A Mental Health Model for Early Intervention

The solution includes a web-app tool designed to provide early intervention for individuals experiencing mental health challenges. The platform uses natural language processing and machine learning algorithms to analyze text input from users and provide personalized support and resources. In this report, we present an overview of the platform's features, the technology used to build it, and our findings from testing its effectiveness. We also discuss the potential impact of the model on the field of mental health care and the broader community.

Here are six pieces of information I think you'd love to know:

1. Objective

The objective of this project is to develop an AI binary classification model using LSTM for early intervention for individuals experiencing mental health challenges. The model will be integrated into a Streamlit web app interface, which will allow users to input text and receive a classification result indicating whether they may be at risk for suicide. If the user is classified as suicidal, they will be offered to be connected with a clinical therapist to help solve their problems.

2. Methodology ✨

To achieve the project objectives, we first collected a dataset consisting of two columns: text and class, where the text column contains the user's input text and the class column contain the corresponding label, indicating whether the text is suicidal or not. The dataset was preprocessed using techniques such as tokenization and word embedding to convert the text data into a format suitable for LSTM model training. We used a Streamlit web app interface to integrate the model into a user-friendly application. The LSTM model was trained on the preprocessed data using the Keras library, and the model's performance was evaluated using metrics such as:

- Accuracy
- Precision & Recall
- F1-score

A summary of model architecture:

An embedding layer with an input dimension of 5000, an output dimension of 16, and an input length of 100.
An LSTM layer with 64 units and a dropout rate of 0.2.
A single dense layer with one neuron and a sigmoid activation function
Trained using binary cross-entropy loss and the Adam optimization algorithm for 10 epochs.

3. Data

The dataset used for this project consists of one dataframe with two columns: text and class. The text column contains user input text, and the class column contains a label indicating whether the text is suicidal or not.

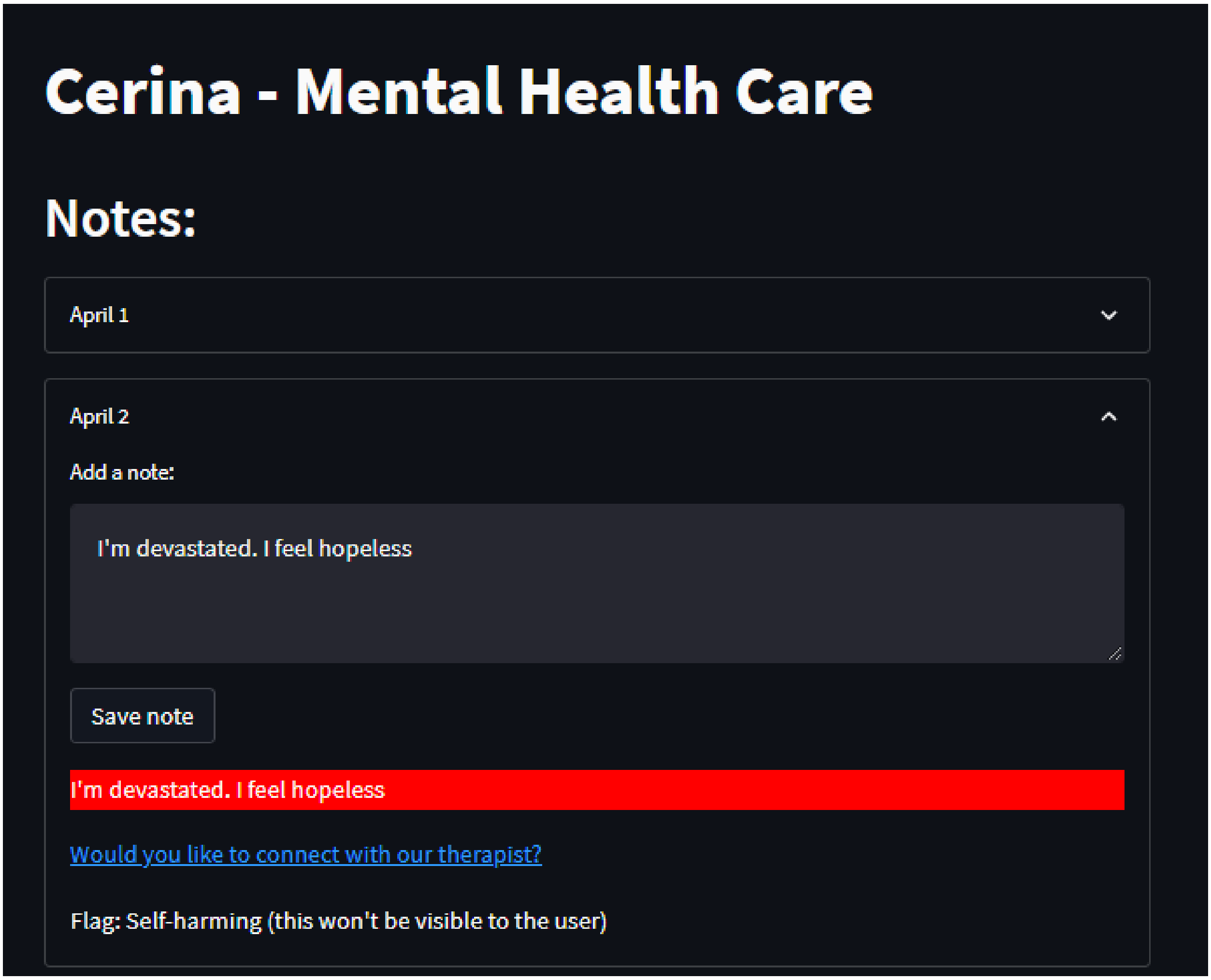
Text	Class
I don't feel so well, I need help	Suicidal
I had a good day today!	Non-suicidal
Please do something, I hopeless	Suicidal
What a lovely evening	Non-suicidal

4. Results!

The developed LSTM model achieved an accuracy of 96% on the test dataset. The precision, recall, and F1 score were also evaluated, with a precision of 96%, recall of 96%, and F1 score of 96%. The confusion matrix was also generated to visualize the model's performance *(macro averages).*

	Precision	Recall	F1-score
Non-suicidal	0.98	0.95	0.96
Suicidal	0.95	0.98	0.96

Web App Performance Screenshots:



<u>Note:</u> **The web app offers the user to be connected to a therapist if they are flagged as 'self-harmful' (*link redirects to a website*).**

5. Discussion!

The results of this project show that an LSTM model can effectively classify text data related to mental health challenges, specifically identifying suicidal text. The project's implications include the potential to identify individuals at risk for suicide early on and connect them with clinical therapists for intervention. However, the study's limitations include the need for further evaluation of a larger and more diverse dataset to improve the model's generalizability.

A technical flowchart outlining the steps involved in your process:

1. Load the dataset from a CSV file and preprocess it.
2. Split the dataset into training and testing sets.
3. Train a machine learning model using the training data and save it to a file.
4. Load the trained model and a tokenizer from their respective files.
5. Define a Streamlit application that allows users to add and view notes.
6. For each note added by the user, predict its sentiment using the loaded model and tokenizer.
7. Display the note with a background color that corresponds to its sentiment score.
8. If a note has a high enough sentiment score to indicate self-harm, display a warning message and offer a link to connect with a therapist.

6. Conclusion!

In conclusion, both the approaches of model building and training worked finely for the problem at hand. Giving a macro average accuracy of **96%.** However, we can certainly improve upon the diversity and usability of the model through prompt engineering, transfer learning, and fine-tuning further.

Thank you!

I had fun working on this task. It was a fulfilling objective to help people with their mental struggles while scaling the solution using AI. Thank you for your time!

Regards,
Ubaid Khan
9039502424
ubaidkhanindore@gmail.com