Integrating Sentiment Analysis to Enhance Mental Health Support Chatbots

Senju Murase
Department of Science and Engineering
Southampton Solent University
Southampton, United Kingdom
https://orcid.org/0009-0003-9571-7230

Jarutas Andritsch
Department of Science and Engineering
Southampton Solent University
Southampton, United Kingdom
jarutas.andritsch@solent.ac.uk

***Abstract*— Mental health conditions have adverse effects on an individual's quality of life. Between 2017 and 2019, there were 52.9 emergency department visits per 1,000 adults for mental health disorders. The rising popularity of artificial intelligence prompted its use in depression detection or mental condition diagnosis, establishing its prominent role in the mental health sector. In this research, we developed an AI chatbot with enhanced contextual intelligence using sentiment analysis to support individuals experiencing mental distress. For this research, we utilized a dataset of text data from 1.6 million posts on the social media platform X (formerly Twitter) due to its open-source availability and accessibility. The collected dataset was cleaned of repetitive characters, special characters, URLs, and numbers. NLP techniques, including tokenization, lemmatization, and stemming, were used for pre-processing. The application identifies sentiments, generates responses, and suggests basic support remedies. We used the Rasa framework to create a hybrid chatbot with customizable configurations. An LSTM model for sentiment analysis was integrated into the Rasa chatbot as a custom action component. A batch size of 32 and an optimal maximum sequence length were selected for balanced training efficiency and accuracy. The LSTM model achieved 76% accuracy in training and validation, enhancing the chatbot text comprehension. Future improvements will include adding personal features and expanding the user base.**

Keywords*— *Deep learning, Natural language processing, Sentiment analysis, Chatbot, Long Short-Term Memory

I. Introduction

Mental health conditions could occur to any individual, hindering them from daily activities in any environment, whether at school or work. Between 2017 and 2019, there were 52.9 emergency department visits per 1,000 adults for mental health disorders [1]. Individuals with severe mental health conditions are seven times more likely to be unemployed, and those with common mental health conditions are three times more likely compared to individuals with stable psychological health [2]. Additionally, a correlation was identified between depression and quality of life in Parkinson's disease patients, affirming that anxiety and depression undeniably diminish the quality of life [3]. Mental disorders decrease productivity and quality of life for many, taking a toll on modern-day society.

In today's world, where almost all youth have smartphones, technology has become a widely popular and practical vessel of communication. Technology has substantially contributed to the healthcare sector, from developing standard medical equipment in clinics to providing doctors or patients with practical resources. Furthermore, artificial intelligence has become a prominent tool in many fields in recent years. Introducing artificial intelligence has unlocked previously unimaginable approaches and techniques. Specifically, in mental health, the use of artificial intelligence for detecting depression and diagnosing other conditions has become an increasingly common research focus [4]. An illustrative example of AI applications targeting anxiety and depression is the chatbot "Tess," which functions on messaging platforms such as WhatsApp. While Tess cannot fully replace qualified professionals, it serves as a valuable alternative in areas that offer small clinical resources [5][6].

This research aimed to use sentiment analysis with a chatbot to provide AI support for individuals dealing with mental health issues. The sentiment analysis integrated within the chatbot can interpret user intentions behind their messages, contributing to the design of an intelligent chatbot that detects emotions, understands contexts, and responds appropriately, offering immediate remedies if necessary. Our focus was on support for depression and anxiety. The paper is organized as follows: Section II reviews sentiment analysis and chatbots in mental health. Section III describes our methodology, including the design and evaluation of the sentiment analysis. Section IV presents the results of the chatbot's performance, followed by a discussion. Section V concludes with a summary of key findings and recommendations for future research.

II. Literature Review

In this section, we explore the increasing use of artificial intelligence in mental health care and examine how AI technologies are being integrated to improve treatment and support.

A. Sentiment analysis in mental health sector

Over the years, sentiment analysis has gotten recognition for its crucial role in the healthcare and medical industries. Its application in evaluating patients' moods, tracking epidemics, monitoring drug reactions, and understanding diseases were highlighted. One challenge in sentiment analysis for mental health is the accumulation and gathering of relevant data. To address this, many researchers have turned to publicly available data from social media, which serves as a dynamic data source [7]. Additionally, several social media platforms, such as X (formerly known as Twitter), offer open-source data on user moods and psychological stability, making this information accessible to the public. One example was when data from social media platforms were used to investigate stress levels among college students who experienced on-campus gun violence [8]. The analyzed sentiments from compiled data from twelve universities across the United States were compared. In another instance, an NLP (Natural Language Processing) technique was implemented to identify Alzheimer's disease and dementia depiction on Twitter [9]. The results revealed that more than 20% of the tweets containing affiliated keywords perpetuate public stigma, which could lead to harmful stereotypes of disabled individuals.

The emergence of deep learning models, a subset of machine learning techniques, has significantly improved the efficiency and accuracy of sentiment analysis models. The LSTM (Long Short-Term Memory) model, a variation of the RNN (Recurrent Neural Network), is renowned for diverse applications, from NLP to forecasting and speech recognition. The LSTM model can also handle long sequences and store past information for an extended period, resolving the vanishing gradient problem of the RNN model to a small capacity [2].

In NLP, emotion recognition or sentiment analysis is a common practice that exercises text classification. Numerous factors play a part in training a deep learning model for sentiment analysis: model selection, data collection, data pre-processing, and hyperparameter settings. Each of these factors contributes to the quality of the model and its performance. While the choice of training model undoubtedly influences the outcome of sentiment analysis, it is crucial to underline the pivotal role of data quality and purity. Cleaning noise and text polarity data, such as punctuation and special characters, is fundamental in training an accurate model [10]. Other indispensable data purification techniques, including tokenization, stop word removal, lemmatization, and removing repetitive characters, are also noteworthy NLP tasks.

Additionally, hyperparameters are vital components that can affect the core setting of the training model. Their presence and scope can drastically change the training process. Examples of hyperparameters include the number of epochs, batch size, maximum sequence length, testing size, number of layers in the neural networks, and choice of activation function in the neural network layer like ReLU, Sigmoid, and Softmax. The number of epochs, for example, represents how often the model will intake the training dataset. The higher the number of epochs, the greater the model's accuracy. However, many epochs also come with the risk of generating an overfitting model. Overfitting occurs when a model is trained overly specifically for a training dataset. Therefore, prediction on text data not included in the dataset results in a wild guess. The batch size also influences whether the model would be overfitting or underfitting. Another case of adjusted hyperparameter setting is the maximum sequence length, the maximum number of word vectors the model can intake. The system will truncate text data that exceeds the maximum sequence length. When the maximum sequence length. If the maximum sequence length count is too large, too many zero factors will be filled [11]. Various evaluation metrics may be employed to assess the quality of the trained model [10]. Model accuracy is a suitable metric when the labels in the training dataset are nearly balanced.

In conclusion, multiple factors contribute to the training process and the model's performance when developing a high-end sentiment analysis model. These factors range from the type of models and datasets to employ to data pre-processing techniques and libraries. Hyperparameters like batch size or number of epochs, which are highly customizable, also directly influence the training speed, model accuracy, and model fitting. Monitoring plots and early stopping is necessary to determine the optimal value of epochs and maximum sequence length and avoid obstacles of overfitting or underfitting.

B. *Chatbot in mental health sector*

Chatbots, whether in text form, speech-to-text recognition, or any other input form, have recently become widely spread. Despite the short span in popularized usage, they have left a lasting impression and infinite potential concerning mental health.

By 2019, chatbots were already being used to prevent self-harm or even as a cognitive therapy service [12]. Some individuals felt uneasy disclosing their views to a human being. Therefore, chatbots are plausible alternatives to conventional cognitive therapy. Chatbots are also used for training and screening patients, stressing their transformative role in the mental health sector. A notable example is the LISSA chatbot, which assisted in training individuals diagnosed with autism or those struggling with interpersonal skills. This example stressed the potential of modern chatbots to revolutionize the field of mental health, with a particular focus on depression and autism [13].

III. METHODOLOGY

This research presented a chatbot system based on the Rasa framework, integrated with a sentiment analysis feature implemented in Python. The sentiment analysis module used an LSTM model. Details of the research methods are described below.

A. *Dataset Prepration*

This research employed a dataset consisting of text data from 1.6 million posts on X. We prepared the dataset for sentiment analysis to fit an uncomplicated format for the machine to encode. Before executing NLP tasks on the data, ensuring the ratio of positive and negative sentiments was paramount. This step was crucial as it directly impacts the model accuracy, which calls for a proportional training dataset. Due to the large volume of 1.6 million posts, we refined the dataset to include only text data and sentiment labels (0 for negative, 1 for positive). The program extracted fifty thousand texts for each sentiment, ensuring a balanced and reduced dataset.

As a data pre-processing step, we utilized NLP techniques such as tokenization, lemmatization, and stemming to modify the texts in the dataset. These techniques made the text encoding more manageable for the machine. Tokenization, for instance, is a process in which the machine converts sentences and words into tokens, facilitating improved comprehension of the input's context. The TensorFlow tokenizer was utilized for training the LSTM model. Furthermore, we created functions to exclude punctuation, special characters, numerical numbers, repetitive characters, and stop words from the data. After the program cleaned the data, we set the text data set as the feature variable and the labels as the target variable.

B. Sentimental Analysis Model

The primary Python library used in this research was TensorFlow, which was employed to train an LSTM model for sentiment analysis. The model's hyperparameters were configured with a batch size of 32, two epochs, a maximum sequence length of 140, and a training-to-testing-to-validation set ratio of 8:1:1.

We used TensorFlow to train an LSTM model for our sentiment analysis. The essential hyperparameters include maximum sequence length, test size, batch size, and number of epochs. Non-compulsory hyperparameters include the number of neural network layers (excluding the LSTM model, which is vital) and the type of layers. The batch size, a hyperparameter that significantly influences the model's quality and fitting, was maintained consistently throughout the training. We chose a batch size of 32, a value widely considered balanced for this task. Similarly, the number of epochs, another crucial hyperparameter, was not set in stone. We iteratively trained multiple variants of the same model, adjusting the epochs from two to five to find the optimal balance between training time and model performance. A minimum of two epochs is required to evaluate the model later by plotting the losses and accuracies. As a result of numerous tests and evaluations, the ideal number of epochs for the LSTM model was two.

The dataset's test size remained constant until the concept of a validation set was introduced. A validation set, apart from a training and testing set, is an explicit testing set used to estimate the model's validation loss or accuracy. Certain functions include a split validation parameter, which divides the initial testing set to generate a validation set according to a given value. The ratio of the training set to the testing set was altered from 8:2 to 8:1, with the validation set covering 10% of the total dataset. To detect the relevant maximum sequence length, we plotted the total token count of the text data after tokenization. The token count fell within 100 tokens. We set the maximum sequence length at 140 to ensure we cover all possible scenarios.

The LSTM model comprises seven neural network layers with separate designated roles and input and output layers. We chose ReLU (Rectified Linear Unit) as the activation function due to its superior performance over others with saturated outputs [14]. Additionally, we selected a Sigmoid activation function, as the model employs binary cross-entropy loss, which requires sigmoid outputs. Including a dropout layer to avoid overfitting in neural networks was also imperative.

C. AI Chatbot Framework

1) AI chatbot architecture

This section discusses the AI chatbot architecture, including the integration of sentiment analysis. Fig. 1 illustrates the architecture of Rasa, a framework implemented for our hybrid chatbot. Rasa is an open source chatbot framework known for its high precision and smooth conversation flow; it is also highly customizable and often used for hybrid chatbots.

The Rasa framework, a versatile system, is built upon two major modules: Rasa NLU and Rasa Core. Rasa NLU, as the name implies, is a Natural language understanding engine that excels at comprehending user messages. On the other hand, Rasa Core, a module of immense flexibility, is responsible for managing the conversational system, giving programmers the power to control the chatbot's following action.

While training data such as intents and entities make up Rasa NLU's training data, actions and stories are the central components of Rasa Core's training data. Intents represent the meanings behind user input messages. When defining the intents, it is crucial to remember the rules in context analysis, such as using N-gram analysis or noun phrase extraction. It is also important to note that the phrases registered under intents should concisely combine keywords. Otherwise, the chatbot cannot distinguish between the different intents. We designed a rule of thumb that two nouns, negative contradiction and adjective together, or negative contradiction followed by adverb and adjective, would make up one intent example. Meanwhile, adverbs and adjectives should be separated and defined as individual entities. The training data entities provide additional information regarding specific words.

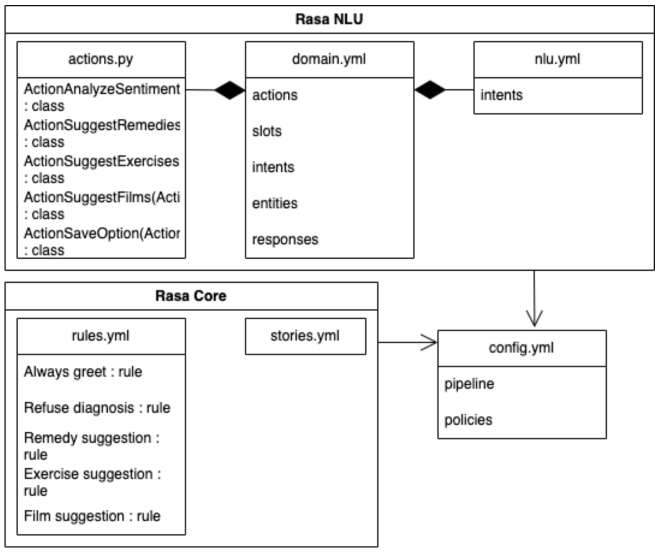

Fig. 1. Rasa framework architecture

The actions training data in Rasa Core guides the chatbot on which actions to perform next. An action may either return a simple message or provide values from Python-coded methods. Equally important is the rules training data, which functions as pre-defined rules the chatbot must adhere to. Lastly, the story training data represents the overall conversation flow, integrating all other training data. Each story captures the dialogue between the user's intent and the chatbot's response.

The Rasa assistant plays different roles depending on the policies and combination of policies in the Rasa framework. Under the rule policies, for example, it functions as a rule-based chatbot, making decisions based on the defined rules in the rules training data. In contrast, under machine learning policies, like the TED (Transformer Embedding Dialogue) policy, it operates as a self-learning chatbot based on the specifications in the stories training data. Our chatbot consists of both policies to train the model to be a hybrid chatbot. This configuration of the Rasa, along with the adaptable policies and training data, makes the Rasa framework a powerful tool for hybrid chatbot development.

2) *Chatbot methods*

We constructed rules that the chatbot would maintain, whether the rule encourages the chatbot to respond in a specific manner or prevents it from taking a particular action. Other steps included listing all possible conversation patterns to simulate the question and response interaction of the user and chatbot and selecting the appropriate developing tools. As we were gathering the software requirements for the chatbot function, we had two essential requirements in mind:

- The application serves solely as a companion to the user and does not, under any circumstances, attempt to diagnose the user's mental state. All permissible and prohibited chatbot responses are governed by pre-established rules.
- If the chatbot detects a negative sentiment, it suggests a remedy for the user. However, scientific studies and evidence must substantiate these suggestions or remedies.

We must assemble a list of terms representing mental disorder names to solidify the rule for restricting chatbot-to-user diagnosis. After that, we created a catalog of all possible queries the user input that would hint at users wanting the chatbot to perform diagnosis. This catalog, combined with the dictionary of disorder names, would act as a regulator to intercept the chatbot from executing unqualified diagnoses. When programming the method for the remedy suggestion feature, we had to gather resources that would act as a tool to convey these remedies, whether that would be links to hospitable videos online or instructions for the users to perform certain chores, relieving their emotional negativity. To target the users' anxiety and depression, we designed numerous support patterns that the chatbot would output randomly. As an alternative to humor therapy, the user could choose between acquiring a link to a short amusing video and receiving recommendations on comedy films and their streaming options. To promote physical activity, the chatbot shares knowledge on quick workouts the user can perform on the spot, along with a hard-coded link to a guidance video. The chatbot offers both aerobic and anaerobic exercises.

Regarding a well-balanced diet to reduce anxiety levels, the program could provide input on uncomplicated vegetable-packed recipes the users could cook at home or advise them to visit a Mediterranean restaurant if they are dining outside. Furthermore, the chatbot could integrate these remedies into everyday conversations, such as suggesting the user consume non-refined grains, vegetables, and fruits on their next trip to the grocery market. On a smaller scale, the users could be advised to drink polyphenol-rich beverages like hot chocolate, tea, or coffee. The chatbot would attempt to discourage users from taking artificial sweeteners or additives and instead indulge in natural sweeteners like sugar or honey. The program should also prevent users from drinking diet drinks and sodas.

Finally, the treatment recommendation feature delivers insights on deep breathing exercises like pursed-lip breathing and shares a link to an online video explaining the techniques in more detail. To put it succinctly, one could set the feature to deter the chatbot from fulfilling chatbot-to-user diagnosis by utilizing a dictionary of words indicating disorder names. Regarding the remedy suggestion feature, the chatbot would unsystematically choose a treatment from the pre-arranged set, varying from recommendations on comedy films to deep breathing techniques.

3) *Chatbot configuration*

In our application, we defined intents that include greetings, farewells, affirmations, denials, and mood expressions. Vocabularies and expressions that depict mental disorders were registered under entities. Moreover, a "sentiment" entity was added to store sentiments analyzed by custom actions based on user messages, enabling the chatbot to respond appropriately.

To integrate the trained sentiment analysis model into the Rasa application, we programmed a Python-coded method under a custom action file. Firstly, the trained model was loaded using the TensorFlow library. Tokenized user input messages were converted into sequences and input into the model for prediction. Messages with predicted values of 0.51 or higher were categorized as positive sentiments, while those below 0.51 were categorized as negative. The method then returned the analyzed sentiment, allowing the Rasa chatbot to respond accordingly. The remedy suggestion action extends the sentiment analysis function. It is triggered by registering a negative value under the sentiment entity. Lastly, the function to prevent the chatbot from performing virtual diagnoses was assigned within the rules training data.

In Rasa, convolution occurs when a user query deviates from or only partially follows the story path. Given the infinite potential for errors, an extended testing phase with numerous test scenarios is essential to achieve a well-rounded, faultless chatbot system.

D. *Sentimental Analysis Model evaluation metrics*

We selected training and validation accuracy as evaluation metrics and created confusion matrices to assess the accuracy of positive and negative sentiment predictions. We also considered model fitting. Despite sufficient accuracies, if the model is overfitted or underfitted, it is not suitable in practice.

E. *Chatbot evaluation metrics*

Table I presents the possible test cases to evaluate the chatbot application. The table holds four columns: Input message, predicted response, and actual response.

The input message is the text the user inputs into the command line. The test case table enables the assessment of one-time and multiple query interactions. The test cases aim to assess whether the given training data is operating appropriately or whether proposed rules are being followed.

It is crucial to assess the difference between the predicted response—the anticipated reaction from the chatbot—and the actual response, which is the message returned by the chatbot. Foreseeing the exact message produced is difficult if the training data includes multiple response patterns. Another purpose of the test case is to ensure the chatbot adheres to the rules and story training data. Generally, generating short stories is preferable to long, intricate ones. Dividing multiple user-to-chatbot dialogues into smaller stories reduces confusion, elevating its efficacy. However, this technique could potentially lead to an issue related to scalability.

One key challenge in testing the chatbot's performance is handling unfamiliar queries. The only current precaution for unfamiliar queries is continually adding unexplored queries to the designated intents. However, it is impossible to anticipate all user input messages. A well-organized method must be designed to effectively manage the traffic of more complex chatbots.

TABLE I. TEST CASES

Input message	**Predicted response**	**Actual response**
Feeling sad	utter_res_to_sa (neg)	utter_res_to_sa (neg)
I'm happy	utter_res_to_sa (pos)	utter_res_to_sa (neg)
I'm very very sad	utter_res_to_sa (neg)	utter_res_to_sa (pos)
3	utter_res_to_option (3) - remedies suggestion	utter_res_to_option (3)
4	utter_res_to_option (unrec)	utter_res_to_option (unrec)
Hello -> Feeling sad -> 2	utter_greet -> utter_res_to_sa (neg) -> utter_res_to_option (2) - provide hotline contact	utter_greet -> utter_res_to_sa (neg) -> utter_res_to_option (2) - provide hotline contact
Could you check if I'm depressed	utter_refuse_diagnosis	utter_refuse_diagnosis
Film	action_suggest_films	action_suggest_films

IV. RESULTS AND DISCUSSION

A. *Sentimental Analysis model evaluation*

Our program provided a clear output on the model's training loss, training accuracy, validation loss, validation accuracy, and training time upon completion. Generally, the favorable validation accuracy is between 70% and 90%. It also generates visual representations of the losses and accuracies for easy comparison, with the X-axis denoting the number of epochs and the Y-axis indicating the value of the loss/accuracy.

Fig. 2 demonstrates the behavior of losses and accuracies during model training. When the validation loss starts to increase, the model stops taking new data. A widening gap between training and validation loss after this point indicates overfitting. Similarly, if training accuracy surpasses validation accuracy, the model is not improving on unseen data and is overfitted. An underfitted model is indicated when the loss or accuracy graphs do not intersect or converge. The LSTM model training was completed with a training accuracy of 77.5% and a validation accuracy of 76.9%. The loss and accuracy plots demonstrated a balanced fit, making it suitable for sentiment analysis. According to the confusion matrix (Fig. 3), 75% of negative sentiments and 78% of positive sentiments were accurately predicted.

B. *Chatbot*

Fundamentally, no programming error was found in our application; the chatbot and custom actions, including sentiment analysis and remedy suggestion functions, operated without faults. Following phrases with negative sentiment, the application offered a series of aids to the user. When the user entered "3" to request a remedy, the application randomly shared an immediate remedy and provided links to a video if available. However, despite the model demonstrating satisfactory evaluation metrics, its practical performance in sentiment analysis raised concerns.

The predicted and actual responses of the test cases usually matched, except when sentiment analysis returned a false sentiment. Initially, phrases like "feeling sad" were correctly analyzed as negative. Nevertheless, phrases indicating a positive sentiment such as "I'm happy" were considered to have an opposite sentiment. Upon evaluation, the method showed varying results based on the number of words rather than the context.

V. CONCLUSION

Ultimately, we accomplished developing a chatbot application that could benefit individuals combating their mental health. We were mandated to be adaptable with modifications, as unforeseen outcomes were inevitable. Although custom actions allowed room for chatbot exploration, we discovered that fundamental steps, such as depicting the training data, had a more significant impact on the overall enactment of the chatbot. Furthermore, the trained sentiment analysis model was successfully integrated with the application. However, the vital performance of the model fell behind. As a result, further efforts are needed to enhance the precision of the analysis.

For further work and recommendations, enhancing the accuracy of the sentiment analysis model by experimenting with various datasets and settings remains an ongoing challenge. Potential alternatives include the RoBERTa and DistilBERT models, both reimplementations of BERT with greater capacity and precision [15]. Moreover, the quality of the chatbot system could be improved, offering more versatile responses and conversation patterns. To achieve this, exploring the enterprise version of the Rasa framework, Rasa X, is advised. Furthermore, expanding the chatbot's target users should be considered. For instance, individuals with autism could be defined as target users by further refining the user intent data in Rasa. A more customizable chatbot system tailored to users should be developed to overcome one of the most common challenges in chatbot development: lack of personalization. Additionally, new features should be developed, such as a speech-to-text transcription function or a function to search for relevant nearby facilities on a digital map.

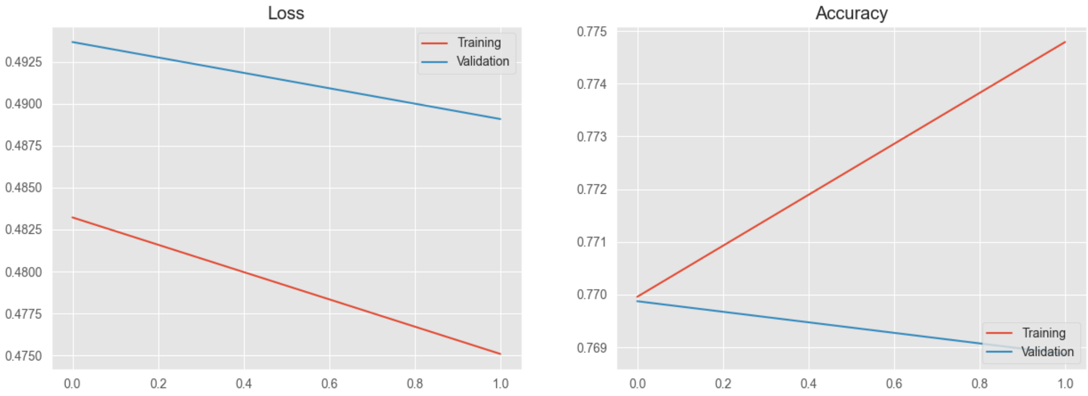

Fig. 2. Losses and accuracies

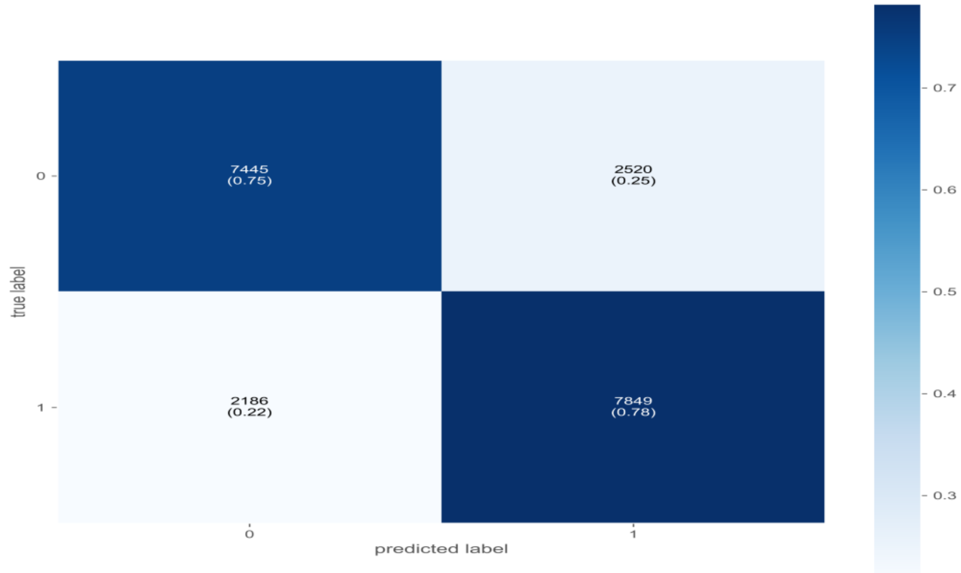

Fig. 3. Confusion matrix

REFERENCES

[1] L. Santo, Z. J. Peters, and C. J. DeFrances, Emergency department visits among adults with mental health disorders: United States, 2017-2019. US Department of Health and Human Services, Centers for Disease Control and …, 2021.

[2] E. P. Brouwers, "Social stigma is an underestimated contributing factor to unemployment in people with mental illness or mental health issues: position paper and future directions," vol. 8, pp. 1–7, 2020.

[3] W. Su et al., "Correlation between depression and quality of life in patients with Parkinson's disease," vol. 202, p. 106523, 2021.

[4] M. L. Joshi and N. Kanoongo, "Depression detection using emotional artificial intelligence and machine learning: A closer review," vol. 58, pp. 217–226, 2022.

[5] S. Zhou, J. Zhao, and L. Zhang, "Application of artificial intelligence on psychological interventions and diagnosis: an overview," vol. 13, p. 811665, 2022.

[6] A. Fiske, P. Henningsen, and A. Buyx, "Your robot therapist will see you now: ethical implications of embodied artificial intelligence in psychiatry, psychology, and psychotherapy," vol. 21, no. 5, p. e13216, 2019.

[7] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," vol. 55, no. 7, pp. 5731–5780, 2022.

[8] K. Saha and M. De Choudhury, "Modeling stress with social media around incidents of gun violence on college campuses," vol. 1, no. CSCW, pp. 1–27, 2017.

[9] N. Oscar, P. A. Fox, R. Croucher, R. Wernick, J. Keune, and K. Hooker, "Machine learning, sentiment analysis, and tweets: An examination of Alzheimer's disease stigma on Twitter," vol. 72, no. 5, pp. 742–751, 2017.

[10] M. Birjali, M. Kasri, and A. Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends," vol. 226, p. 107134, 2021.

[11] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, "Sentiment analysis of comment texts based on BiLSTM," vol. 7, pp. 51522–51532, 2019.

[12] A. N. Vaidyam, H. Wisniewski, J. D. Halamka, M. S. Kashavan, and J. B. Torous, "Chatbots and conversational agents in mental health: a review of the psychiatric landscape," vol. 64, no. 7, pp. 456–464, 2019.

[13] A. A. Abd-Alrazaq, M. Alajlani, A. A. Alalwan, B. M. Bewick, P. Gardner, and M. Househ, "An overview of the features of chatbots in mental health: A scoping review," vol. 132, p. 103978, 2019.

[14] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," vol. 503, pp. 92–108, 2022.

[15] D. Cortiz, "Exploring transformers models for emotion recognition: A comparision of BERT, DistilBERT, RoBERTa, XLNET and ELECTRA," 2022, pp. 230–234.