessment of psychiatric disorders

using speech: A systematic review," *Laryngoscope Investig. Otolaryngol.*, vol. 5, no. 1, pp. 96–116, 2020.

[27] M. Valstar, "Automatic behaviour understanding in medicine," *RFMIR 2014 - Proc. 2014 ACM Roadmapping Futur. Multimodal Interact. Res. Incl. Bus. Oppor. Challenges, Co-located with ICMI 2014*, pp. 57–60, 2014.

[28] A. Vinciarelli, M. Pantic, H. Bourlard, and A. Pentland, "Social signals, their function, and automatic analysis: A survey," *ICMI'08 Proc. 10th Int. Conf. Multimodal Interfaces*, pp. 61–68, 2008.

[29] F. Ringeval *et al.*, "AVEC 2017 - Real-life Depression , and Affect Recognition Workshop and Challenge To cite this version : HAL Id : hal-02080874 AVEC 2017 – Real-life Depression , and A ect Recognition Workshop and Challenge," 2019.

[30] Y. Li, L. Yang, H. Chen, D. Jiang, and H. Sahli, "Audio Visual Multimodal Classification of Bipolar Disorder Episodes," *2019 8th Int. Conf. Affect. Comput. Intell. Interact. Work. Demos, ACIIW 2019*, pp. 115–120, 2019.

[31] E. Ciftci, H. Kaya, H. Gulec, and A. A. Salah, "The Turkish Audio-Visual Bipolar Disorder Corpus," *2018 1st Asian Conf. Affect. Comput. Intell. Interact. ACII Asia 2018*, 2018.

[32] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM J. Res. Dev.*, vol. 44, no. 1–2, pp. 207–219, 1959.

[33] D. Michie, "'Memo' Functions and Machine Learning," *Nature*, vol. 218, no. 5136, pp. 19–22, 1968.

[34] M. M. T. Carbonel, G. jaime, Michalski, S.Ryszard, "An Overview of Machine Learning." p. 23, 1983.

[35] L. Vinet and A. Zhedanov, *Machine Learning for Audio, image, and video analysis*, vol. 44, no. 8. 2011.

[36] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets Geoffrey," *Neural Comput.*, vol. 18, pp. 1527–1554, 2006.

[37] G. E. Hinton, "Deep belief networks," *http://www.scholarpedia.org/article/Deep_belief_networks*, 2009. .

[38] A. J. Holden *et al.*, "Reducing the Dimensionality of Data with Neural Networks," 2006.

[39] I. G. and Y. B. and A. Courville, *Deep Learning*. MIT Press, 2016.

[40] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[41] Z.-H. Zhou, "Ensemble Learning," *Scholarpedia*, vol. 4, p. 2776, 1990.

[42] K. Yun, A. Huyen, and T. Lu, “Deep neural networks for pattern recognition,” *Adv. Pattern Recognit. Res.*, pp. 49–79, 2018.

[43] M. Chen, S. Mao, and Y. Liu, “Big data: A survey,” *Mob. Networks Appl.*, vol. 19, no. 2, pp. 171–209, 2014.

[44] A. B. R. Shatte, D. M. Hutchinson, and S. J. Teague, “Machine learning in mental health: A scoping review of methods and applications,” *Psychol. Med.*, vol. 49, no. 9, pp. 1426–1448, 2019.

[45] A. Sano *et al.*, “Recognizing academic performance, sleep quality, stress level, and mental health using personality traits, wearable sensors and mobile phones,” *2015 IEEE 12th Int. Conf. Wearable Implant. Body Sens. Networks, BSN 2015*, 2015.

[46] M. G. R. Alam, S. F. Abedin, M. Al Ameen, and C. S. Hong, “Web of objects based ambient assisted living framework for emergency psychiatric state prediction,” *Sensors (Switzerland)*, vol. 16, no. 9, 2016.

[47] S. Klöppel *et al.*, “Applying automated MR-based diagnostic methods to the memory clinic: A prospective study,” *J. Alzheimer’s Dis.*, vol. 47, no. 4, pp. 939–954, 2015.

[48] A. König *et al.*, “Automatic speech analysis for the assessment of patients with predementia and Alzheimer’s disease,” *Alzheimer’s Dement. Diagnosis, Assess. Dis. Monit.*, vol. 1, no. 1, pp. 112–124, 2015.

[49] V. Mitra, A. Kathol, E. Shriberg, C. Richey, and M. Graciarena, “The SRI AVEC-2014 evaluation system,” *AVEC 2014 - Proc. 4th Int. Work. Audio/Visual Emot. Challenge, Work. MM 2014*, pp. 93–101, 2014.

[50] B. Hao, L. Li, R. Gao, A. Li, and T. Zhu, “Sensing subjective well-being from social media,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8610 LNCS, pp. 324–335, 2014.

[51] B. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2012.

[52] G. E. Dahl, D. Yu, L. Deng, and A. Acero, “Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition,” *IEEE Trans. Audio, Speech Lang. Process.*, vol. 20, no. 1, pp. 30–42, 2012.

[53] D. H. Kim, W. J. Baddar, J. Jang, and Y. M. Ro, “Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition,” *IEEE Trans. Affect. Comput.*, vol. 10, no. 2, pp. 223–236, 2019.

[54] W. S. Chu, F. De La Torre, and J. F. Cohn, “Learning Spatial and Temporal Cues for Multi-Label Facial Action Unit Detection,” *Proc. - 12th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2017 - 1st Int. Work. Adapt. Shot Learn. Gesture Underst. Prod. ASL4GUP 2017, Biometrics Wild, Bwild 2017, Heteroge*, pp. 25–32, 2017.

[55] K. Y. Huang, C. H. Wu, and M. H. Su, “Attention-based convolutional neural network and long Short-term memory for Short-term detection of mood disorders based on elicited speech responses,” *Pattern Recognit.*, vol. 88, pp. 668–678, 2019.

[56] X. Ma, H. Yang, Q. Chen, D. Huang, and Y. Wang, “DepAudioNet: An efficient deep model for audio based depression classification,” *AVEC 2016 - Proc. 6th Int. Work. Audio/Visual Emot. Challenge, co-located with ACM Multimed. 2016*, pp. 35–42, 2016.

[57] J. Gratch *et al.*, “The distress analysis interview corpus of human and computer interviews,” *Proc. 9th Int. Conf. Lang. Resour. Eval. Lr. 2014*, pp. 3123–3128, 2014.

[58] C. Zhang and Yunqian Ma, *Ensemble Machine Learning*. 2012.

[59] S. Corchs, E. Fersini, and F. Gasparini, “Ensemble learning on visual and textual data for social image emotion classification,” *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 8, pp. 2057–2070, 2019.

[60] T. Dissanayake, Y. Rajapaksha, R. Ragel, and I. Nawinne, “An ensemble learning approach for electrocardiogram sensor based human emotion recognition,” *Sensors (Switzerland)*, vol. 19, no. 20, pp. 1–24, 2019.

[61] H. Jiang *et al.*, “Detecting Depression Using an Ensemble Logistic Regression Model Based on Multiple Speech Features,” *Comput. Math. Methods Med.*, vol. 2018, 2018.

[62] S. Balani and M. De Choudhury, “Detecting and characterizing mental health related self-disclosure in social media,” *Conf. Hum. Factors Comput. Syst. - Proc.*, vol. 18, pp. 1373–1378, 2015.

[63] M. M. Aldarwish and H. F. Ahmad, “Predicting Depression Levels Using Social Media Posts,” *Proc. - 2017 IEEE 13th Int. Symp. Auton. Decentralized Syst. ISADS 2017*, pp. 277–280, 2017.

[64] V. Leiva and A. Freire, “Towards suicide prevention: Early detection of depression on social media,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10673 LNCS, pp. 428–436, 2017.

[65] Z. S. Syed, K. Sidorov, and D. Marshall, “Automated Screening for Bipolar Disorder from Audio/Visual Modalities,” pp. 39–45, 2018.

[66] Z. Ren, J. Han, N. Cummins, Q. Kong, M. D. Plumbley, and B. W. Schuller, “Multi-instance Learning for Bipolar Disorder Diagnosis using Weakly Labelled Speech Data,” no. November, pp. 79–83, 2019.

[67] A. Kumar, A. Sharma, and A. Arora, “Anxious Depression Prediction in Real-time Social Data,” *SSRN Electron. J.*, vol. 2019, pp. 1–7, 2019.

[68] A. U. Hassan, J. Hussain, M. Hussain, M. Sadiq, and S. Lee, “Sentiment analysis of social networking sites (SNS) data using machine learning approach for the measurement of

depression," *Int. Conf. Inf. Commun. Technol. Converg. ICT Converg. Technol. Lead. Fourth Ind. Revolution, ICTC 2017*, vol. 2017-Decem, pp. 138–140, 2017.

[69] K. Al-Jabery, T. Obafemi-Ajayi, G. R. Olbricht, T. N. Takahashi, S. Kanne, and D. Wunsch, "Ensemble statistical and subspace clustering model for analysis of autism spectrum disorder phenotypes," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2016-Octob, pp. 3329–3333, 2016.

[70] A. Naghavi, T. Teismann, Z. Asgari, M. R. Mohebbian, M. Mansourian, and M. Á. Mañanas, "Accurate diagnosis of suicide ideation/behavior using robust ensemble machine learning: A university student population in the middle east and north africa (mena) region," *Diagnostics*, vol. 10, no. 11, 2020.

[71] S. Abdullah, M. Matthews, E. Frank, G. Doherty, G. Gay, and T. Choudhury, "Automatic detection of social rhythms in bipolar disorder," *J. Am. Med. Informatics Assoc.*, vol. 23, no. 3, pp. 538–543, 2016.

[72] M. Faurholt-Jepsen *et al.*, "Voice analysis as an objective state marker in bipolar disorder," *Transl. Psychiatry*, vol. 6, no. May, p. e856, 2016.

[73] B. Schuller, M. Valstar, F. Eyben, G. McKeown, R. Cowie, and M. Pantic, "AVEC 2011 - The first international audio/visual emotion challenge," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6975 LNCS, no. PART 2, pp. 415–424, 2011.

[74] F. Ringeval *et al.*, "AVEC 2018 workshop and challenge: Bipolar disorder and cross-cultural affect recognition," *Proc. 2018 Audio/Visual Emot. Chall. Work.*, pp. 3–13, 2018.

[75] F. Ringeval *et al.*, "AVEC 2018 Workshop and Challenge: Bipolar disorder and cross-cultural affect recognition," *AVEC 2018 - Proc. 2018 Audio/Visual Emot. Chall. Work. co-located with MM 2018*, pp. 3–13, 2018.

[76] Z. Du, W. Li, D. Huang, and Y. Wang, "Bipolar Disorder Recognition via Multi-scale Discriminative Audio Temporal Representation," pp. 23–30, 2018.

[77] S. Amiriparian *et al.*, "Audio-based Recognition of Bipolar Disorder Utilising Capsule Networks," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2019-July, no. November, 2019.

[78] R. A. Rasmussen and S. S. Bohn, "Dynamicroutingbetween capsules," *Appl. Biosaf.*, vol. 22, no. 4, pp. 185–186, 2017.

[79] F. Ringeval, A. Sonderegger, J. Sauer, and D. Lalanne, "Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions," *2013 10th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2013*, no. i, 2013.

[80] S. Li, B. Cai, Y. Zhao, X. Xing, Z. He, and W. Fan, "Multi-modality Hierarchical Recall based on GBDTs for Bipolar Disorder Classification," pp. 31–37, 2018.

[81] Q. Le and T. Mikolov, “Distributed Representations of Sentences and Documents,” *Print on Demand*, vol. 9, no. 2, p. 42, 2003.

[82] M. Valstar *et al.*, “AVEC 2014 – 3D Dimensional Affect and Depression Recognition Challenge,” pp. 3–10, 2014.

[83] M. Valstar *et al.*, “AVEC 2016 - Depression, mood, and emotion recognition workshop and challenge,” *AVEC 2016 - Proc. 6th Int. Work. Audio/Visual Emot. Challenge, co-located with ACM Multimed. 2016*, pp. 3–10, 2016.

[84] F. Ringeval *et al.*, “AVEC 2019 workshop and challenge: State-of-mind, detecting depression with ai, and cross-cultural affect recognition,” *AVEC 2019 - Proc. 9th Int. Audio/Visual Emot. Chall. Work. co-located with MM 2019*, no. Avec, pp. 3–12, 2019.

[85] A. Beck, R. Steer, R. Ball, and W. Ranieri, “Comparison of Beck Depression 1 in Psychiatric Inventories -1A and - Outpatients,” *J. Pers. Assess.*, vol. 67, no. 3, pp. 588–597, 1996.

[86] H. P. Espinosa, H. J. Escalante, L. Villaseñor-Pineda, M. Montes-Y-gómez, D. Pinto-Avedaño, and V. Reyes-Meza, “Fusing affective dimensions and audio-visual features from segmented video for depression recognition INAOE-BUAP’s participation at AVEC’14 Challenge,” *AVEC 2014 - Proc. 4th Int. Work. Audio/Visual Emot. Challenge, Work. MM 2014*, no. November, pp. 49–55, 2014.

[87] A. Jan, H. Meng, Y. F. A. Gaus, F. Zhang, and S. Turabzadeh, “Automatic depression scale prediction using facial expression dynamics and regression,” *AVEC 2014 - Proc. 4th Int. Work. Audio/Visual Emot. Challenge, Work. MM 2014*, no. August, pp. 73–80, 2014.

[88] M. Nasir, A. Jati, P. G. Shivakumar, S. N. Chakravarthula, and P. Georgiou, “Multimodal and multiresolution depression detection from speech and facial landmark features,” *AVEC 2016 - Proc. 6th Int. Work. Audio/Visual Emot. Challenge, co-located with ACM Multimed. 2016*, no. October, pp. 43–50, 2016.

[89] L. Yang, D. Jiang, L. He, E. Pei, M. C. Oveneke, and H. Sahli, “Decision tree based depression classification from audio video and language information,” *AVEC 2016 - Proc. 6th Int. Work. Audio/Visual Emot. Challenge, co-located with ACM Multimed. 2016*, no. June 2018, pp. 89–96, 2016.

[90] J. R. Williamson *et al.*, “Detecting depression using vocal, facial and semantic communication cues,” *AVEC 2016 - Proc. 6th Int. Work. Audio/Visual Emot. Challenge, co-located with ACM Multimed. 2016*, pp. 11–18, 2016.

[91] T. Dang *et al.*, “Investigating word affect features and fusion of probabilistic predictions incorporating uncertainty in AVEC 2017,” *AVEC 2017 - Proc. 7th Annu. Work. Audio/Visual Emot. Challenge, co-located with MM 2017*, no. October, pp. 27–35, 2017.

[92] K. KYLE, “The suite of linguistic analysis tools (salat),” 2017. [Online]. Available: https://www.linguisticanalysistools.org/.

[93] L. Yang, E. Pei, D. Jiang, M. C. Oveneke, X. Xia, and H. Sahli, “Multimodal measurement of depression using deep learning models,” *AVEC 2017 - Proc. 7th Annu. Work. Audio/Visual Emot. Challenge, co-located with MM 2017*, no. January 2018, pp. 53–54, 2017.

[94] F. Eyben, F. Weninger, F. Gross, and B. Schuller, “Recent developments in openSMILE, the munich open-source multimedia feature extractor,” *MM 2013 - Proc. 2013 ACM Multimed. Conf.*, no. May, pp. 835–838, 2013.

[95] B. Sun *et al.*, “A random forest regression method with selected-text feature for depression assessment,” *AVEC 2017 - Proc. 7th Annu. Work. Audio/Visual Emot. Challenge, co-located with MM 2017*, pp. 61–68, 2017.

[96] D. DeVault *et al.*, “SimSensei kiosk: A virtual human interviewer for healthcare decision support,” *13th Int. Conf. Auton. Agents Multiagent Syst. AAMAS 2014*, vol. 2, no. 1, pp. 1061–1068, 2014.

[97] C. A. E. Nickerson, “A Note On " A Concordance Correlation Coefficient to Evaluate Reproducibility " Published by : International Biometric Society Stable URL : http://www.jstor.org/stable/2533516 REFERENCES Linked references are available on JSTOR for this article : You may,” *Biometrics*, vol. 53, no. 4, pp. 1503–1507, 2016.

[98] S. Yin, C. Liang, H. Ding, and S. Wang, “A multi-modal hierarchical recurrent neural network for depression detection,” *AVEC 2019 - Proc. 9th Int. Audio/Visual Emot. Chall. Work. co-located with MM 2019*, pp. 65–71, 2019.

[99] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[100] K. W. G. Huang, Z. Liu, L. Maaten, “Densely Connected Convolutional Networks Gao,” *Am. J. Vet. Res.*, vol. 39, no. 9, pp. 1442–1446, 2017.

[101] W. Fan, Z. He, X. Xing, B. Cai, and W. Lu, “Multi-modality depression detection via multi-scale temporal dilated CNNs,” *AVEC 2019 - Proc. 9th Int. Audio/Visual Emot. Chall. Work. co-located with MM 2019*, pp. 73–80, 2019.

[102] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, no. Mlm, pp. 4171–4186, 2019.

[103] M. R. Makiuchi, T. Warnita, K. Uto, and K. Shinoda, “Multimodal fusion of BERT-CNN and gated CNN representations for depression detection,” *AVEC 2019 - Proc. 9th Int. Audio/Visual Emot. Chall. Work. co-located with MM 2019*, pp. 55–63, 2019.

[104] A. Ray, S. Kumar, R. Reddy, P. Mukherjee, and R. Garg, “Multi-level attention network using text, audio and video for depression prediction,” *AVEC 2019 - Proc. 9th Int.*

Audio/Visual Emot. Chall. Work. co-located with MM 2019, pp. 81–88, 2019.

[105] D. Cer *et al.*, “Universal sentence encoder for English,” *EMNLP 2018 - Conf. Empir. Methods Nat. Lang. Process. Syst. Demonstr. Proc.*, pp. 169–174, 2018.

[106] A. Vinciarelli, M. Pantic, and H. Bourlard, “Social signal processing: Survey of an emerging domain,” *Image Vis. Comput.*, vol. 27, no. 12, pp. 1743–1759, 2009.

[107] J. F. Cohn and F. De La Torre, “Automated Face Analysis for A ff ective Computing,” pp. 131–150, 2014.

[108] J. M. Girard, J. F. Cohn, M. H. Mahoor, S. Mavadati, and D. P. Rosenwald, “Social risk and depression: Evidence from manual and automatic facial expression analysis,” *2013 10th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2013*, 2013.

[109] M. D. Samad, N. DIawara, J. L. Bobzien, J. W. Harrington, M. A. Witherow, and K. M. Iftekharuddin, “A Feasibility Study of Autism Behavioral Markers in Spontaneous Facial, Visual, and Hand Movement Response Data,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 2, pp. 353–361, 2018.

[110] T. Tron, A. Peled, A. Grinsphoon, and D. Weinshall, “Automated facial expressions analysis in schizophrenia: A continuous dynamic approach,” *Commun. Comput. Inf. Sci.*, vol. 604, pp. 72–81, 2016.

[111] B. Derntl, E. M. Seidel, I. Kryspin-Exner, A. Hasmann, and M. Dobmeier, “Facial emotion recognition in patients with bipolar I and bipolar II disorder,” *Br. J. Clin. Psychol.*, vol. 48, no. 4, pp. 363–375, 2009.

[112] B. C. Ko, “A brief review of facial emotion recognition based on visual information,” *Sensors (Switzerland)*, vol. 18, no. 2, 2018.

[113] and A. T. David Bau∗, Bolei Zhou∗, Aditya Khosla, Aude Oliva, “Network Dissection: Quantifying Interpretability of Deep Visual Representation Seminar Report,” 2018.

[114] D. Bau *et al.*, “GaN dissection: Visualizing and understanding generative adversarial networks,” *7th Int. Conf. Learn. Represent. ICLR 2019*, 2019.

[115] R. Walecki, O. Rudovic, B. Schuller, V. Pavlovic, and M. Pantic, “Deep Structured Learning for Facial Expression Intensity Estimation,” *Cvpr*, 2017.

[116] T. Surasak, I. Takahiro, C. H. Cheng, C. E. Wang, and P. Y. Sheng, “Histogram of oriented gradients for human detection in video,” *Proc. 2018 5th Int. Conf. Bus. Ind. Res. Smart Technol. Next Gener. Information, Eng. Bus. Soc. Sci. ICBIR 2018*, pp. 172–176, 2018.

[117] D. E. King, “Dlib-ml: A machine learning toolkit,” *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, 2009.

[118] R. Szeliski, “Computer vision: algorithms and applications,” *Choice Rev. Online*, vol. 48,

no. 09, pp. 48-5140-48–5140, 2011.

[119] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep Face Recognition,” no. Section 3, pp. 41.1-41.12, 2015.

[120] H. W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, “Deep learning for emotion recognition on small datasets using transfer learning,” *ICMI 2015 - Proc. 2015 ACM Int. Conf. Multimodal Interact.*, pp. 443–449, 2015.

[121] A. M. Bukar and H. Ugail, “Convnet features for age estimation,” *Proc. Int. Conf. Comput. Graph. Vis. Comput. Vis. Image Process. 2017 Big Data Anal. Data Min. Comput. Intell. 2017 - Part Multi Conf. Comput. Sci. Info*, no. 2010, pp. 94–102, 2017.

[122] I. J. Goodfellow *et al.*, “Challenges in representation learning: A report on three machine learning contests,” *Neural Networks*, vol. 64, pp. 59–63, 2015.

[123] L. M. T. Baltruˇsaitis, A. Zadeh, Y. Chong Lim, “OpenFace 2 . 0 : Facial Behavior Analysis Toolkit,” *13th IEEE Interna- tional Conf. Autom. Face Gesture Recognit.*, pp. 59–66, 2018.

[124] V. Sangeetha and K. J. R. Prasad, “Deep Residual Learning for Image Recognition,” *Indian J. Chem. - Sect. B Org. Med. Chem.*, vol. 45, no. 8, pp. 1951–1954, 2006.

[125] T. F. Yap, J. Epps, E. Ambikairajah, and E. H. C. Choi, “Voice source under cognitive load: Effects and classification,” *Speech Commun.*, vol. 72, pp. 74–95, 2015.

[126] M. El Ayadi, M. S. Kamel, and F. Karray, “Survey on speech emotion recognition: Features, classification schemes, and databases,” *Pattern Recognit.*, vol. 44, no. 3, pp. 572–587, 2011.

[127] J. Weiner, C. Herff, and T. Schultz, “Speech-based detection of Alzheimer’s disease in conversational German,” *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*, vol. 08-12-Sept, pp. 1938–1942, 2016.

[128] R. Kliper, S. Portuguese, and D. Weinshall, “Prosodic analysis of speech and the underlying mental state,” *Commun. Comput. Inf. Sci.*, vol. 604, pp. 52–62, 2016.

[129] B. S. Helfer, T. F. Quatieri, J. R. Williamson, D. D. Mehta, R. Horwitz, and B. Yu, “Classification of depression state based on articulatory precision,” *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*, pp. 2172–2176, 2013.

[130] A. P. Association, *Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition (DSM-5)*, 5th ed. Arlington, VA: American Psychiatric Publishing, 2013.

[131] Z. N. Karam *et al.*, “Ecologically valid long-term mood monitoring of individuals with bipolar disorder using speech,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 4858–4862, 2014.

[132] N. Cummins, S. Scherer, J. Krajewski, S. Schnieder, J. Epps, and T. F. Quatieri, “A review of depression and suicide risk assessment using speech analysis,” *Speech Commun.*, vol. 71,

no. April, pp. 10–49, 2015.

[133] L. Muda, M. Begam, and I. Elamvazuthi, “Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques,” vol. 2, no. 3, pp. 138–143, 2010.

[134] F. Eyben *et al.*, “The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing,” *IEEE Trans. Affect. Comput.*, vol. 7, no. 2, pp. 190–202, 2016.

[135] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, “Real-life voice activity detection with LSTM Recurrent Neural Networks and an application to Hollywood movies,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 1, no. 3, pp. 483–487, 2013.

[136] A. R. Sonnenschein, S. G. Hofmann, T. Ziegelmayer, and W. Lutz, “Linguistic analysis of patients with mood and anxiety disorders during cognitive behavioral therapy,” *Cogn. Behav. Ther.*, vol. 47, no. 4, pp. 315–327, 2018.

[137] K. Kyle and S. A. Crossley, “Automatically Assessing Lexical Sophistication: Indices, Tools, Findings, and Application,” *TESOL Q.*, vol. 49, no. 4, pp. 757–786, 2015.

[138] R. A. Calvo, D. N. Milne, M. S. Hussain, and H. Christensen, “Natural language processing in mental health applications using non-clinical texts,” *Nat. Lang. Eng.*, vol. 23, no. 5, pp. 649–685, 2017.

[139] K. Collier, B. Bickel, C. P. van Schaik, M. B. Manser, and S. W. Townsend, “Language evolution: Syntax before phonology?,” *Proc. R. Soc. B Biol. Sci.*, vol. 281, no. 1788, 2014.

[140] S. A. Golder and M. W. Macy, “Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures,” *Science (80-.).*, vol. 333, no. 6051, pp. 1878–1881, 2011.

[141] S. A. Crossley, L. K. Allen, K. Kyle, and D. S. McNamara, “Analyzing Discourse Processing Using a Simple Natural Language Processing Tool,” *Discourse Processes*, vol. 51, no. 5–6. Taylor & Francis, pp. 511–534, 2014.

[142] S. A. Crossley, K. Kyle, and D. S. McNamara, “Sentiment Analysis and Social Cognition Engine (SEANCE): An automatic tool for sentiment, social cognition, and social-order analysis,” *Behav. Res. Methods*, vol. 49, no. 3, pp. 803–821, 2017.

[143] N. Ono, “Affective Norms for English Words (ANEW): Instruction Manual and Affective Ratings,” *Japanese J. Med. Electron. Biol. Eng.*, vol. 22, pp. 14–15, 1984.

[144] S. M. Mohammad and P. D. Turney, “Crowdsourcing a word-emotion association lexicon,” *Comput. Intell.*, vol. 29, no. 3, pp. 436–465, 2013.

[145] E. Cambria, R. Speer, C. Havasi, and A. Hussain, “SenticNet: A publicly available semantic resource for opinion mining,” *AAAI Fall Symp. - Tech. Rep.*, vol. FS-10-02, pp. 14–18,

2010.

[146] H. D. L. and J. Z. Namenwirth, "The Lasswell value dictionary," *New Haven*, 1969.

[147] T. Dang *et al.*, "Investigating word affect features and fusion of probabilistic predictions incorporating uncertainty in AVEC 2017," *AVEC 2017 - Proc. 7th Annu. Work. Audio/Visual Emot. Challenge, co-located with MM 2017*, pp. 27–35, 2017.

[148] L. Zhang, J. Driscol, X. Chen, and R. H. Ghomi, "Evaluating acoustic and linguistic features of detecting depression sub-challenge dataset," *AVEC 2019 - Proc. 9th Int. Audio/Visual Emot. Chall. Work. co-located with MM 2019*, no. 1, pp. 47–53, 2019.

[149] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *1st Int. Conf. Learn. Represent. ICLR 2013 - Work. Track Proc.*, pp. 1–12, 2013.

[150] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations ofwords and phrases and their compositionality," *Adv. Neural Inf. Process. Syst.*, pp. 1–9, 2013.

[151] C. Besana, M. Memoli, P. M. Salvioni, R. A. Finazzi, F. Inversi, and C. Rugarli, "Linguistic Regularities in Continuous Space Word Representations," *Subst. Use Misuse*, vol. 26, no. 5, pp. 505–513, 1991.

[152] L. Ma and Y. Zhang, "Using Word2Vec to process big text data," *Proc. - 2015 IEEE Int. Conf. Big Data, IEEE Big Data 2015*, no. October 2015, pp. 2895–2897, 2015.

[153] Tin Kam Ho, "Random Decision Forests," *Proc. 3rd Int. Conf. Doc. Anal. Recognit.*, pp. 278–282, 1995.

[154] N. Abramson, D. Braverman, and G. Sebestyen, "Pattern recognition and machine learning," *IEEE Trans. Inf. Theory*, vol. 9, no. 4, pp. 257–261, 1963.

[155] N. H. Farhat, "Support-Vector Networks CORINNA," *IEEE Expert. Syst. their Appl.*, vol. 7, no. 5, pp. 63–72, 1992.

[156] G. Bin Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," *IEEE Int. Conf. Neural Networks - Conf. Proc.*, vol. 2, pp. 985–990, 2004.

[157] A. Aksoy, Y. E. Ertürk, S. Erdoğan, E. Eyduran, and M. M. Tariq, "Long Short-Term Memory Sepp," *Pak. J. Zool.*, vol. 50, no. 6, pp. 2199–2207, 1997.

[158] Thomas G. Dietterich, "Ensemble Learning," *Handb. brain theory neural networks*, vol. 2, pp. 110–125, 2002.

[159] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Inf. Fusion*, vol. 37, pp. 132–156, 2017.

[160] R. Richman and M. V. Wüthrich, “Bagging predictors,” *Risks*, vol. 8, no. 3, pp. 1–26, 2020.

[161] Y. Freund and R. E. Schapire, “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting,” *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[162] D. H. Wolpert, “Stacked Generalization,” *Elsevier*, vol. 87545, no. 505, pp. 1–57, 1992.

[163] S. Picco *et al.*, “Data Mining and Machine Learning for Software Engineering,” *Intech*, no. tourism, p. 13, 2016.

[164] Y. Koren, “The BellKor Solution to the Netflix Grand Prize,” no. August, pp. 1–10, 2009.

[165] M. Piotte and M. Chabbert, “The Pragmatic Theory solution to the Netflix Grand Prize,” no. August, pp. 1–92, 2009.

[166] S. Džeroski and B. Ženko, “Is combining classifiers with stacking better than selecting the best one?,” *Mach. Learn.*, vol. 54, no. 3, pp. 255–273, 2004.

[167] Y. Attabi and P. Dumouchel, “Anchor models for emotion recognition from speech,” *IEEE Trans. Affect. Comput.*, vol. 4, no. 3, pp. 280–290, 2013.

[168] S. Saxena and J. Verbeek, “Convolutional neural fabrics,” *Adv. Neural Inf. Process. Syst.*, pp. 4060–4068, 2016.

[169] J. Snoek, O. Rippel, and R. P. Adams, “Scalable Bayesian Optimization Using Deep Neural Networks arXiv:1502.05700v2,” 2012.

[170] B. Zoph and Q. V. Le, “Neural architecture search with reinforcement learning,” *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, pp. 1–16, 2017.

[171] R. J. Willia, “Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning,” *Mach. Learn.*, vol. 8, no. 3, pp. 229–256, 1992.

[172] C. Olah, “Understanding LSTM Networks,” 2015. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[173] Z. Zhang, F. Ringeval, B. Dong, E. Coutinho, E. Marchi, and B. Schuller, “Enhanced semi-supervised learning for multimodal emotion recognition,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 2016-May, pp. 5185–5189, 2016.

[174] A. H. Yazdavar *et al.*, “Semi-Supervised approach to monitoring clinical depressive symptoms in social media,” *Proc. 2017 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Mining, ASONAM 2017*, pp. 1191–1198, 2017.

[175] I. Cohen, N. Sebe, F. G. Cozman, T. S. Huang, H. P. Labs, and P. Alto, “Semi-Supervised Learning for Facial Expression Recognition Categories and Subject Descriptors,” *Search*.

[176] A. Rasmus, H. Valpola, M. Honkala, M. Berglund, and T. Raiko, “Semi-supervised learning

with Ladder networks," *Adv. Neural Inf. Process. Syst.*, vol. 2015-Janua, pp. 3546–3554, 2015.

[177] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol, "Stacked denoising autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.

[178] M. Pezeshki, L. Fan, P. Brakel, A. Courville, and Y. Bengio, "Deconstructing the ladder network architecture," *33rd Int. Conf. Mach. Learn. ICML 2016*, vol. 5, pp. 3527–3539, 2016.

[179] S. Joseph, "Australian Literary Journalism and 'Missing Voices': How Helen Garner finally resolves this recurring ethical tension," *Journal. Pract.*, vol. 10, no. 6, pp. 730–743, 2016.

[180] R. A. Fisher, "XV.—The Correlation between Relatives on the Supposition of Mendelian Inheritance," *Trans. R. Soc. Edinburgh*, vol. 52, no. 2, pp. 399–433, 1919.

[181] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[182] H. Robbins and S. Monro, "A Stochastic Approximation Method," vol. 22, no. 3, pp. 400–407, 2014.

[183] P. Liashchynskyi and P. Liashchynskyi, "Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS," no. 2017, pp. 1–11, 2019.

[184] L. I.-K. Lin, "A Concordance Correlation Coefficient to Evaluate Reproducibility," *Biometrics*, vol. 45, no. 1, p. 255, 1989.