Analysis of medical conversations for the detection of depression

Ioana-Raluca Zaman
University Politehnica of Bucharest
Splaiul Independentei, Bucharest, Romania
ioana.raluca.zaman@gmail.com

Ștefan Trăușan-Matu
Politehnica University of Bucharest
313 Splaiul Independentei, Bucharest, Romania
and
Research Institute for Artificial Intelligence
and
Academy of Romanian Scientists
stefan.trausan@cs.pub.ro

Traian Rebedea
Politehnica University of Bucharest
313 Splaiul Independentei, Bucharest, Romania
traian.rebedea@cs.pub.ro

ABSTRACT
Depression is becoming more common, affecting people's lives with symptoms such as: low mood, fatigue, insomnia, restlessness, worthlessness, weight changes or even recurrent thoughts of death or suicide [9]. The objectives of this paper are to identify linguistic features that serve as indicators of depression and to develop a classification system using transcripts from real patients with potential depression. To achieve this, a series of experiments was conducted, using the extracted linguistic features and Deep Learning (DL) models for Natural Language Processing (NLP). The experiments include a range of approaches, including the fusion of text features with numerical features, the division of the dialogues into specific formats and the usage of Zero-Shot Learning techniques. Regarding the data, the Distress Analysis Interview Corpus (DAIC-WOZ) [14] from University of Southern California dataset was utilized, along with additional data generated using ChatGPT.

Author Keywords
Natural Language Processing; Deep Learning; Transformers; Transcripts Analysis; Linguistic Features; Depression Detection

ACM Classification Keywords
I.2.7 Natural Language Processing

DOI: 10.37789/rochi.2023.1.1.4

INTRODUCTION

The incidence of depression among adults is approximately 1 in 15 individuals annually, with approximately 1 in 6 people experiencing depression at some point in their lifetime. According to a study conducted by Our World In Data, the estimated global population affected by depression is around 3.4%, corresponding to approximately 264 million individuals worldwide [10]. Automatic detection and analysis of mental illnesses would help in terms of both performance (i.e., diagnosing and monitoring patients would take less time) and availability (i.e., more patients could afford it). The analysis of medical conversations (e.g., answers at questionnaires or description of certain images by patients) will provide more information and context about a person's mental state than the analysis of a single element (e.g., social media post), this contributing to a more accurate automatic detection of depression.

Given the multifaceted nature of depression, characterized by varying symptomatology across individuals, this research paper endeavors to investigate depression within dialogues to acquire a comprehensive contextual understanding. Furthermore, the data is gathered in a controlled environment and dynamic context. Specifically, the conversations were between a patient and an agent controlled by a human, creating an environment designed to enhance the detection of depression-related indicators. A significant proportion of the existing literature on depression detection focuses on studying it in social media posts [4, 26], the analysis of real medical data creating challenges in both data processing and acquiring appropriate access to such data.

STATE OF THE ART

Linguistic Features of Depression

Tølbøll elaborated a complex review of depression-specific linguistic features presented in 26 papers [28]. These studies are based on both static (e.g., essays or social media posts) and dynamic (e.g., therapy sessions or questions answering) contexts. The main aspects analyzed are first-person pronouns and words related to positive and negative emotions. For these aspects, the difference between the depressed and non-depressed individuals is quantified by Cohen's *d* and Pearson's *r* [28]. Cohen's *d* is a coefficient used to measure the effect size of the difference between two means, resulting in values between 0 and 1 [6]

and Pearson's *r* is a coefficient used to measure the strength between two variables having values between -1 and 1 [7]. One conclusion is a Cohen's *d* of 0.44 (i.e., a medium effect) and a Pearson's *r* of 0.19 (i.e., a positive correlation) between first-person singular pronoun usage and depression.

Regarding the use of emotions related words, Tølbøll [28] identified differences with a Cohen's *d* of 0.72 (i.e., a strong effect) between negative words and depression and a Pearson's *r* of -0.21 (i.e., a negative correlation) between the use of positive emotion words and depression. The results of this method can be utilized to find the most popular topics in discussions.

Wolohan et al. [29] analyzed Reddit [25] posts from 12,106 users and using Term Frequency - Inverse Document Frequency (TF-IDF) selected the 100 most in use words. They concluded that depressed users utilized more therapy and medications terms, the authors dividing them into three categories: therapy (e.g., therapist), depression-specific medications (e.g., Xanax) and alternative medications (e.g., Kratom). Other leading topics were games and manga characters (e.g., Goku, Nyx) or Reddit-specific matters. Another observation was that depressed user posts are written to the 2nd person singular, referring to Reddit readers as "you" (e.g., yeah, thank you). Also, in this study it was reconfirmed that depressed users talk more about themselves, using more first-person pronouns.

Related Work

Dinkel et al. [11] presented a multi-task approach (i.e., combining binary classification task with the depression severity task) using the DAIC-WOZ dataset and achieved state-of-the-art results, reaching a F1 score of 0.84. Furthermore, starting from the idea that word-embeddings represent the context poorly, they [11] illustrated a comparison between word-embeddings and sentence-level embeddings and pretrained embeddings vs. freshly trained embeddings. Additionally, a comparison between four pooling functions (Time, Mean, Max and Attention) was illustrated in the paper. The architecture of the model contained 3 layers of the Bidirectional Gated Recurrent Unit (BGRU), each of them being followed by a Dropout layer with a 20% probability and a Pooling layer having as possible options, the previously mentioned pooling functions.

Regarding the first comparison, Dinkel et al., [11] utilized embeddings from two models, more precisely: fastText [3] and Word2Vec [21]. After running all the experiments, it was confirmed that the pre-trained embeddings have better results. One characteristic highlighted in this paper is that the data utilized is *spare data* (i.e., transcripts from clinical conversations). Due to the fact that the labels are assigned per an entire dialog, the authors proved that sentence-level embeddings had improved results than word-embeddings. The sentence-level embeddings performed better than word-embeddings for both fastText and Word2Vec. After testing all four pooling functions on all the four models, the authors concluded that ELMo had the most significant improvements by adding a pooling layer, especially for the Mean function. That function also performed the best for BERT, while fastText and Word2Vec had better results by using the Attention function.

Alhanai et al. [2] utilized the same dataset for detecting depression. Both audio and text features were used as input in a Long Short-Term Memory (LSTM) neural network, but the focus will be only on the text features. Because the corpus consists of interviews in which the patient answers questions, the authors [2] took into account the type of question asked in one of their approaches. Each participant *i* was asked a subset of questions q_i of a set of predefined questions *Q*. The two main ideas were: (1) select only the most *k* informative questions and (2) assign them weights, for that, the authors utilized a matrix-based representation. In order to measure the relevance for each question, the authors evaluated the model in the cases where only the specific rows of a certain question from the matrix were used and save the results as follows: *c(j)* – the performance obtained when only the rows from the question $j, j \in \{1 : Q\}$ were used. One experiment presented in the paper utilized a LSTM model and had two approaches, a sequence modeling and a multi-model one. In the first approach, the text features were utilized separately from the audio ones. The F1 scores for the text and audio features were 0.67 respectively 0.63. In the multi-model approach the results improved, reaching a F1 score of 0.77, a precision of 0.71 and a recall of 0.83. The topology of the multi-model consisted of two LSTM branches (i.e., one for audio features and one for textual features) and the results of each branch were concatenated and used as input for a feedforward layer.

PROPOSED SOLUTION

In this section the methodology of the analysis and the experiments will be presented. Several aspects of the proposed approach will be discussed in detail, each in a separate section as follows. In the first section, the corpus utilized will be described and aspects of it will be analyzed. Following, the numerical features extracted from the dataset and their analysis will be presented. The next section will illustrate the experiments carried out. Ultimately, the last section will present the method by which new conversations were generated with ChatGPT.

Corpus

One method used by psychiatrists to monitor and diagnose patients is through questionnaires. By associating certain scores with participants' answers to questions, their mental condition can be measured by an exact (i.e., numerical) method. One example of this type of questionnaire is Patient Health Questionnaire (PHQ-8) [19] which is also

used in classification of the participants from DAIC-WOZ corpus. The questionnaire consists of eight questions, each of them referring to a symptom (e.g., poor appetite or overeating). Those who participate in completing the questionnaire have to quantify how much of each symptom bothered them in the last two weeks. The answer can be: *Not at all*, *Several days*, *More than half the days* or *Nearly every day*, each of them having an associated score of 0 to 3. At the end, the scores from all the questions are accumulated and based on this final score the patient is diagnosed. This score is used to determine the patient's level of depression and has a value between 0 and 24. Using these scores both a binary classification (i.e. *non-depressed* and *depressed*, score < 10 respectively score > 9) and multiclass classification (i.e. *none/minimal*, *mild*, *moderate*, *moderately severe*, *severe* having as interval separation points: 5, 10, 15 and 20) can be performed. Consequently, the participants from categories *none/minimal* and *mild* (i.e., the first two classes from multiclass classification) are considered *non-depressed* in the binary classification.

A note that should be mentioned is that we gained access to the dataset through a signed agreement. The corpus contains 189 conversations conducted by a virtual interviewer called Ellie which is controlled by a human, the group of participants consisting of 87 women and 102 men. Each participant is identified by an ID to which they are associated: gender, PHQ-8 score together with responses to the questions from the PHQ-8 questionnaire. The dataset is divided into a training set, validation set and test, each having 107, 35 respectively 47 dialogues. Figure 1 shows that most patients suffer from mild depression, while the few suffer from severe depression. In terms of binary classification the dataset is not balanced, it contains 133 non-depressed and 56 depressed participants.

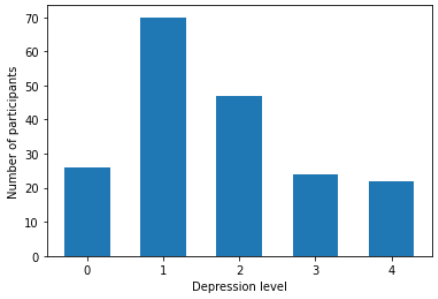

Figure 1. Distribution of participants by the level of depression from the PHQ-8 questionnaire

Extraction and Analysis of Linguistic Features

Before preprocessing, all data was saved in a dataframe which contains the following columns:

- personId – the unique identifier for each participant
- question – the question asked by the agent Ellie
- answer – the message said by participant to the corresponding question asked by Ellie

In order to improve the results, preprocessing techniques such as: tokenization, conversion to lowercase and removal of stop words are applied to the data. Given that the texts used are transcripts, some words require certain transformations. For instance, "l_a" refers to Los Angeles; these types of words are replaced before analysis. Moreover, in this step, contractions (e.g., can't, I'm) are replaced with the two-word version. The final step is stemming, this being the process in which each word is replaced with its base (e.g., flying → fly).

Before computing any feature, the train dataframe was divided into two dataframes: one containing only the depressed participants and one only with the non-depressed participants. After that, all replies from the participants were concatenated in order to compute a general analysis based on all the replies from the depressed or non-depressed participants. To analyze the Part of Speech tagging (POS), for that we used the pos_tag package from Natural Language ToolKit (NLTK) [22] Python library. The methods from this package return a list of pairs (i.e., the first element is a word and the second is the part of speech of the word). In the analysis of POS, one difference is that depressed participants make more use of VGB (i.e., verb, present participle or gerund), while non-depressed utilize more NNP (i.e., noun, proper, singular). Another distinction is that non-depressed participants used more prepositions or conjunctions (e.g., near, among). Regarding verbs, depressive participants tend to use more past participle (e.g., unsettled) and present tenses.

For extraction of sentiments and emotions, we utilized the NRCLex [23] sentiment lexicon. This lexicon consists of a list of words and the emotions and sentiments they are associated with. These are: *anticipation*, *anger*, *disgust*, *fear*, *joy*, *sadness*, *surprise* and *trust* as emotions and *negative* and *positive* as sentiments. Other features were computed using the TextBlob [27] package, these being *polarity* and *subjectivity*. After analyzing the score extraction results, we have concluded that depressed participants had higher scores for: fear (0.56 compared to 0.49), sadness (0.66 compared to 0.59) and negative emotions (0.1 compared to 0.088), while the non-depressed participants had higher scores for polarity (0.164 compared to 0.144). Another observation was that, in both categories (depressed and non-depressed), males had lower scores than females for fear and sadness.

The main source of extracting features was pyConverse [24], a Conversational Transcript Analysis Python library. Using this package, 16 transcript specific features were extracted, for instance: total_time (the entire duration of the conversation), total_replies (the total number of replies), total_replies_Ellie (the total number of replies of the agent), total_replies_participant (the total number of replies of the

participant) or empathy. In addition to all these features, word_speed (i.e., the number of words per second) and char_speed (i.e., the number of chars per second).

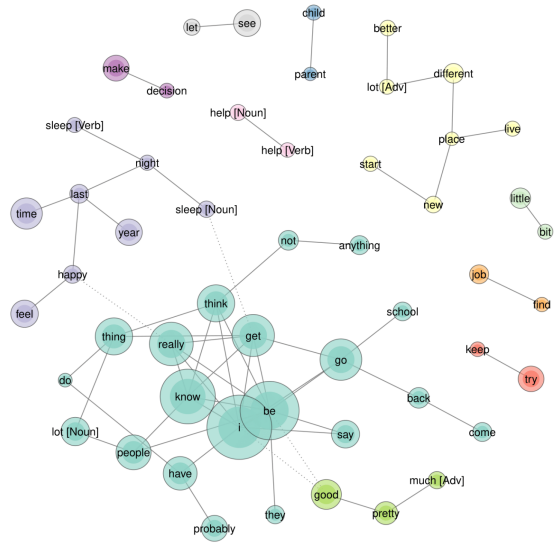

Figure 2. The most common words used in the dialogues of depressed patients as subgraphs

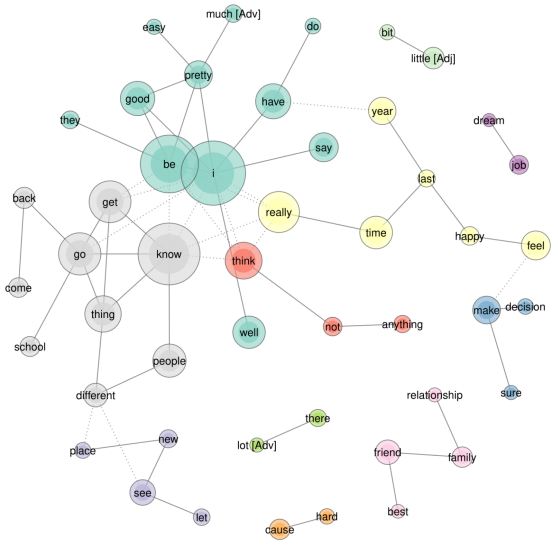

Figure 3. The most common words used in the dialogues of non-depressed patients as subgraphs

Other analyzed linguistic features are the top topics in conversations and the most relevant words. To compare these characteristics between depressed and non-depressed participants, only the replies from them (i.e., the participants) were extracted and used as input for the KH Coder [17] open-source tool with which the Co-Occurrence Networks of words were plotted. Words associated with circles connected and having the same color, have similar appearance patterns and the size of the circles is proportional to the frequency of occurrence of the respective words. Regarding the top words, the non-depressed participants' plot contains a subgraph consisting of *best-friend-family-relationship* while the depressed participants' plot contains only a subgraph consisting of *child-parent*. Another difference is that the depressed participants' plot has a subgraph composed of *sleep[Verb]-night-sleep[Noun]*, insomnia or hypersomnia being known problems for depressed patients. Furthermore, in terms of professional life, the non-depressed participants' plot contains the subgraph *dream-job*, while the other plot has the subgraph formed of *find-job*. A dissimilarity is also the fact that the depressed participants' plot contains a subgraph composed only of *help[Noun]-help[Verb]*. The graphs displaying the most frequently used words by participants with depression and those without depression are presented in Figure 2 and Figure 3, respectively.

Experiments

Baseline approaches

The first approach is one based on Deep Learning models, using MentalBERT and MentalRoBERTa [16]. These two models are specifically created for the detection of mental health problems by pretraining with posts from subreddits as "r/depression" and "r/mentalhealth" [16]. After the preprocessing step, all conversations were saved in a dataframe in which on each row is a question addressed to the participant and the answer to it. Due to the fact that the conversations are based on a series of questions asked by an agent chosen from a predefined set, the question and the answer were concatenated in the following format *'question?[QASEP]answer'*, mentioning that *[QASEP]* was added as a special token for the two models (e.g., 'where are you from originally?[QASEP]i am from los angeles california'). If one question was answered with multiple replies, these were concatenated. The two models were fine-tuned for 20 epochs for the two classification tasks using the following hyperparameters: optimizer - Adam, adam_epsilon - 1e-8, learning_rate - 5e-5, weight_decay - 0.01. Due to the fact that the results obtained by the model are per question-answer pair (QA pair), for classification of the entire conversation, the average of all the classification results of each QA from each dialogue was computed.

Additionally, the conversations were divided into chunks. One of the main limitations in the approach of dividing the dialogues into QA pairs is the fact that in the classification of a pair, only those two replies are used, eliminating a large part of the context provided by the entire dialogue. Taking into account that a MentalBERT is based on the Transformer architecture, it has a limit of 512 tokens [8] and the fact that an entry (i.e., represented by a QA pair)

has under 100 tokens, most of them having even under 50 tokens, more context could be gained by dividing dialogues into chunks of a maximum of 512 tokens. In the previous experiments, a conversation was divided into approximately 40 QA pairs, in the case of division in chunks, a conversation will contain not more than 10 chunks. Moreover, a chunk will consist of several QA pairs separated by a special token, more precisely [PSEP]. Therefore, a chunk will have the following format: *"[PSEP] question1? [QASEP] answer1 [PSEP] question2? [QASEP] answer2 [PSEP]..."*. While the application of this approach yielded unsatisfactory results when applied to text input, it had favorable outcomes when implemented with Zero-Shot models.

The second approach is features-based, using linguistic characteristics extracted and CatBoostClassifier [12] models. Before being used as input, the features were scaled with MinMaxScaler and the categorical variables (e.g., overall_participant and overall_Ellie) were converted into dummy variables with get_dummies for pandas library. The best results from all these experiments are illustrated in Table 1.

Fusion of text and numerical features

In this subsection will be presented two methods by which the fusion between the text and the previously extracted numerical features is implemented. The first approach is based on the idea of calculating the weighted average between the results obtained after the classification of the text using MentalBERT and the classification using numerical features using CatBoost. In the second approach, a Multimodal Transformer model [15] was used in order to combine the text with the numerical features directly.

To optimize the performance using the predicted scores from CatBoost and MentalBERT, it was needed to determine two optimal values. These variables were computed by conducting experiments on the validation set. The first one is the variation of the proportion between the two scores, more precisely, the weight for the scores from MentalBERT (the other weight is complementary and thus the weighted average can be calculated). The second one is the threshold for the weighted average result (i.e, the threshold by which the final score - weighted average is converted to a label). Both values were searched in the interval [0, 1] with a step of 0.05. This method of finding the optimal threshold was also used in the experiment where the input was represented only by text features. In the experiment where only numerical features were used, there was no need, because they were extracted per an entire dialogue.

The second approach was based on a Multimodal Transformer model [15]. This kind of model is used for classification or regression tasks and operates on the idea of unifying non-text features (i.e., numerical or categorical) with text features. First and foremost, the text input is processed by a Transformer, in this case the chosen model was MentalBERT. Following that, the output from the neural network is joined with the numerical features in a *Combining module*. One of the most relevant training arguments of the model is *combine_feat_method*, this represents the mode in which the three types of features are merged into the module. The argument has seven possible values and the method with the best performance was *"individual mlps on cat and numerical feats then concat"* the results being shown in Table 1 for the Multimodal experiment.

Zero-Shot approach

Zero-shot learning techniques have gained significant attention in the field of Machine Learning (ML), offering solutions for classifying unseen classes or categories when labeled training data is unavailable, this being a common problem in real-world scenarios [5]. The model utilized in this paper is Task-aware Representation of Sentences (TARS) [1], which facilitates Zero-Shot learning by leveraging task-specific information during pre-training to generate contextualized representations that enable effective classification even with limited labeled data.

In order to use the TARS model, a TARSClassifier object was loaded using the flair [13] Python package, the model used was *'tars-base'*. For the Zero-Shot approach the only argument needed for classification was a list containing the classes, these being *'depressed'* and *'non-depressed'*. As in the previous experiments, each section of a conversation (i.e., QA pair or chunk) was classified individually and then the average of all sections per dialogue was calculated, resulting in a score in the interval [0, 1]. Due to the fact that in the first experiment (i.e., in which the input was divided into QA pairs and the task was Zero-Shot with depressed and non-depressed classes) the final scores had small values, the best threshold was chosen in the interval [0, 0.2] with a step of 0.01 and for the experiment in which the input was divided into chunks, the best threshold was chosen in the interval [0, 1] with a step of 0.05, both on the validation set.

In the Zero-Shot Sentiment task, as classifier a model from flair package specialized in sentiment classification was used. The model was trained on movie and product reviews and for a sentence, it predicts two scores and a final label which can be *positive* or *negative* [1]. For each section of a dialogue a score was associated, 1 if the label *negative* was associated with it and 0, otherwise. The idea from which this experiment started was that, if a dialogue is labeled as *positive*, it would be from a patient who does not suffer from depression, on the other hand, if it is labeled *negative*, then it is from a patient suffering from depression. Also, in this experiment, for both approaches (i.e., input divided into QA pairs or chunks), the threshold was chosen in the interval [0, 1] with a step of 0.05.

Generative Data Synthesis with ChatGPT

As mentioned previously, the dataset is not balanced, the depressed class containing less than a third of all data. To mitigate the impact of this imbalance, 100 new dialogues were generated with ChatGPT, all classified as depressed. One method to create new data is to start from a static textual data (e.g., articles, narratives etc.) and apply certain transformations to it in order to convert it into dynamic textual data (e.g., dialogues).

The methodology on which the new data were generated was inspired by a paper that illustrates the creation of the SODA (SOcial DiAlogues) [18]. The process of the generation of dialogues is depicted in Figure 4. The first step of the generation is acquiring the triplets from the Knowledge Graph $ATOMIC^{20}_{20}$ which can be accessed on the official page Mosaic Knowledge Graphs (Allen Institute of AI). In order to create all the 100 samples, 34 triplets were used, for each triplet being generated three narratives and for each narrative, one conversation. A triplet is represented by a tuple consisting of a Head, a Relation and a Tail. All tuples have as Head, a general entity named *'PersonX'*, as Relation *'Feels Depressed'* and the Tail corresponds to the cause of depression (e.g., "Breakup or divorce", "Financial difficulties") or to a symptom of the disease (e.g., "Lack of appetite", "Difficulty sleeping"). In order to generate more triples using ChatGPT, the first step was to give as examples to the model the triples from the Knowledge Graph. Following that, for each triplet, three narratives were generated and after generating the narrative, for each one of them, a conversation was created. All the conversations, their labels and their conversion into QA pairs can be accessed on Google Drive[1].

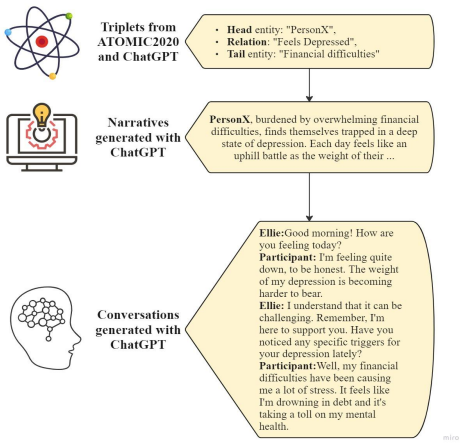

Figure 4. An illustration of the conversation generation process.

[1]https://drive.google.com/drive/folders/1zAH3wtHskzgbIplwQHvEvgRSMhPJMqop

RESULTS

Regarding the results, the most relevant metrics used were macro F1 and Area Under the Curve (AUC) and for the detection of the best thresholds, AUC was utilized as the main metric and in case of equality then macro F1. In the first experiments, for almost all tasks and metrics MentalBERT outperformed MentalRoBERTa; one explanation for this fact is that MentalRoBERTa is too robust for the small dataset. For the binary classification, MentalBERT had a macro F1 score of 0.57 compared to 0.56 and for the multiclass classification MentalBERT had a score of 0.374 compared to a score of 0.286, all these results being from the test set. The first model also had better results on the validation set.

Regarding the features-based approach using the CatBoost model, for the binary classification, BERT-based models performed better, the numerical features experiment having a macro F1 score of 0.5. However, for the multiclass classification the results were similar, the CatBoost model achieving a macro F1 score of 0.37.

Experiment	Threshold	F1	AUC
Only text features	0.05	**0.712**	0.678
Only numeric features	-	0.606	0.532
Fusion of the features	0.6 and 0.2	0.686	**0.7**
Multimodal	0.05 and 0.95	0.64	0.576

Table 1. Results for the experiment of fusion the textual features (MentalBERT) with the numerical ones (CatBoost) for the test set

Table 1 shows the results for the two models used separately and together. The table highlights that in the experiment in which both textual and numerical features are used, the results have improved, for both macro F1 score and AUC. Nevertheless, both in the experiment in which only textual features were used, and in the one in which they were combined, the thresholds used to classify the final score were small (e.g., 0.05 and 0.2), meaning that only a small part of the QA pairs from a dialogue of a depressed patient, were classified as depressed (i.e., score 1). Clearly, there were certain pairs of values for the 2 variables (i.e., the weight and the threshold) for which the values of the metrics were equal; an example of values for the best results for the validation set are 0.6 for the weight for MentalBERT and 0.2 for the threshold. Regarding the results for the approach using a Multimodal Transformer, we concluded that the combine method significantly influenced the results (e.g., the method with the lowest outcome having an AUC of 0.312 compared to 0.64), despite this, the results were weaker than in the other approach (i.e., the experiment in which the weighted average was used).

Experiment	Division	Threshold	F1	AUC
Zero-Shot	QA pairs	0.07	0.64	0.576
	Chunks	0.3	0.654	**0.751**
Zero-Shot Sentiment	QA pairs	0.6	0.64	0.576
	Chunks	0.6	**0.712**	0.678

Table 2. Results for the experiment of Zero-Shot learning using TARS for the test set

Regarding the Zero-Shot experiments, the results in which conversations were divided into QA pairs are poorer, the main reason being that most entries were classified in the same class. In the Zero-Shot Sentiment experiment, for the validation set and the division into QA pairs, the results were the best, achieving a macro F1 score of 0.803 and an AUC of 0.795. For the test set, again, dividing into chunks helped more than dividing into QA pairs, but the results from the first experiment remain superior, with an AUC of 0.751 compared to 0.678.

Experiment	Threshold	ΔF1	ΔAUC
Only text features	0.28	-0.034	+0.022
Only numeric features	-	**+0.047**	**+0.068**
Fusion of the features	0.7 and 0.5	0	0

Table 3.Comparative Analysis of Results in Previous Experiments after adding the new generated data for the test set

For both *Only text* and *Only numeric* experiments, the results improved, but not significantly, and for the F*usion of features* experiment the results remained the same. Nevertheless, a difference can be observed in the elements that form the confusion matrix (TN - True Negative, FP– False Positive, FN - False Negative, TP - True Positive). For the *Only text* experiment, the results are: TN = 20, FP = 12, FN = 2, TP = 12 for the test set. Due to the fact that all the average scores were in the interval [0.2, 0.4] the threshold was chosen in that interval with a step of 0.01. For the *Only numeric* experiment, the results are: TN = 24, FP = 8, FN = 6, TP = 8 for the test set. On the fusion task, the results are the same as in the experiment without new data: TN = 20, FP = 12, FN = 2, TP = 12 for the test set. By adding the new data, the model tendency to classify in the majority class decreased, this can be deduced both from the confusion matrices and from the fact that the chosen threshold has higher values. One explanation for the small improvements is that the generated dialogues were less complex and perhaps more obvious than those in the original dataset.

CONCLUSIONS

Depression is a complex phenomenon with a variety of symptoms, a part of them being detectable by analyzing people's speech from a linguistic point of view. This paper has presented the process of analysis and classification of depression with the help of the techniques of NLP and ML. We performed an analysis of the conversations from the point of view of POS and sentiments and emotions and detected relevant differences between the transcripts from depressed and non-depressed participants. After that, we presented a set of experiments starting from baseline approaches, to strategies of improvement as: dividing the entries into QA pairs or chunks, combining the texts with the extracted features or even Zero-Shot approaches, achieving satisfying results. Additionally, a systematic method of generating data using ChatGPT was presented.

In terms of future work, we plan to generate new data that is more relevant to the original dataset. A problem with the generated data was that depression was too evident; in future conversations, this should not be so obvious. For example, one potential approach is to prompt the model to behave like a patient with depression who is trying to hide their illness. Another idea would be for the generated dialogues to be built on the same set of questions as the original ones. Moreover, new methods, such as extracting the most relevant segment of a conversation, can be experimented with to gain as much context as possible.

ACKNOWLEDGMENTS

We would like to thank Jill Boberg from University of Southern California for giving us access to the DAIC-WOZ database.

REFERENCES

1. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., & Vollgraf, R. FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. In *Association for Computational Linguistics* (2019) 54-59.

2. Alhanai, T., Ghassemi, M., & Glass, J. Detecting Depression with Audio/Text Sequence Modeling of Interviews. In *Interspeech* (2018), 1716-1720

3. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. Enriching Word Vectors with Subword In *Association for Computational Linguistics* (2017), 135-146

4. Bucur, A.-M., & Dinu, L. P. Detecting Early Onset of Depression from Social Media Text using Learned Confidence Scores. In *Academia University Press* (2020), 73-78.

5. Chiticariu, L., Li, Y., & Reiss, F. Rule-based information extraction is dead! Long live rule-based information extraction systems!. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (2013), 827-832.

6. Cohen, J. (1988). Statistical power analysis for the behavioral sciences. In *Journal of the American Statistical Association* (1988), 680-681.

7. Correlation coefficient: simple definition, formula, easy steps. Accessed June 25, 2023. URL: Correlation Coefficient: Simple Definition, Formula, Easy Steps | Statistics How To

8. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2019) arXiv:1810.04805

9. Depression. Accessed June 25, 2023. URL: Depression | National Institute of Mental Health

10. Depression Rates by Country 2023. Accessed June 25, 2023. URL: Depression Rates by Country 2023 | World Population Review

11. Dinkel, H., Wu, M., & Yu, K. Text-based depression detection on sparse data (2020) arXiv:1904.05154

12. Dorogush, A., Ershov, V., & Yandex, A. CatBoost: gradient boosting with categorical features support (2018) arXiv:1810.11363

13. Flair. Accessed June 25, 2023. URL: Flair | PyPi

14. Gratch, J., Artstein, R., Lucas, G., Stratou, G., Scherer, S., Nazarian, A., Wood, R., Boberg, J., DeVault, D., Marsella, S., Traum, D., Rizzo, S., & Morency, L.-P. The Distress Analysis Interview Corpus of human and computer interviews. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (2014), 3123-3128.

15. Gu, K., & Budhkar, A. A Package for Learning on Tabular and Text Data with Transformers. In *Proceedings of the Third Workshop on Multimodal Artificial Intelligence* (2021), 69-73.

16. Ji, S., Zhang, T., Ansari, L., Fu, J., Tiwari, P., & Cambria, E. MentalBERT: Publicly Available Pretrained Language Models for Mental Healthcare. (2021) arXiv:2110.15621

17. KHCoder. Accessed June 25, 2023. URL: KHCoder

18. Kim, H., Jack Hessel, Jiang, L., West, P., Lu, X., Youngjae, Zhou, P., Le Bras, R., Alikhani, M., Kim, G., & Choi, Y. SODA: Million-scale Dialogue Distillation with Social Commonsense Contextualization (2023) arXiv:2212.10465

19. Kroenke, K., Strine, T. W., Spitzer, R. L., Williams, J. B. W., Berry, J. T., & Mokdad, A. H. The PHQ-8 as a measure of current depression in the general population. In *Journal of Affective Disorders* (2009). 163–173.

20. Le, Q. V., & Mikolov, T. Distributed Representations of Sentences and Documents (2014) arXiv:1405.4053

21. Mikolov, T., Chen, K., Corrado, G., & Dean, J. Efficient Estimation of Word Representations in Vector Space. (2013) arXiv:1301.3781

22. Natural Language Toolkit. Accessed June 25, 2023. URL: Natural Language Toolkit | NLTK Project

23. NRCLex4.0. Accessed June 25, 2023. URL: NRCLex4.0 | PyPi

24. PyConverse. Accessed June 25, 2023. URL: PyConverse | PyPi

25. Reddit. Accessed June 25, 2023. URL:Reddit

26. Tadesse, M. M., Lin, H., Xu, B., & Yang, L. Detection of Depression-Related Posts in Reddit Social Media Forum. In *IEEE Acces*s (2019), 44883–44893.

27. TextBlob: Simplified Text Processing. Accessed June 25, 2023. URL: TextBlob: Simplified Text Processing

28. Tølbøll, K. B. Linguistic features in depression: a meta-analysis. In *Journal of Language Works* (2019) 39–59.

29. Wolohan, J., Hiraga, M., Mukherjee, A., Sayyed, Z. A., & Millard, M. Detecting Linguistic Traces of Depression in Topic-Restricted Text: Attending to Self-Stigmatized Depression with NLP. In P*roceedings of the First International Workshop on Language Cognition and Computational Models* (2018), 11–21.