National Research University Higher School of Economics

Faculty of Computer Science

Bachelor's Programme in Data Science and Business Analytics

BACHELOR'S THESIS

Research Project

Predicting the severity of depression with audio files and artificial intelligence

Prepared by the student of Group 192, Year 4,

Gursoy Alexandr-Efe

Supervisor:

Laboratory Head Vision Modelling Laboratory, Soroosh Shalileh

Moscow

2023

Table of Contents

Abstract

Depression is a prevalent mental disorder that affects millions of people worldwide. It is often a big challenge to correctly diagnose depression and may pose certain challenges. To address these challenges people are trying to find reliable ways to diagnose depression as soon as possible. In this paper we are going to investigate the possibility of predicting the depression severity using Machine Learning. We inspect a methodology of predicting the depression severity using the audio files of the participant. Our methodology involves investigating a way to predict the severity of depression using raw audio time series using time series forecasting models for forecasts and Dynamic Time Warping for classification. This paper compares the performance of three models on this task: DeepAR, N-Beats and Temporal Fusion Transformers. The best performing model among these three gave a weighted F1-score equal to 58.4%.

1. Introduction

Depression is a common mental disorder **[1]**, the three main symptoms of which are persistent low mood, loss of interest and lack of energy **[2]**. It can lead to significant impairments in daily functioning and can even lead to suicide in severe cases**[1]**. Accurate measurement of depression severity is crucial for effective diagnosis and treatment. However, diagnosing and treating depression can be challenging due to the varying nature of symptoms. The reliability of these measures can be limited by factors such as social desirability bias[1] **[3]**. As a result, there is a growing interest in developing objective and reliable methods for measuring depression severity.

Deep learning models have shown promise in various applications, including time series forecasting, speech recognition, and natural language processing. A lot of researches have been conducted in order to predict the depression severity of the person and the number of attempts keeps increasing. Studies considered using video, text and audio information for this task. One promising approach considered in those researches is to use physiological data, such as audio recordings, to measure depression severity. Audio data has the potential to provide more objective measures of depression severity, as it captures changes in vocal tone, pitch, and speech patterns, which have been shown to be associated with depression **[27]**.

In this research, we want to develop a novel classification method of depression severity prediction. Specifically, we want to determine the most effective time series forecasting model for audio data in our domain by comparing the performance of DeepAR **[4]**, Temporal Fusion Transformers (TFT) **[5]**, and NBeats **[6]** and their applicability to this classification task. We will then develop a

[1] a tendency to overreport more favorable characteristics while underreporting socially unfavorable attitudes and actions

classification method by using Dynamic Time Warping to compare the forecasted time series with the training set and classify the depression severity of the person. Finally, we aim to contribute to the field of depression severity classification by providing a novel approach to measure depression severity using audio data.

This paper is structured as follows. We begin by discussing related work in Section 2. Section 3 discusses the approach we are using to detect depression severity of participants in detail, provides a comprehensive explanation of each of the steps partaken, and gives details on how training and fine-tuning was conducted. Section 4 presents the results of the study and their interpretation, it includes analysis and interpretation of the data collected through the methods outlined in Section 3, and we draw conclusions in Section 5.

2. Relevant work

Depression identification:

The AVEC 2016 workshop and challenge**[7]** focused on predicting depression using audio data. For the task of depression prediction using audio recordings, authors used prosodic features, and spectral features. Authors use Support Vector Machines (SVM) and Random Forests (RF) to predict the depression of a participant. An F1-score of 0.41 for SVM and Mean Absolute Error (MAE) of 5.72 for RF was achieved on DAIC-WOZ dataset**[8]**. Another work**[9]** using the same dataset has extracted spectral, cepstral, prosodic and voice quality features and applied a Deep Convolutional Neural Networks for classification. The text data, extracted from the audio, was also utilized in the research. Authors of the research obtained MAE of 4.359. In "Detecting Depression with Audio/Text Sequence Modeling of Interviews,"**[11]** the authors propose a novel approach to detect depression by modeling the sequence of audio and text data from interviews. The researchers use a hybrid CNN-RNN model to process the audio data and reach 67% F1-score. Salekin et al.**[10]** proposes a new feature modeling technique NN2Vec, which maps raw audio signals to fixed-length embeddings, which are then used to represent audio segments. They use BLSTM-MIL classifier achieving 85.44% F1-score and accuracy equal to 96.7% in detecting speakers' depression symptoms on DAIC-WOZ dataset. Another work **[14]** approaches the prediction of depression using both audio and textual data by combining a 1d-CNN and Bidirectional LSTM. The audio features are extracted using Mel Spectrograms and are passed to 1d-CNN, and the text features are extracted via pretrained ELMo model. The authors combine those two features and feed them into two fully connected networks, reaching 85% F1-score on the DAIC-WOZ dataset.

Approaches to audio classification:

Several studies **[23][24]** on detecting emotions and mental disorders from speech have recognized that temporal properties in a speech may provide important information about mental disorders and used sequential classifiers such as Long Short-Term Memory (LSTM)**.**

Tzirakis et al.**[12]** proposes a deep learning-based approach for emotion recognition from speech signals. The proposed model is a combination of a convolutional neural network (CNN) and a LSTM network. The CNN is used to extract relevant features from the input speech signal, while the LSTM network is used to capture the temporal dynamics of the speech signal. A novel approach to audio classification was developed by **[15],** where a 1 dimensional CNN is built and raw audio time series are fed into it. This approach proved to be efficient at environmental sound classification, showing similar performance with DCNN which is fed log-scaled mel-spectrograms together with different kinds of augmented data **[16]**. Audio classification using Convolutional Recurrent Neural Networks is discussed in **[17].** The underlying idea of this architecture is that temporal patterns are better aggregated with RNN then CNN and they use CNN for local feature extraction. The proposed architecture of CRNN outperforms SOTA model **[18]** in the music tagging

Time Series Forecasting:

Time series forecasting is a powerful tool that allows the analysis and prediction of temporal data. In recent years, several deep learning methods have been developed for time series forecasting. DeepAR**[4]** is a state-of-the-art probabilistic forecasting method that is based on recurrent neural networks (RNNs) and autoregressive models. DeepAR is particularly effective when dealing with long-term dependencies and seasonality in time series data. Temporal Fusion

Transformers**[5]** is another advanced deep learning method for time series forecasting that combines the strengths of RNNs and transformers. TFT can model both short-term and long-term dependencies in time series data and has achieved state-of-the-art performance on several benchmark datasets. N-Beats**[6]** is a novel deep learning architecture that is designed specifically for time series forecasting. N-Beats uses a fully convolutional network (FCN) and a self-attention mechanism **[25]** to model the temporal patterns in time series data.

3. Methodology

3.1 Data Description and Preprocessing

The data was collected by HSE University, Center for Language and Brain. Participants had to present answers across three tasks. Each task contained three variants of stimuli to help them answer. Below are the types of tasks presented to participants and stimuli provided for answering them:

- Narratives. For stimuli one of the three comics was used: “Superman”, “Discovery of the World”, “Wonderful Day”.
- Stories. As a stimuli, one of three questions about notable occasions in the participant’s life was used: “please tell me about the best or the most memorable gift you have received”; “please tell me about the best or the most memorable trip you have gone on”; “please tell me about the best or the most memorable party you have had”.
- Instructions. For this, IKEA’s self-assembly furniture manuals for a chair, a table, or a bench were used.

Overall, there are 276 participants, 138 participants in the psychiatric clinic and 140 participants in the control group. Participants in the psychiatric clinic were assessed with clinical scales by a psychiatrist. The control group completed online questionnaires to assess their psychiatric symptoms. Each participant has 3 recorded audios. In this dataset, absence of depression is labeled by “0”, light depression is “1”, moderate depression is denoted by “2”, and severe depression is “3”

The audio recordings were sampled at **8 kHz**, and audio was trimmed at 25 dB **[22]** to reduce silent parts in the beginning and the end.

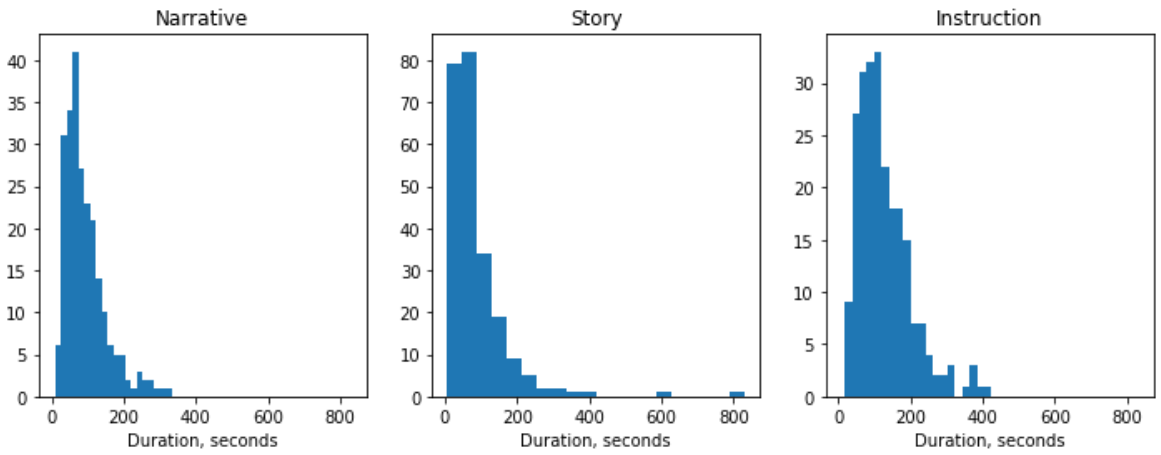

Figure 1. Distribution of audio recording lengths by type of task.

Table 1. Minimum, maximum durations and 25th, 50th, 75th percentiles of durations in seconds over all three types of tasks.

	min	25%	50%	75%	max
duration	6	52.75	83	131	832

Each audio recording was cut into multiple 5 second segments. Figure 1 and Table 1 show the distribution of audio recordings lengths. Due to some recordings being too long a maximal length of recording was set to 500 seconds. That is, a single audio can be split into a maximum of 100 five-second parts. This was done to ensure that all audio recordings in the dataset have the same length and can be used to train and evaluate the model consistently.

The dataset is split into 80% training and 20% test. For each patient there are three audio recordings for each type of stimuli: narrative, story and instruction. Age, sex and education are omitted.

3.2 Overview of classification approach

Our approach for classifying the depression severity of people involves two main steps: forecasting time series extracted from the persons audio files across all three tasks and classifying the severity of depression of a person using forecasted time series.

The first step is to build a model that would forecast the time series extracted from the audio files. The preprocessed audio data would then be used to forecast time series using our models. Time series forecasting models can identify patterns and trends in the audio data that may not be immediately apparent from raw time series data. Overall, by forecasting time series extracted from audio, the classification of the severity of depression may be enhanced and allow for more accurate classification because the models will aim for the identification of patterns and trends in the data.

On the second step, we perform a classification to identify the severity of depression of the person. This step involves using Dynamic Time Warping **[19]** to calculate the distance between the predicted and actual time series data from the training set.

3.3 Models

In this section we explore the three deep learning models for time series audio forecasting, which we are using in our methodology: DeepAR**[4]**, N-Beats**[6]**, and Temporal Fusion Transformers**[5]**. These models were selected for their effectiveness in time series forecasting tasks and were trained on the audio recordings of depressed and non-depressed individuals. To ensure consistency and comparability across the models, we have set common hyperparameters to be the same for our not fine-tuned models, which we considered as baselines, which are presented in Table 2.

Table 2. Shared hyperparameters set for each of the not fine-tuned models.

learning rate	1e-3
optimizer	AdamW
dropout	0.1
prediction length	200
encoder length	200

3.3.1 DeepAR

The DeepAR **[4]** architecture is based on an autoregressive approach to time series forecasting, where the model learns to predict future values of a time series based on past observations (figure 2). Autoregressive models are designed to capture the dependencies between the past and future values in time series, and are commonly implemented using a recurrent neural network architecture such as the Long Short-Term Memory network **[20]** or the Gated Recurrent Unit network **[21]**. The DeepAR model's main goal is to model the conditional distribution of the future of time series given its past.

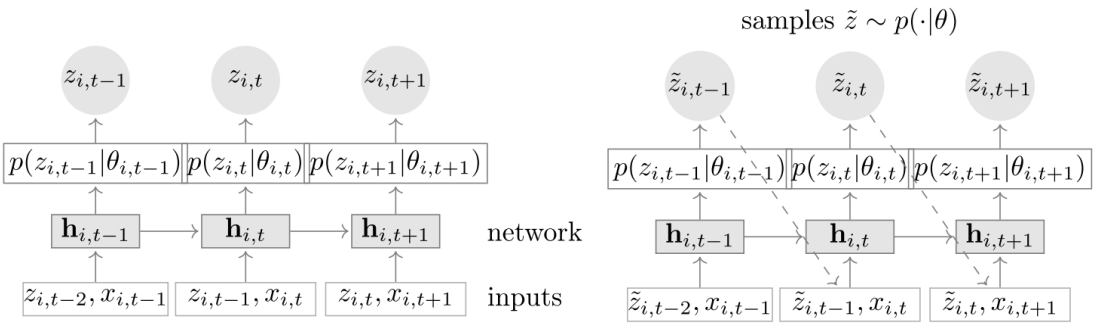

Figure 2. Summary of the DeepAR model. **[4]**

To train the DeepAR model, we provide it with a set of historical observations and predict a certain number of future time steps. The model then outputs a distribution of possible values for each of these time steps, which represents the uncertainty in the prediction.

Base DeepAR was trained using the common hyperparameters stated at the beginning of the section together with the following model-specific parameters (Table 3).

Table 3. Base DeepAR hyperparameters

rnn layers	2
hidden size	10

3.3.2 N-Beats

The N-BEATS **[6]** architecture is a deep learning model designed for time series forecasting that uses a stack of fully connected neural networks (blocks) to predict future values of a time series. A fully connected neural network consists of a series of fully connected layers, that is in turn is a collection of neurons, each of

which is connected to every other neuron in the other layer. A simple version of a fully connected neural network can be seen in **figure 3.**

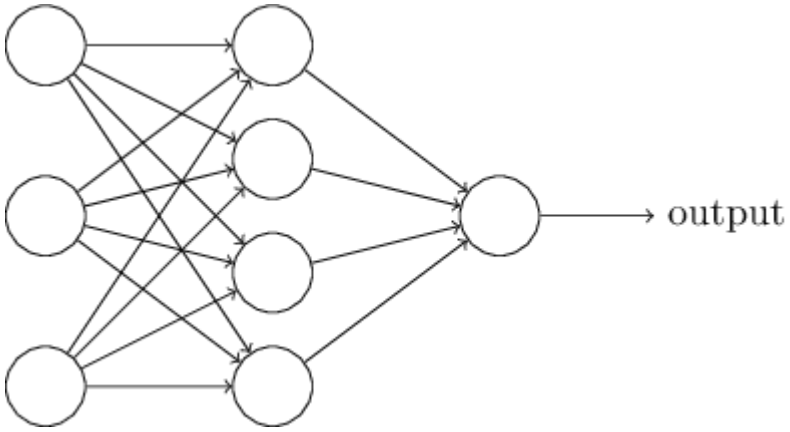

Figure 3. A simple fully connected neural network. Each circle represents a neuron and the arrows, or the edges, represent weights and biases between layers.

Figure 4 describes the architecture of N-Beats, each block is composed of a series of fully connected layers and each stack is composed of a series of blocks. Intuitively, each block in the model outputs a partial forecast of the time series **y** that it can approximate well, and passes the residual of **x** to the next block, which represents a signal, or the input time series given for context, removing its portion. The final forecast is then the sum of all partial forecasts of stacks, present in the model.

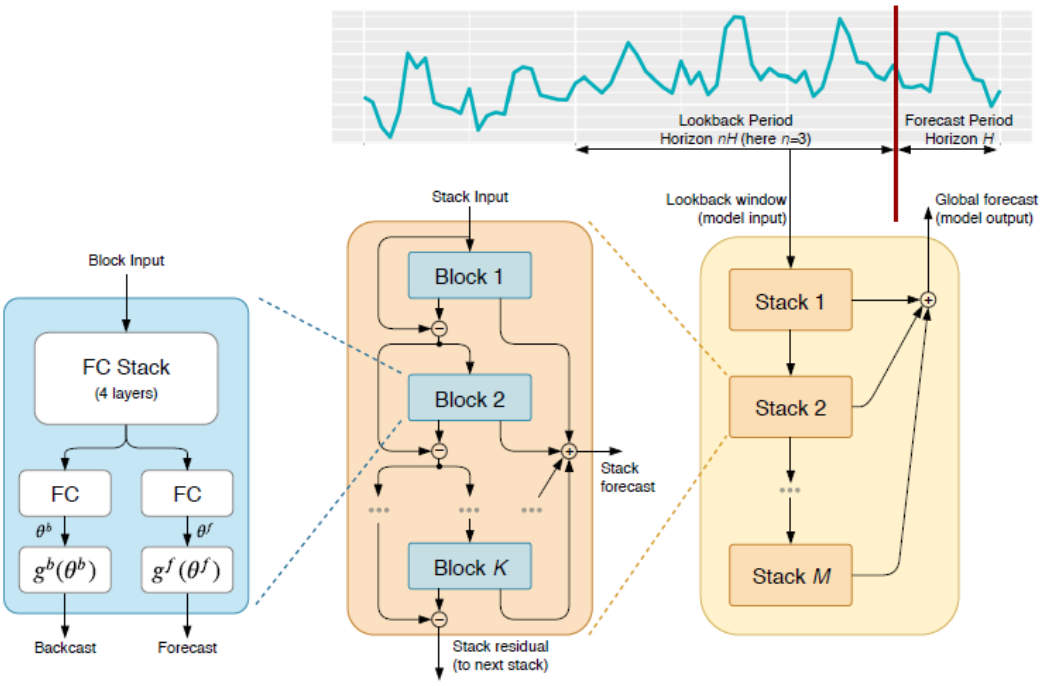

Figure 4. Architecture of N-Beats model. **[6]**. θ is the output from fully connected layers. g(θ) is a function that is defined by the configuration of the model. If model architecture is defined to be "generic", g(θ) becomes a linear function with learnable parameters, if the architecture is chosen to be "interpretable", g(θ) decomposes the output into trend and seasonality **[26]**. Output with superscript **b** is the backcast output passed to next block and superscript **f** is the forecast, which is aggregated between blocks and passed to the output of the model.

Base N-Beats was trained using the common hyperparameters stated at the beginning of the **Models** section together with the model-specific parameters presented in Table 4.

Table 4. Base N-Beats hyperparameters.

num blocks	1
num block layers	4
width	512
backcast loss ratio	0.5

Here is the brief explanation of each of the hyperparameters:

- num blocks controls the number of fully connected neural network blocks in the N-BEATS model. Increasing the number of blocks can allow the model to capture more complex patterns in the data.
- num block layers control the number of fully connected layers in each block.
- width controls the number of hidden units in each fully connected layer of the model.
- backcast loss ratio controls the relative weighting of the backcast and forecast losses in the loss function used to train the model and can take values in range 0.0-1.0

3.3.3 Temporal Fusion Transformers

TFT's **[5]** architecture consists of gating mechanisms, variable selection networks, which select relevant input variables at each time step, encoders to account for any status features fed into the network, temporal processing for learning time-varying inputs, both short and long-term, and prediction intervals giving quantile forecasts.

Base TFT was trained using the common hyperparameters stated at the beginning of the **Models** section together with the model-specific parameters shown in Table 5.

Table 5. Base TFT hyperparameters

hidden size	16
LSTM layers	2
Attention head size	4

Here is the brief explanation of each of the hyperparameters:

- Hidden size determines the number of neurons in each hidden layer of the model.
- LSTM layers determine the number of Long Short-term Memory layers in the model.
- Attention head size is the number of attention heads used in each multi-head attention layer in the model.

3.4 Classification

Dynamic Time Warping (DTW) is a distance-based method for measuring the similarity between two time series with different lengths or sampling rate. The basic idea behind DTW is to find the optimal alignment between two time series by warping one of the time series along the time axis. This allows for matching of similar patterns in the time series, even if they occur at different points in time. DTW works by finding the path of least resistance through a cost matrix that represents the pairwise distances between the elements of the two time series.

Formally, the optimization problem is the following:

$$DTW_q(x, x') = \min_{\pi \in \mathcal{A}(x,x')} \left(\sum_{(i,j)\in\pi} d(x_i, x'_j)^q \right)^{\frac{1}{q}} \quad \textbf{(1)}$$

Here, an alignment path π of length *K* is a sequence of *K* index pairs $((i_0, j_0), ..., (i_{K-1}, j_{K-1}))$ and $\mathcal{A}(x, x')$ is the set of all admissible paths. In simple words, an alignment path is a set of pairs of indices that tells which elements of two time series to compare. The two series may have different numbers of time steps and DTW uses an alignment path to match up the time steps of the two series in a way that allows for meaningful comparison. The alignment path defines a mapping between the time steps of the two series that maximizes their similarity. The set of all admissible paths is a collection of all possible mappings between two time series, which follow two rules:

- The sequence is monotonically increasing in both and and all time series indexes should appear at least once.
- Beginning of time series are matched together.

In this paper, we use DTW to compare the predicted time series with time series from the training set, and classify the test observation based on the most similar observation from the training set. To do so, we compute the euclidean euclidean distance measure **(2)** between the predicted time series and all time series from the training set. Figure 5 provides an example of matching the two time series using DTW. The pair that yields the minimum distance measure is chosen and the test observation is assigned the class from the most similar training observation.

$$d(p, q) = \sqrt{(p - q)^2} \quad \textbf{(2)}$$

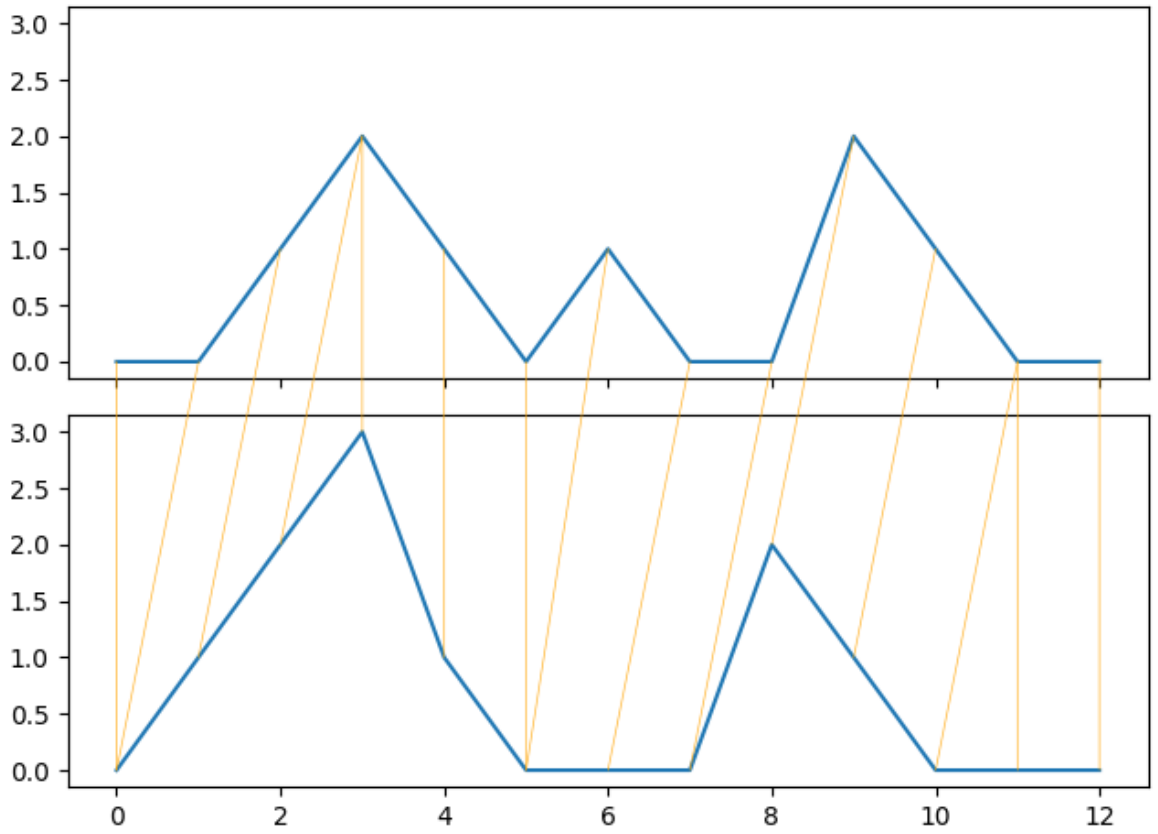

Figure 5. Sample picture of DTW usage. Blue lines represent two time series and yellow lines show how the DTW algorithm matched the points of the time series.

After assigning each of the predicted time series a class, we merge previously cut segments into one. If the initial audio was cut into more than 2 parts, we find the segment with the shortest distance to the train data and assign the same class as predicted by this observation. Test observation is assigned the class of the audio part that has minimum distance.

Table 6. Sample predictions.

id	pred_severity	min_dist	actual_severity
0_PN-204	1	0.0439	0
1_PN-204	0	0.0072	0
0_PD-070	0	0.0071	0

Table 6 shows sample predictions. Column id of an audiofile is starting with a number, which represents the n-th segment of an audiofile of the person, PN are the participants in the psychiatric clinic and PD is the control group. Recording of PN-204, for instance, was cut into two parts marked, so the id's begin with 0 and 1 accordingly. The first cut of the participants recording was predicted to have a light depression

3.5 Fine-Tuning

In this section, we describe the fine-tuning process for the three models used in our study. To fine-tune the hyperparameters of the models under consideration we are using the Grid Search algorithm. Due to computational intensity, different sets of hyperparameters (i.e. number of blocks and number of layers in one block) were combined to decrease the number of trained models during the Grid Search.

3.5.1 DeepAR fine-tuning

For DeepAR **[4]**, we trained the model using all three types of audio recordings as targets. This approach enables the model to learn the temporal

dependencies and patterns across the different stimuli types. Table 7 shows the resulting hyperparameters set for the model.

Table 7. DeepAR fine-tuning results. Hyperparameters: p - prediction length, c - context length, h - hidden size, RNN_n - number of RNN layers.

	hyperparameters			
	p	c	h	RNN_n
DeepAR	200	200	32	4

3.5.2 N-Beats fine-tuning

For N-Beats **[6]**, three separate models were trained, each having set a target to a specific type of stimuli (narrative, story, and instruction). Table 8 shows the final hyperparameter sets for each of the N-Beats models after applying Grid Search.

Table 8. N-Beats fine-tuning results. Hyperparameters: p - prediction length, c - context length, B - number of blocks, N_B - number of FC layers in each block, W - width, L_b - backcast loss ratio

	stimuli	hyperparameters					
		p	c	B	N_B	W	L_b
N-Beats	narrative	400	800	2	8	512	0.2
	story	400	800	2	4	1024	0.2
	instruction	400	800	4	4	1024	0.2

3.5.3 TFT fine-tuning

Three TFT **[5]** models were trained in the similar manner to N-Beats training.

Table 9. TFT fine-tuning results. Hyperparameters: p - prediction length, c - context length, h - hidden size, N_A - attention head size, $LSTM_N$ - number of LSTM layers.

	stimuli	parameters				
		p	c	h	N_A	$LSTM_N$
TFT	narrative	200	200	128	4	2
	story	200	200	128	4	2
	instruction	200	200	32	4	2

4. Experimental results and analysis

4.1 Evaluation Metrics

In multiclass classification, some of the metrics are calculated separately for each of the classes and in our case there are four classes, which are then aggregated into weighted metric accounting for class imbalance.

The results of classification are measured using the following performance metrics for classification: F1-score, precision, recall and accuracy.

Before proceeding to describe each of the metrics, we define the weights of each class in (3), which are used for computing weighted F1-score, precision and recall.

$$w_i = \frac{\text{No. of samples in class i}}{\text{Total number of samples}} \tag{3}$$

4.1.1 Precision

Formula (4) is precision calculated on i-th class.

$$Precision_i = \frac{TP(class = i)}{TP(class = i) + FP(class = i)} \tag{4}$$

Formula (5) is the weighted precision, which we are using for evaluation in our research.

$$Precision_{weighted} = \Sigma_{\text{i=1 to n}} w_i * Precision_i \tag{5}$$

4.1.2 Recall

Formula (6) is recall calculated on i-th class.

$$Recall_i = \frac{TP(class = i)}{TP(class = i) + FN(class = i)} \tag{6}$$

Formula (7) is the weighted recall, which we are using for evaluation in our research.

$$Recall_{weighted} = \Sigma_{\text{i=1 to n}} w_i * Recall_i \tag{7}$$

4.1.3 F1-score

Formula (8) is recall calculated on i-th class.

$$\text{F1-score}_i = \frac{2 * Precision_i * Recall_i}{Precision_i + Recall_i} \tag{8}$$

Formula (9) is the weighted recall, which we are using for evaluation in our research.

$$\text{F1-score}_{weighted} = \Sigma_{\text{i=1 to n}} w_i * \text{F1-score}_i \tag{9}$$

TP (true positives) is the number of observations that are predicted the same as actual class TP is calculated for. FP (false positives) is the number of observations that are classified as different classes but are actually the class FP is calculated for. FN (false negatives) is the number of observations that are predicted as different classes, rather than the one FN is calculated for.

4.2 Forecasts of audio time series

The **figures 6, 7, 8** present two audio time series that were predicted by three models: N-Beats, DeepAR, TFT.

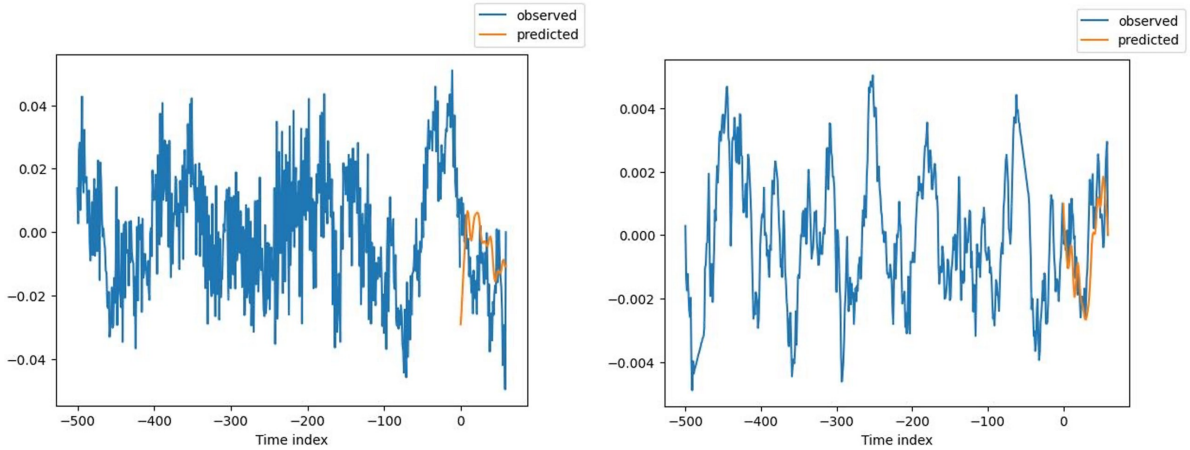

Figure 6 (a, b). N-Beats predictions

N-Beats (**Figure 6**) shows the most promising results, most closely predicting the audio out of three models and approximately capturing the overall future trends in the time series.

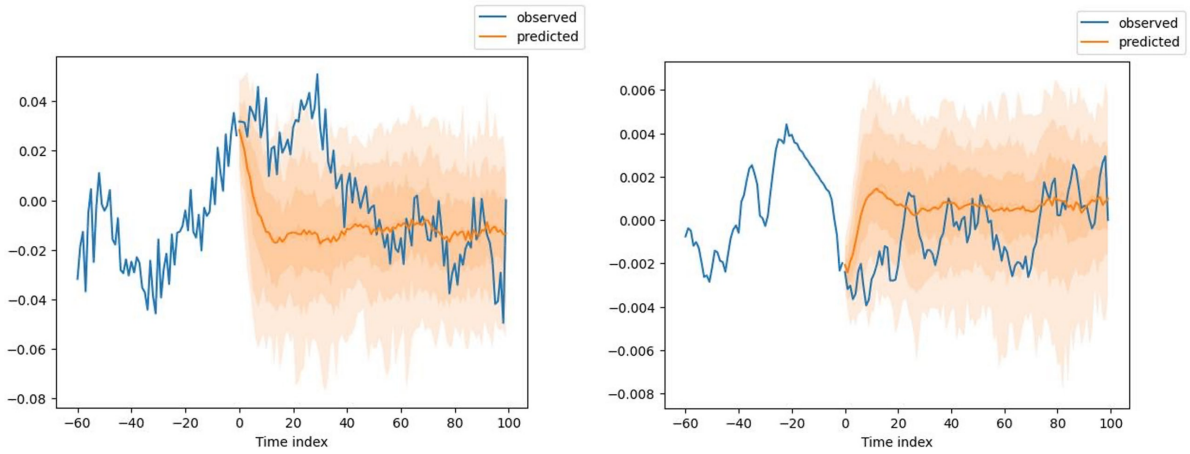

Figure 7 (a, b). DeepAR predictions

DeepAR on **Figure 7** has shown poor performance, missing the overall dynamics by a lot. This may be caused due to lack of training and simplicity of the model. This is further aimed to be resolved in the fine-tuning stage.

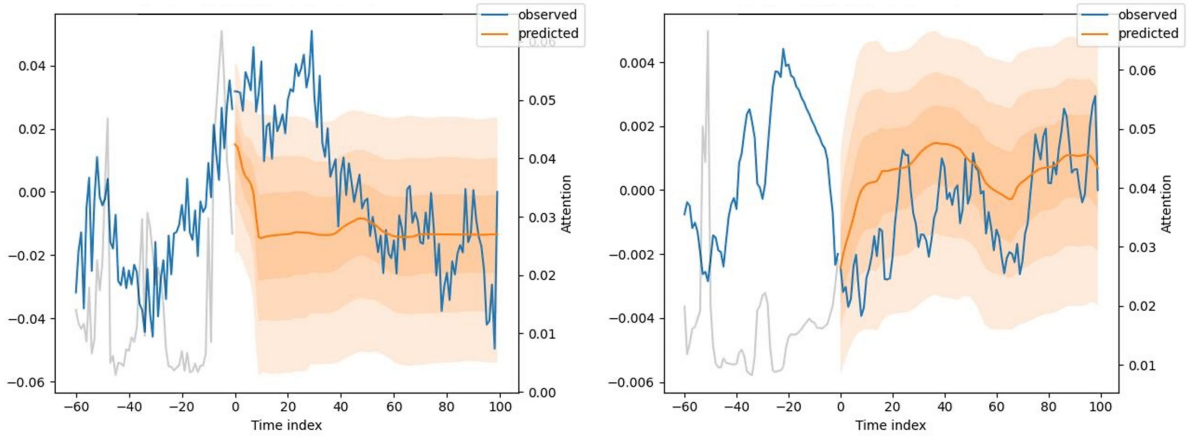

Figure 8 (a, b). TFT predictions

Temporal Fusion Transformers are presented on **Figure 8**. They have a result in between, sometimes capturing the trends and correctly predicting future trends in time series. The issue is most likely to persist the same as with DeepAR, an increased complexity of the model will be required.

4.3 Results for depression severity prediction

The results are computed using a stratified 10-fold cross-validation. The data is splitted into 10 random folds, each consisting of train split and test split being 80% and 20% of data respectively. This method helps to ensure a robust evaluation of the models' generalization capabilities. The four metrics are used to evaluate model performance are weighted f1-score, weighted precision, weighted recall, and accuracy.

Table 10 presents the performance of three not fine-tuned models (DeepAR, N-Beats, and TFT) on a classification task. Based on the table, it appears that among three models N-Beats has the highest f1-score (0.444), precision (0.458), and recall (0.438). DeepAR has an f1-score of 0.35, precision of 0.35, and recall of 0.354. TFT appears to perform the worst, with an f1-score of 0.356, precision of 0.36, recall of 0.354.

Table 10. Not fine-tuned model results

	precision	recall	f1-score
DeepAR	0.3521	0.3541	0.3495
$NBeats_{narrative}$	**0.4578**	**0.4375**	**0.4436**
$TFT_{narrative}$	0.3604	0.3541	0.3568

Table 11 presents the evaluation results of the fine-tuned models on the prediction of depression levels across different stimuli types. With fine-tuning, the advantages of NBeats model have vanished. Across present stimuli types, the most informative seems to be narrative, as N-Beats and TFT are performing the best

among two other stimuli, when trained on it specifically. DeepAR has outperformed all other models in terms of identifying participants with depression.

Table 11. Model results for depression severity prediction: the average values and standard deviation of evaluation metrics over 10 different data splits.

Model	Metrics		
	$Precision_w$	$Recall_w$	$F1_w$
$NBeats_{narrative}$	0.576±0.022	0.575±0.024	0.568±0.020
$NBeats_{story}$	0.411±0.043	0.389±0.045	0.397±0.043
$NBeats_{instruction}$	0.470±0.032	0.439±0.025	0.441±0.024
$TFT_{narrative}$	**0.579±0.038**	0.591±0.031	0.576±0.024
TFT_{story}	0.447±0.028	0.485±0.024	0.458±0.027
$TFT_{instruction}$	0.501±0.052	0.497±0.033	0.485±0.036
DeepAR	0.565±0.043	**0.625±0.039**	**0.584±0.036**

In case of binary classification, TFT turns out to be the best performing model, reaching a weighted F1-score of 65% and accuracy of 66%, as is shown in the Table 12.

Table 12. Model results for binary classification of identifying depression: the average values and standard deviation of evaluation metrics over 10 different data splits.

Model	Metrics		
	$Precision_w$	$Recall_w$	$F1_w$
$NBeats_{narrative}$	0.618±0.040	0.620±0.040	0.614±0.042
$NBeats_{story}$	0.492±0.043	0.477±0.043	0.481±0.043
$NBeats_{instruction}$	0.541±0.045	0.512±0.037	0.516±0.037
$TFT_{narrative}$	**0.659±0.039**	**0.664±0.040**	**0.655±0.042**
TFT_{story}	0.531±0.044	0.554±0.034	0.536±0.042
$TFT_{instruction}$	0.575±0.046	0.558±0.045	0.562±0.044
DeepAR	0.648±0.052	0.656±0.046	0.638±0.048

5. Conclusion

In conclusion, this project presents an approach to predict the depression severity using raw time series extracted from the audio files and various time series forecasting techniques.

A processing pipeline for predicting depression severity of a person was developed. Three models have been implemented and fine-tuned over the course of this work, which were aimed at learning to forecast the audio time series with their consequent classification using Dynamic Time Warping algorithm. Among the models used, DeepAR performed best in predicting the severity of depression, but it should be taken into account that it was originally provided with more information due to its architecture. DeepAR has reached a weighted F1-score of 58.4%, being just slightly lower than the TFT trained on narrative type audio recordings. In terms of binary classification, TFT trained on narrative type audio recordings have shown the best results getting 65.5% weighted F1-score and 66.4% weighted recall. Also, models trained specifically on narrative type of tasks have shown to be the most efficient in terms of performance. However, these results cannot be considered good, because the data is imbalanced and presented methods just slightly outperform random prediction.

For further research, including demographic data may be considered. Also, we were limited in computational power and further increasing the complexity of the researched models in this paper can be explored. Depression by itself may have very varying symptoms from person to person **[13]**, so that in order to have more representativeness data augmentation may also be considered and more participants may be required.

GitHub repository

6. Bibliography

[1] Depression, W. H. O. (2017). Other common mental disorders: global health estimates. Geneva: World Health Organization, 24.

[2] Peveler, R., Carson, A., & Rodin, G. (2002). Depression in medical patients. Bmj, 325(7356), 149-152.

[3] Latkin, C. A., Edwards, C., Davey-Rothwell, M. A., & Tobin, K. E. (2017). The relationship between social desirability bias and self-reports of health, substance use, and social network factors among urban substance users in Baltimore, Maryland. Addictive behaviors, 73, 133-136.

[4] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting, 36(3), 1181-1191.

[5] Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748-1764.

[6] Oreshkin, B. N., Carpov, D., Chapados, N., & Bengio, Y. (2019). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. arXiv preprint arXiv:1905.10437.

[7] Valstar, M., Gratch, J., Schuller, B., Ringeval, F., Lalanne, D., Torres Torres, M., ... & Pantic, M. (2016, October). Avec 2016: Depression, mood, and emotion recognition workshop and challenge. In Proceedings of the 6th international workshop on audio/visual emotion challenge (pp. 3-10).

[8] Gratch, J., Artstein, R., Lucas, G., Stratou, G., Scherer, S., Nazarian, A., ... & Morency, L. P. (2014). The distress analysis interview corpus of human and computer interviews. University of Southern California Los Angeles.

[9] Yang, L., Sahli, H., Xia, X., Pei, E., Oveneke, M. C., & Jiang, D. (2017, October). Hybrid depression classification and estimation from audio video and text information. In Proceedings of the 7th annual workshop on audio/visual emotion challenge (pp. 45-51).

[10] Salekin, A., Eberle, J. W., Glenn, J. J., Teachman, B. A., & Stankovic, J. A. (2018). A weakly supervised learning framework for detecting social anxiety and depression. Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies, 2(2), 1-26.

[11] Al Hanai, T., Ghassemi, M. M., & Glass, J. R. (2018, September). Detecting Depression with Audio/Text Sequence Modeling of Interviews. In Interspeech (pp. 1716-1720).

[12] Tzirakis, P., Zhang, J., & Schuller, B. W. (2018, April). End-to-end speech emotion recognition using deep neural networks. In 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP) (pp. 5089-5093). IEEE.

[13] Goldberg, D. (2011). The heterogeneity of “major depression”. World Psychiatry, 10(3), 226.

[14]Lin, L., Chen, X., Shen, Y., & Zhang, L. (2020). Towards automatic depression detection: A BiLSTM/1D CNN-based model. Applied Sciences, 10(23), 8701.

[15] Abdoli, S., Cardinal, P., & Koerich, A. L. (2019). End-to-end environmental sound classification using a 1D convolutional neural network. Expert Systems with Applications, 136, 252-263.

[16]Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal processing letters, 24(3), 279-283.

[17] Choi, K., Fazekas, G., Sandler, M., & Cho, K. (2017, March). Convolutional recurrent neural networks for music classification. In 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP) (pp. 2392-2396). IEEE.

[18] Choi, K., Fazekas, G., & Sandler, M. (2016). Automatic tagging using deep convolutional neural networks. arXiv preprint arXiv:1606.00298.

[19] Müller, M. (2007). Dynamic time warping. Information retrieval for music and motion, 69-84.

[20] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[21] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

[22] Šrámková, H., Granqvist, S., Herbst, C. T., & Švec, J. G. (2015). The softest sound levels of the human voice in normal subjects. The Journal of the Acoustical Society of America, 137(1), 407-418.

[23] Le, D., Aldeneh, Z., & Provost, E. M. (2017, August). Discretized Continuous Speech Emotion Recognition with Multi-Task Deep Recurrent Neural Network. In Interspeech (pp. 1108-1112).

[24] Ma, X., Yang, H., Chen, Q., Huang, D., & Wang, Y. (2016, October). Depaudionet: An efficient deep model for audio based depression classification. In Proceedings of the 6th international workshop on audio/visual emotion challenge (pp. 35-42).

[25] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[26] Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition. J. Off. Stat, 6(1), 3-73.

[27] Cannizzaro, M., Harel, B., Reilly, N., Chappell, P., & Snyder, P. J. (2004). Voice acoustical measurement of the severity of major depression. Brain and cognition, 56(1), 30-35.