Depression Analysis and Detection by Social Media Posts

Ali Baran Taşdemir
Hacettepe University
Department of Computer Engineering
06810 Beytepe, Ankara
alibaran@tasdemir.us

***Abstract*—Depression (major depressive disorder) is a serious mood disorder that affects people's communication and the way that they interact with others. Depression is increasing alarmingly throughout the world, especially among the young population. In this paper, we are working on early risk detection by using the social media posts of users to detect signs of depression and anxiety. To do that, we collect text-based data from posts that are shared on social media platforms like Twitter and Reddit. We use a deep-learning-based method that labels posts by their risk of depression.**

***Index Terms*—depression detection, sentimental analysis, deep learning, neural networks**

I. Introduction

Depression (major depressive disorder) is a serious mood disorder. Individuals with depression can have symptoms like feeling constantly tired, unhappy, hopeless, and loss of interest. Depending on the depression severity the effects of symptoms can change. There are some common symptoms of depression but it affects people in different ways. Thus, individuals with suspicion of depression should see a doctor as soon as possible. The common belief is that depression is not a genuine health condition. But the truth is it is a serious mood disorder. In reality, around 280 million people suffer from depression worldwide[1]. Also, research shows that the cases of depression are increased as a post-effect of COVID-19 [6], [7].

The usage of social media is becoming rapidly common among people around the world. There are an estimated 4.7 billion social media users which are 59% of the total global population. The young population on social media is increasing too. Around %80 of people between 18-29 ages use at least one social media in the US[2]. Early risk detection systems are a growing need with the popularity of social media platforms. Fake news, scams, hate speech, and cyberbullying are some risk factors that we can detect by using Natural Language Processing and Machine Learning methods.

However, there are effective treatments for different kinds of depression. But it should be detected first to offer a treatment for the individual. Although, it is known that there are effective treatments, around 75% of people with depression or anxiety, never receive treatment. There are different reasons for that but one of the important reasons is the lack of trained professionals.

There are some methodologies that psychiatry doctors and clinics follow like consultation-based Diagnostics and the Statistical Manual of Mental Disorders (DSM). And there are ways of detecting depression by applying a questionnaire to the patient. These questionnaires evaluate the level of depression by scoring the answers given by the patient. There are common questionnaires widely used by clinics such as Beck Depression Inventory (BDI) [1], that can be applied to ages 13 to 80. Patient Health Questionnaire (PHQ-9) [9], which is a self-report measure that takes one to five minutes. It can be taken online[3] and it is available in multiple languages.

These questionnaires result in structured data that researchers in computer science and machine learning can benefit to develop a data-driven machine learning approach to detect depression. Zhao and Feng [14] used the BDI questionnaire for Chinese recruits to develop a machine-learning model to detect depression. Jain et al. [5] propose a machine learning method by using a modified version of the PHQ-9 questionnaire. Also, Kim and Lee [8] proposed a meta-analysis by using studies published about depression screening.

Questionnaires provide valuable data for machine learning applications but they have some disadvantages. These questionnaires should be handled by trained professionals. This is not eliminating one of the major problems for depression treatments. And this is not always available for most individuals with depression. And these questionnaires were designed for the cases at their time. But the world is changing so fast, and these are not always optimized for the detection of depression in the current conditions.

To address these problems, in this paper, we propose a data-driven, machine-learning-based methodology for the early detection of depression. By utilizing the computational power and machine learning algorithms, we can compensate for the need for the trained professional requirement for the prior analysis. Of course, the most reliable way to diagnose depression is still trained professionals but with an automated early detection system, we can increase awareness and give advice to users with risk to see a professional clinic. We

[1]https://www.who.int/news-room/fact-sheets/detail/depression
[2]https://www.pewresearch.org/internet/fact-sheet/social-media/
[3]https://www.phqscreeners.com/

use text-based social media posts to create a machine-learning model.

II. Related Works

We consider two main approaches to depression analysis by the data they use; clinic-based questionnaire data and social media.

A. Questionnaire-Based Methods

Depression is a mental disorder and it is diagnosed and treated by the trained-professionals which are psychiatry doctors and clinics. Their diagnosis is based on two main approaches; consulting, and a questionnaire-based approach. Since questionnaires provide easy-to-use structured data it is more suitable for machine-learning applications. But these methods need answers from the patient and this makes these methods weak candidates for a risk detection system. Because each individual should voluntarily answer questions to get results.

Zhao and Feng [14] propose a machine learning method to assess the severity of depression for Chinese military recruits by using traditional questionnaires. They are using the Chinese version of the Beck Depression Inventory-II (BDI-II). They evaluate BDI-II on randomly selected 1000 Chinese male recruits and by the answers to the questionnaire they train machine learning models like support vector machine (SVM), decision tree (DT), and neural network (NN). The experiment results show that SVM is the most suitable algorithm for their task.

Jain et al. [5], propose a machine learning method by using the PHQ-9 questionnaire applied to the students and their parents to detect depression. They modified the PHQ-9 by adding some features like age, sex, etc. They trained a machine learning algorithm to perform a multi-class classification to detect several levels of depression which are mild, moderate, moderately severe, severe, and none. Their experiment shows that their classifier achieved 83.8% accuracy.

Priya et al. [13], propose a machine-learning algorithm to predict anxiety, depression, and stress. They collect data from employed and unemployed individuals from different communities through the Depression, Anxiety, and Stress Scale (DASS 21) questionnaire. They set five levels of severity which are mild, moderate, moderately severe, severe, and none. They experiment on five different algorithms; decision tree, random forest tree, naive Bayes, support vector machine, and k-nearest neighbors. They found that the random forest tree is the most successful algorithm for the classification task. Also, they found that the most important question from DASS 21 for detecting depression is "I felt that life was meaningless".

B. Social Media Data

In 2022, there are an estimated 4.7 billion social media users, which is equal to 59% of the total population. People share their daily routines, thoughts, and feelings through social media. The usage of social media posts or connections (or any kind of information on social platforms) is more accessible in comparison to clinical questionnaires. Because these questionnaires required the assistance of a professional. And also people are hesitant to voluntarily see a psychiatrist, and self-diagnose or have a suspicion of depression due to different reasons like social pressure, financial issues, or not having access to a clinic or professional.

But social media is accessible to everyone and in most cases, it contains so much valuable information for the user. This aspect of social media makes it suitable for machine learning applications of depression detection. But these methods have some downsides for data collection. The data on social media is not always anonymous. And to collect data from a social media platform, the researcher needs to have the authorization and consent of the user. This approach has some legal constraints and while it makes users' information safer it also makes it difficult to collect data for research.

Mann et al. [12], propose a depression symptom detection system for higher education students by using a machine-learning model. They use social media posts on Instagram with a questionnaire composed of a set of questions to get information on social media usage. To label their data, they also use the BDI questionnaire and use four levels of depression; minimal, mild, moderate, and severe. They use Instagram image posts, image captions, and a combination of both separately and train three different classifiers. Their results show that textual information such as image captions contains more information about the detection of depression.

Li et al. [10], propose a method to detect signs of depression in dialog-based events. Since dialogs commonly occur on social media platforms (WhatsApp, Reddit) we consider that paper in this section. They use DAIC [3] and DailyDialog [11] corpora to train a model. These datasets are dialogs between two or more people in English and labeled by PHQ-9 scores. They are using a Bi-LSTM and RNN as a classifier and achieved a 70.6% F1 score.

Also, Jain et al. [5] experimented with machine learning classifiers by using data from Twitter and Reddit. They applied a step of preprocessing like stemming and using TF-IDF-generated features from textual data. They achieved 86% accuracy with social media posts.

III. Methodology

In this paper, we propose a deep learning network to detect depression signs of a social media user by using textual social media posts. We have a dataset $D = \left\{< U_{i_t}, U_{i_y} >\right\}_{i=0}^{N}$ consisting of N samples of textual posts t from users U who are already diagnosed with depression and others denoted as y. Our classification goal is to find a mapping of $\gamma(U_t) \rightarrow 0, 1$ for each textual post t of user U in the dataset D. The problem is a binary classification where label-0 denotes *"no-symptoms for depression"*, and label-1 denotes *"risk of depression"*.

A. Network Architecture

The proposed method learns to classify textual posts for a specific user U and detect signs of depression. For a post from a user U is classified by a binary label that denotes the

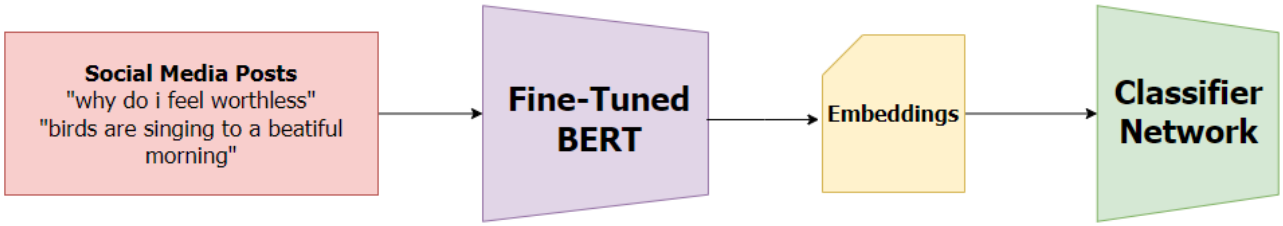


Fig. 1: The diagram of our classification framework. We use fine-tuned BERT model to generate embeddings of the textual data. And by using these rich representations we train a simple neural network to predict the labels of the posts.

risk of depression on the post. We utilize only the textual information which is the context of the post. To use that information on a deep learning model we need to vectorize the text by using feature extraction methods. This process is called embedding. After we transformed the textual data into a vector representation we pass this vector to a deep-learning classifier to predict the label of the post.

The classifier process every post in the dataset and predicts a label for the given sample. With that predicted label and the ground-truth rating, we calculate the binary cross-entropy error,

$$-\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

where N denotes the number of samples on the dataset, y_i is the ground-truth label for the sample i and $\hat{y}_i$ denotes the prediction of the model for the sample i.

*1) **Input**:* The network uses a tokenized text as input to the feature extractor section. The tokenization process results in three token inputs to the BERT. These are input ids, token type ids, and self-attention masks. Input id is a representation of words where each word in the vocabulary is represented by an integer. Token type ids denote the sequence that the token belongs in a sentence(input). Self-attention mask denotes which tokens should be attended to.

*2) **Feature Extractor**:* The feature extractor section of the model is responsible for generating representation vectors for each textual data. There are several techniques to represent text in a d-dimensional vector space. Our model, uses BERT [2] to get embeddings for textual data. The extractor generates 768-dimensional representation vectors.

*3) **Classifier Network**:* The classifier network is a simple fully connected network that is trained to classify posts, transformed into vector representations, if they have a risk of depression or not. The learning task is a ”binary classification”. We have two labels, ”risk of depression” and ”no risk for depression”.

*4) **Output**:* The output is a probability distribution between two labels. By using probability distributions we can utilize not only class predictions, but also probability values. For example, if a post is not classified as ”risk of depression” but has a close probability distribution that favors the ”no risk” class it may be still worth monitoring.

With the introduced neural network architecture above we tag posts by their risk factor for depression by using only textual data. There are no requirements for additional questionnaires or user information.

IV. Experimental Setup

To evaluate our depression detection system, we describe the dataset used and test setups that will show the performance of our detection system.

A. Dataset

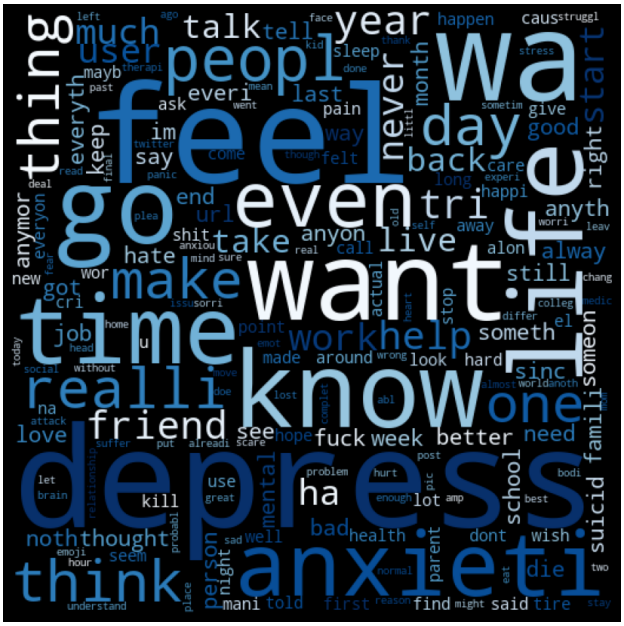
With the insight from previous works, we determined to use text-based data to train our classifier. Today’s biggest social media platforms which heavily consist of textual posts are Twitter and Reddit. That’s why we utilized posts from these platforms. The first dataset is **Twitter Dataset** [4] which is publicly available at Kaggle. The dataset consists of tweets from depression-diagnosed users and normal users which has around 10K samples. The second dataset is **Reddit Dataset** [5] which is publicly available at Kaggle. The dataset is collected from the social media platform Reddit. Reddit is based on subreddits that are specifically designed by topic. The posts which are diagnosed as depression are collected from depression subreddits. The dataset includes 7650 posts from users.

We combined these two datasets and created a dataset with 18045 samples. 11900 samples labeled as label-0 which is ”no-symptoms for depression” and 6145 samples labeled as label-1 which is ”risk of depression”.

The dataset includes only textual data. As an initial look at the dataset, we inspect the dataset by using some statistical methods. By using word clouds for two types of posts, we analyzed the common words used in both types of posts (Figure 2). Posts that include a risk of depression commonly contains words like, ”depression”, ”anxiety”, and ”feel”. The other posts, without any symptoms of depression, contain words like, ”day”, ”love”, ”good”, ”thank”, ”work”, etc. This

[4]https://www.kaggle.com/datasets/gargmanas/sentimental-analysis-for-tweets

[5]https://www.kaggle.com/datasets/infamouscoder/depression-reddit-cleaned



(a) Label-1, posts with the symptoms of depression

(b) Label-0, posts without the symptoms of depression

Fig. 2: Word clouds of two types of posts.

also shows us that, people tend to share their feelings about their lives and their experiences on social media. Even if they have or do not have the symptoms of depression they describe and tell their emotions and experiences. But we can observe that people with depression are more likely to share more negative posts than others while others commonly post some positive words like ”good”, ”thank”, or ”love”.

B. Experiments

We set up experiments to 1) show the accuracy of our classification for detecting signs of depression, 2) to show the performance of modern deep learning-based feature extraction methods (BERT and LSTM), 3) to compare our method with other works to detect depression which are using questionnaire data.

To compare results we will use several evaluation metrics. These are F1-score and classification accuracy. The F1 score is the harmonic mean of precision and recall values. The F1 score provides a better evaluation of the model if the detection is important. For example, if the model labels every post as ”risk of depression”, the model's recall will be %100 but the precision will be low. The model tags everyone with ”risk of depression”.

1) Feature Extraction Techniques: In this experiment, we used different methods for embedding the textual data to evaluate their performance and decide which technique is a better fit for the task.

The proposed architecture includes a feature extraction section. The quality of the embedding vectors gathered from this section is crucial for the classification task. BERT [2] is a transformer-based model developed by Google in 2019. BERT is one of the best-performing models for various NLP downstream tasks. Thus, BERT is the first candidate for the feature extractor section. We use a pre-trained version of BERT from Google Hugging Face. And generated embedding vectors with a length of 768.

The second alternative is using a pre-trained BERT model and training again with our dataset. This process is called ”fine-tuning”. As the second method, we train the same pre-trained BERT model and continue training by 3 epochs with our dataset.

The last alternative is training an embedding network from scratch (See Figure 3. We used LSTM networks [4] for this task. Long-Short Term Memory networks are strong networks for sequenced data like text. The feedback connections make the network well-suited for classification, regression, or segmentation tasks for sequence data. For our experiment, we want to utilize LSTM networks to generate an embedding vector from our textual data.

2) Comparison with Baseline Methods: In this experiment, we experiment on the second section of our proposed architecture. After we obtain the vector representation of the textual data we use a classifier to detect labels of the sample. This experiment asks that, *which algorithm or model is more efficient for our task?*

We experiment on models with different hidden layers and complexity. We also experiment with some traditional machine learning algorithms.

The first model is the support vector machine (SVM) algorithm. We use the SVM algorithm with radial basis function kernel (RBF) and regularization parameters as 1. In this experiment, we use Python SkLearn implementation of the SVM algorithm.

The second model is a Fully Connected Network. If the

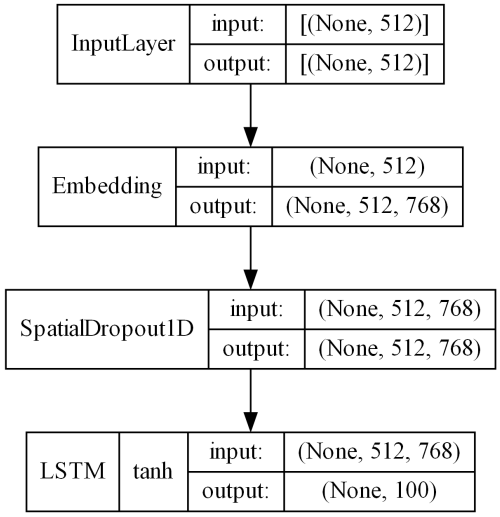


Fig. 3: LSTM embedding network

embeddings from the feature extractor are the high quality we should be able to distinguish data samples by a simpler neural network. In this experiment, we are evaluating different sizes of neural networks and our feature extractors' performance.

3) Comparison with Literature: As we mentioned in Section II, several methods exist for depression detection. We compare our results with questionnaire-based methods. These methods require a professional depression questionnaire and volunteers that answer questions. The dataset for these methods is structured. But the sample size is smaller than other data-driven approaches.

Also, another type of approach is utilizing social media and user information. This approach is data-driven and utilizes textual data and other auxiliary information like user profiles, recently shared posts, etc.

We are using these two works to compare with our model; Mann et al. [12] and Jain et al. [5].

V. Results

As we described in Section IV-B, we set up three different experiments. We merged two datasets (detail Section IV-A), and in total, we have 18045 labeled social media posts from users. For experiments, we use a PC with 32GB of memory, a 3.6 GHz AMD Ryzen 5 processor, and an NVIDIA GeForce GTX 1660 GPU with CUDA support. All codes are written in Python and available at https://github.com/alibtasdemir/DepresNet.

A. Feature Extraction Techniques

For this experiment, we test different feature extraction techniques for the initial part of the model. We use a pre-trained BERT model, fine-tuned version of BERT on our dataset, and train an LSTM embedding network from scratch.

The BERT model has around 110 million trainable parameters and the model itself is computationally costly. But BERT is a successful and strong model that is widely used amongst researchers, its pre-trained version is available online. Google Hugging Face shares a BERT version trained on Toronto Book Corpus [15] and Wikipedia.

The other feature extractor is an LSTM network that we train from scratch and generate embedding based on our dataset. Our embedding network (Figure 3) has one Embedding layer and an LSTM layer. The model has 39M trainable parameters. The model has trained 10 epochs on the dataset.

And the last feature extractor is a finetuned version of the BERT model on our dataset. We trained the BERT model on our dataset for 3 epochs and generated embedding vectors by using the weights of this finetuned BERT model.

As we see in Figure 4, we can see that the embeddings for all methods produced distinguishable representation vectors. But there are some visible differences in techniques. BERT embeddings have some clear clusters but there are too many overlapping data samples that have different labels. This means that embeddings are not ideal.

On the other hand, LSTM and finetuned BERT embeddings produced representation vectors that can easily be separated from each other. While finetuned BERT embeddings can be separated by a linear function, LSTM network embeddings can be separated by more complex functions.

B. Comparison with Baseline Methods

In this experiment, we compared the performances of our baseline classifiers and feature extractors. We trained 12 different models by combining three feature extractors (BERT, Finetuned BERT, and LSTM) and four different classifiers.

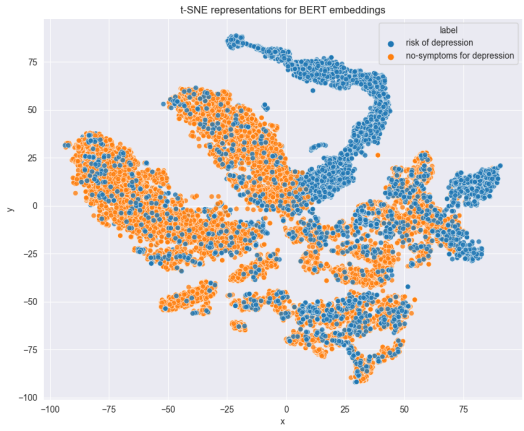
We used one traditional machine learning algorithm SVM and three fully connected networks with different hidden layers and parameters. We configured fully connected networks by increasing their size and giving them more depth. And we use three different models which are small, medium, and large.

The small model has two fully connected layers. The initial layer uses the reLu activation function and the final layer uses the sigmoid activation function. We use one dropout layer in between the first and last fully connected layer. The model has 24.5K parameters (3.3K for LSTM embedding because of the embedding vector size).

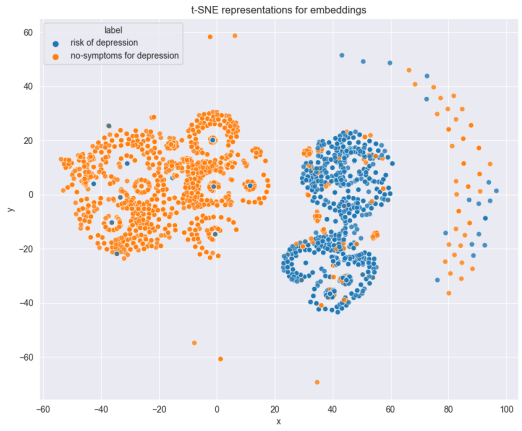
The medium model has three fully connected layers. All layers except the last one use the reLu activation function and the final layer use the sigmoid activation function. We use the dropout layer between fully connected layers. And the model has 205K parameters (34K for LSTM embeddings).

The large model has four fully connected layers. All layers except the last one use the reLu activation function and the final layer use the sigmoid activation function. We use the dropout layer between fully connected layers. And the model has 1M parameters (374K for LSTM embeddings).

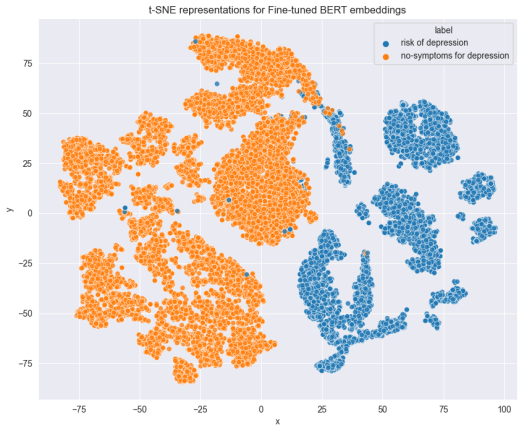
The results using these models are shown in Table I. The first observation about the results is that finetuned BERT and LSTM embeddings outperform the BERT embeddings. This observation supports our comments on the previous experiment. There is an improvement in performance for LSTM embeddings by increasing fully connected networks. The F1



(a) t-SNE representation for BERT embeddings



(b) t-SNE representation for LSTM network embeddings



(c) t-SNE representation for finetuned BERT embeddings

Fig. 4: t-SNE representations for feature extractors

score for LSTM + Small FCN is 0.968 and 0.978 for the LSTM + Large FCN. There is a 0.01 points increase in terms of F1 score but we increase the number of parameters $\times 100$.

This experiment shows us that, even with a small classifier we can get the best results. And also this experiment supports our comments on the embeddings from the previous experiment. Quality embeddings are easily separable from each other.

C. Comparison with the Literature

As the last experiment, we compare our best-performing model from the previous experiments to the selected works from the literature (See Section II and IV-B). We compare our work with Mann et al. [12] and Jain et al. [5].

Jain et al. [5] use both questionnaire-based and data-driven-based approaches. They apply a professional questionnaire to the high school students and also retrieve their social media posts with their auxiliary information like age, sex, regularity in school, etc. They train a set of traditional machine learning

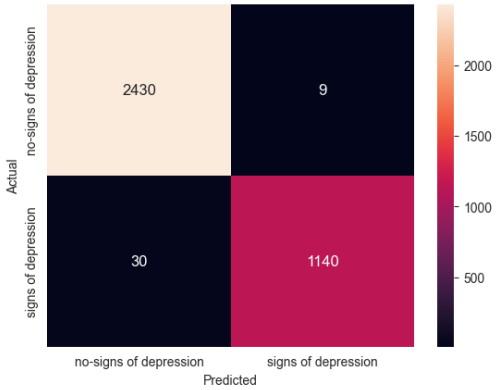


Fig. 5: Confusion matrix for predictions of Fine Tuned BERT + Small FCN

	Accuracy	F1 Score	# of Parameters
BERT + SVM	0.878	0.878	-
Fine Tuned BERT + SVM	0.989	0.989	-
LSTM + SVM	0.968	0.968	-
BERT + Small FCN	0.866	0.866	24674
Fine Tuned BERT + Small FCN	0.989	0.989	24674
LSTM + Small FCN	0.978	0.978	3298
BERT + Medium FCN	0.862	0.862	205154
Fine Tuned BERT + Medium FCN	0.989	0.989	205154
LSTM + Medium FCN	0.978	0.978	34146
BERT + Large FCN	0.843	0.843	1058146
Fine Tuned BERT + Large FCN	0.988	0.988	1058146
LSTM + Large FCN	0.978	0.978	374114

TABLE I: Baseline methods with three different text embedding techniques and 4 different classifiers. SVM denotes the support vector machine, Small FCN is a fully connected network with two layers, Medium FCN is a fully connected network with three layers and has more parameters than the small model, and the Large FCN is a fully connected network with four layers.

	Precision	Recall	F1 Score
Jain et al. [5]	0.876	0.821	0.826
Mann et al. [12]	0.690	0.920	0.790
Fine Tuned BERT + Small FCN	**0.986**	**0.989**	**0.992**

TABLE II: Comparison of our architecture by recent works.

algorithms. They obtain the best performance from Logistic Regression Classifier.

Mann et al. [12] use a hybrid approach by collecting data from social media and applying a questionnaire to the higher education students to get their evaluation. They use a multimodal model which uses both images from posts and the textual context (like the caption).

The results show that our outperforms the other two methods. Using professional questionnaire data can be beneficial, we show that without applying this kind of voluntary basis actions, we can tag a post by looking at the textual context.

VI. Conclusions

In this work, we propose a data-driven deep learning-based approach to detect signs of depression on textual social media posts. We compared our results with recent works using different approaches. The results show that data-driven deep learning methods can detect signs of depression by using only textual data.

As future work, the dataset might be expanded and the problem can be transformed into regression or multi-class classification to detect different danger levels of depression. Also utilizing the user information besides the textual post can increase the accuracy of the prediction.

References

[1] Aaron T Beck, Robert A Steer, Gregory K Brown, et al. Manual for the beck depression inventory-ii. *San Antonio, TX: Psychological Corporation*, 1(82):10–1037, 1996.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Jonathan Gratch, Ron Artstein, Gale Lucas, Giota Stratou, Stefan Scherer, Angela Nazarian, Rachel Wood, Jill Boberg, David DeVault, Stacy Marsella, et al. The distress analysis interview corpus of human and computer interviews. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA LOS ANGELES, 2014.

[4] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[5] Swati Jain, Suraj Prakash Narayan, Rupesh Kumar Dewang, Utkarsh Bhartiya, Nalini Meena, and Varun Kumar. A machine learning based depression analysis and suicidal ideation detection system using questionnaires and twitter. In *2019 IEEE Students Conference on Engineering and Systems (SCES)*, pages 1–6. IEEE, 2019.

[6] Natarajan Kathirvel. Post covid-19 pandemic mental health challenges. *Asian journal of psychiatry*, 53:102430, 2020.

[7] Jagdish Khubchandani, Sushil Sharma, Fern J Webb, Michael J Wiblishauser, and Sharon L Bowman. Post-lockdown depression and anxiety in the usa during the covid-19 pandemic. *Journal of Public Health*, 43(2):246–253, 2021.

[8] Sunhae Kim and Kounseok Lee. Screening for depression in mobile devices using patient health questionnaire-9 (phq-9) data: A diagnostic meta-analysis via machine learning methods. *Neuropsychiatric Disease and Treatment*, Volume 17:3415–3430, Nov 2021.

[9] Kurt Kroenke, Robert L Spitzer, and Janet BW Williams. The phq-9: validity of a brief depression severity measure. *Journal of general internal medicine*, 16(9):606–613, 2001.

[10] Chuyuan Li, Chloé Braud, and Maxime Amblard. Multi-task learning for depression detection in dialogs. *arXiv preprint arXiv:2208.10250*, 2022.

[11] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*, 2017.

[12] Paulo Mann, Aline Paes, and Elton H Matsushima. See and read: detecting depression symptoms in higher education students using multimodal social media data. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 440–451, 2020.

[13] Anu Priya, Shruti Garg, and Neha Prerna Tigga. Predicting anxiety, depression and stress in modern life using machine learning algorithms. *Procedia Computer Science*, 167:1258–1267, 2020.

[14] Mengxue Zhao and Zhengzhi Feng. Machine learning methods to evaluate the depression status of chinese recruits: A diagnostic study. *Neuropsychiatric disease and treatment*, 16:2743, 2020.

[15] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015.