

A
PROJECT SCHOOL REPORT
ON
MENTAL HEALTH PREDICTION SYSTEM

A. SAI VARDHAN	245322748002
A. VEENA SREE	245322748004
B. KARTHIK	245322748008
G. MANIKANTA	245322748024
K. MYTHILI	245322748033
K. TEJASHWINI	245322748035

Under the guidance

of

Dr. Shilpa Choudhary

Assistant professor, CSE(AIML)



NEIL GOGTE INSTITUTE OF TECHNOLOGY

Kachavanisingaram Village, Hyderabad, Telangana 500058.

AUGUST, 2024

NEIL GOGTE INSTITUTE OF TECHNOLOGY

A Unit of Keshav Memorial Technical Education (KMTES)

Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

CERTIFICATE

*This is to certify that the project work entitled “MENTAL HEALTH PREDICTION SYSTEM” is a bonafide work carried out by “Sai Vardhan, Veena Sree, Karthik, Manikanta, Mythili, Tejashwini” of II-year IV semester **Bachelor of Engineering in CSE (AIML)** during the academic year **2023-2024** and is a record of bonafide work carried out by them.*

Project Mentor

Dr. Shilpa Choudhary

Assistant professor, CSE(AIML)

ABSTRACT

One of the greatest challenges to individuals and health systems are major depression and other mental health disorders. Depression severity prediction can be important for treatment planning and intervention. Traditional methods of assessment of depression usually require manual assessment of clinical interviews or rating questionnaires, which is time-consuming and may have variability in interpretation. In this regard, this paper proposed the use of advanced NLP techniques and deep learning models in predicting depression severity levels from textual data.

In this paper, we use the fine-tuned BERT model on the Depression Severity Levels dataset to classify text data into four classes of severity: minimum, mild, moderate, and severe. The strategy incorporates the Groq API to boost performance and model scalability. We show that, compared with conventional machine learning approaches, the proposed BERT-based model classifies very accurately the severity of depression.

It will, therefore, be a very reliable and efficient tool in ensuring that the classification of severity is done effectively in depressed patients, as required by mental health professionals to properly assess and diagnose mental illness in their patients, hence offering timely and proper interventions. The results have implications for general practices in handling mental health disorders and the resulting improved patient outcomes.

CONTENT

S. NO.	TITLE	PAGE NO.
	ABSTRACT	ii
	TABLE OF CONTENTS	iii
	LIST OF FIGURES	v
	LIST OF TABLES	vi
1	Introduction	
	1.1 Machine Learning Algorithms	2
	1.2 Deep Learning Algorithms in Predicting Mental Health	2
		4
	1.3 Objectives	4
	1.4 Contributions	4
	1.5 Challenges	
2	Literature Survey	
	2.1 Introduction	6
	2.2 Discussion	13
3	Proposed Work, Architecture, Technology Stack & Implementation Details	14
	3.1 DATA COLLECTION	16
	3.2 DATA PRE-PROCESSING	18
	3.3 MODEL SELECTION	21
	3.4 FINE-TUNING	27
	3.5 PROCESS OF INTEGRATION AND INTERFACE	32

4	Results & Discussions		32
	4.1	CONFIGURATION	33
	4.2	EVALUATION PARAMETERS	33
	4.3	ACCURACY	33
	4.4	QUALITATIVE ANALYSIS	36
	4.5	QUANTITATIVE ANALYSIS	42
	4.6	SUMMARY OF RESULTS (QUALITATIVE ANALYSIS & QUANTITATIVE ANALYSIS)	46
5	Conclusion & Future Scope		48
6	References		49

LIST OF FIGURES

Fig. No.	Figure Name	Page No.
3.1	Architecture diagram	15
3.2	Imbalanced Distribution of Labels in the dataset	17
3.3	Balanced Distribution of Labels in the dataset	17
3.4	Before Tokenization and After Tokenization	18
3.5	Length of sentence before and after adding the CLS and SEP tokens	19
3.6	Length of the sentence after padding	19
3.7	Before and after truncation	20
3.8	Attention Mask	20
3.9	Training Loss using different Hyperparameters	28
3.10	Training Loss, Validation Loss, Accuracy using different Hyperparameters	29
3.11	Training Loss, Validation Loss, Accuracy using different Hyperparameters	30
3.12	Training Loss, Validation Loss, Accuracy using different Hyperparameters	31
4.1	Evaluation Metrics of BERT model	34
4.2	Confusion Matrix	35
4.3	Prediction of LLAMA(ex-1)	37
4.4	Prediction of GEMMA(ex-1)	38
4.5	Prediction of MISTRAL(ex-1)	38
4.6	Prediction of BERT model(ex-1)	39
4.7	Prediction of LLAMA(ex-2)	40

LIST OF TABLES

Table No.	Table Name	Page No.
3.1	Fine-Tuning of Parameters	25
3.2	Hyperparameters Used in the Experiments	27
3.3	Adjusted Hyperparameters for Improved Accuracy	28
3.4	Hyperparameters for Extended Epoch Training	29
3.5	Model Training Hyperparameters and Settings	30
3.6	Parameter Specifications and Optimal Values	31

CHAPTER-1

INTRODUCTION

In this digital age, where much of today's interaction occurs, mental health professionals are called upon to engage in the difficult battle of timely and relevant help to those in need. Through the traditional methods of assessment of mental health, all too often, the help that many people need is easily missed. The personal and social burdens of poor mental health are very high. To date, it is stated that over 20% of adult in the U.S. have suffered from at least one case of a mental disorder, and 5.6% of adults are already suffering from severe psychotic disorders that bring forward conspicuous disablements of everyday functioning. It is depression and anxiety combined that cause almost \$1 trillion in lost productivity to the global economy yearly. The World Health Organization states that the quantum of the mental health workforce in India is grossly inadequate.

For the vast number of people who have ailments related to the mind and mental disorders, however, there is a shortage of both psychiatrists and psychologists. There are only three psychiatrists and psychologists per 1,00,000 people, as data from the international body shows. Owing to the pandemic and its influential factors, it predicted that the number of patients having any mental disorder would touch 20 percent in the country. About 56 million Indians live with depression, while 38 million Indians live with some type of anxiety disorder.

The detection of mental health conditions from text data has been one of the major research efforts in Natural Language Processing (NLP) and Computational Social Science (CSS) in the last few years. Most of these efforts are oriented toward the creation of domain-specific machine learning models that are tailored for specific tasks, for example, stress detection, depression prediction, or suicide risk assessment. More often than not, they become very rigid and tightly follow some predefined set of tasks only.

Another interest has been in chatbots for mental health services. Although most of the systems deployed so far are rule-based, more advanced models of language understanding can hugely benefit them to make them more effective. The same light is that the methods used sometimes introduce bias, and at times even offer harmful advice to the user.

1.1 Machine Learning Algorithms in Predicting Mental Health

SVM has been used in the classical classification of text data from mental health conditions and regression to distinguish levels of anxiety separately from depressive depression based on linguistic features. Random Forest has been used in the prediction of mental health states by aggregating the results of multiple decision trees, which makes it have strong predictions from different features that are generated from text data. The Naive Bayes approach has been implemented in the area of text classification, and it was also tried out in the prediction of mental health from textual data. The algorithm has shown good workability with large datasets. It has been applied in mental health prediction, where it is compared with the labelled examples of new data text to be classified with instances concerning the most common level of the nearest neighbours, whereas such an algorithm is not efficient for a more practical and easy mental health prediction approach.

1.2 Deep Learning Algorithms in Predicting Mental Health

Of course, there are a lot of applications for such technologies, like auto-encoders in the detection of anomalies in mental health data, especially those patterns very different from normal behaviour and hence potentially pointing towards forerunners of mental health issues. It can generate text data artificially through generative adversarial networks, on which models in mental health prediction are retrained to increase small datasets in robustness. Attention mechanisms have applied to several deep learning models in embracing performance improvement towards different mental health prediction challenges by understanding differential importance among words or phrases in a text. Graph Neural Networks have been applied in several studies in social networks and interaction analysis for mental health condition prediction and, consequently, social context modelling.

Other challenges to the integration of machine learning and deep learning models into AI for mental health include issues of data privacy and ethics, the intrinsic complexity of mental health, data quality and availability, overgeneralization or overfitting, and a host of ethical and legal considerations. Natural language would be an important ingredient, especially in the diagnosis and treatment of mental illnesses. Therefore, LLM will soon become a very powerful tool to realize what mental state the user is in based on what has been written.

There are very few recent studies available that have evaluated LLMs for mental health-related tasks, most of them in zero-shot settings with simple prompt engineering. This preliminary work showed that large LMs had initial capabilities for predicting mental health disorders by

natural language with very promising but still rather limited performance compared to state-of-the-art domain-specific Natural Language Processing models. This gap is expected since the general-purpose LLMs are not particularly trained in any mental health tasks.

We have conducted a number of experiments using another creation from Google Brain in 2018, BERT. BERT shook the world of Machine Learning with the Transformer architecture, where only self-attention mechanisms were used to understand the context relationship of words in a sentence. While very few traditional models follow sequential processing, BERT processes full sequences of words all at once; hence, it understands the full context for every word.

In this paper, we used the GROQ API framework to improve performance and functionality for a variety of large language models that were specifically used for mental health prediction. The GROQ API framework puts greater emphasis on interaction, processing efficiency, and contextual understanding within the LLM—very important in generating an accurate identification and diagnosis of mental health conditions from text data. Specifically, we have used models Llama3-70b-8192, Gemma-7b-it, and Mixtral-8X7b-32768.

Then there was the special strong point in parsing extended dialogues and posts from social networks: Llama3-70b-8192 would work quickly to pick up trends that can be representative of symptoms of various mental conditions, such as depression or anxiety. Gemma-7b utilizes tasking that requires the fine-grained interpretation of small units of text for a precise interpretation of a single post or comment, using these to diagnose the severity of conditions such as depression and anxiety, being indicated through linguistic cues and user interactions. Mixtral-8X7b-32768 is a model specifically good at generating data from several text inputs and, therefore, offers a holistic and detailed assessment of an individual's state of mental health. Mixtral-8X7b-32768 offers great insight into mental health states by bringing insights from diverse textual data. In this respect, AI and mental health present certain prospects involving benefits at the assessment scale and treatment processes for mental health.

It is going to be at the top of the application for cutting-edge APIs if applied to advanced NLP models like Bidirectional Encoder Representations from Transformers (BERT). The planting of more flexible, exact, and sensitive tools for professionals in mental health and general users is a great deal. However, this must definitely be the direction going forward: care in privacy, ethics, and the complexity of mental health so that these technologies turn out safely and effectively.

We open-source our model at MENTAL_HEALTH-BERT_LLM on hugging face. We encourage the community to experiment, innovate, and share their findings to collectively enhance the capabilities and understanding of mental health prediction technologies.

1.3 Objectives

Project targets focused on the prediction of mental health using LLMs aim at significantly developing the domain.

- We intend to build robust predictive models that will have the capacity to accurately identify and classify text data to infer mental health conditions like depression and anxiety by using LLMs. This would be accomplished by fully evaluating techniques such as fine-tuning the model to improve the performance of the model.
- Fine-tuning models like BERT using different mental health data sources improves the prediction accuracy and reliability across various text sources.
- Integrating the GROQ API framework, we can further enhance the interaction and processing capabilities of LLMs toward their effectiveness in tasks involving mental health prediction.

1.4 Contributions

This project contributes by way of a new application of Large Language Models in mental health prediction. This would thus seek to develop a strong model in predicting the severity of depression levels from online text data and hence facilitate the early detection and intervention of mental health conditions. This tool can aid health professionals in several ways: providing accurate, on-time assessments for the enhancement of diagnostic capabilities while continuing to empower resource allocation. In the final analysis, the purpose of this project is to help patient outcomes by enhancing a large mental health care system through the integration of state-of-the-art natural language processing.

1.5 Challenges

There are several challenges to the use of LLMs in mental health prediction. First in this row come the challenges of data privacy and security: sensitive mental health data needs to be processed under very strict regulations. High-quality and labelled data is hard to access, with data in this domain usually appearing relatively unstructured and scarce. Since LLMs are "black box" models, interpretation is very difficult and requires explainability to engender trust by both healthcare practitioners and patients. On ethical grounds, there are concerns if there is bias in the training datasets, which needs to be mitigated so that the predictions are fair. Other

major challenges include generalizing models across divergent populations and real-time processing system efficiency. It is further complicated by the addition of complexity through sources of data from integrating multi-modal sources. Coming up with an adaptive learning system by which learning will improve continuously without a degradation in performance is a quite demanding task.

CHAPTER-2

LITERATURE SURVEY

2.1 Introduction

This literature survey is an effort to look into the available research and methodologies in the area of Machine Learning (ML), deep learning, and Large Language Models (LLMs) relating to the prediction and analysis of mental health. Various projects and researches will be studied with the view to finding appropriate techniques and technologies to improve the accuracy and reliability of our system that predicts the level of depression using textual input. An overview of all ML algorithms, deep learning frameworks, LLMs, including their applications, strengths, and limitations, shows a comprehensive overview of the literature in this area, thus justifying our approach for this project.

R.M. et al.[\[1\]](#) offers a way of calculating the filtration coefficient using machine learning methods for ore-bearing rocks in uranium deposits. This study underlines the conventional method developed over 50 years ago that yields an R^2 of 0.32 and is often inaccurate. The authors have suggested a new approach using multiple machine learning algorithms like Gradient Boosting and Neural Networks that create models with accuracy increases of 20%–75%. The best result was obtained by LightGBM regressor: $R^2 = 0.710$. The application of the new method to the Inkai deposit data is shown to outperform traditional methods in better planning and optimization of uranium mining using in situ leaching techniques. This research has been funded by the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan and partially supported by the Slovak Research and Development Agency.

Jue Li and Chang Wu.[\[2\]](#) addresses the analysis of construction accident reports. A new method that combines convolutional neural networks with term frequency-inverse document frequency for automatic classification and information extraction from accident texts is proposed. On classification tasks, the proposed model reached an accuracy of about 84%, far surpassing conventional ways. Further, TF-IDF analysis provided significant accident areas and risky operations. The results demonstrate how deep learning and text mining have advanced the domain of safety management with respect to accident narrative analysis.

D.K. Sharma et al. [3], reviews the increasing problems of fake images on social media. In this paper, different methods for detection, digital forensics, and machine/deep learning techniques have been reviewed. This paper has therefore pointed out the strengths and limitations of those approaches: although traditional forensic techniques can ensure high accuracy, they really struggle with multi-manipulated images. Deep learning methods, mostly CNNs and Vision Transformers, provide much better performance in the identification of fake images, including those generated by GANs. This paper points out the requirement of multimodal approaches including text and visual data and urges the construction of larger, real-time labeled datasets toward better detection accuracy. This work was supported by the Brain Pool program of the Korean government and the Basic Science Research Program.

Ebrahim Khalilzadeh, Saman Jahandideh, and Vahid Noroozi propose FutureCite [4], a framework where machine learning is combined with both text and graph mining for predicting the future citation levels of research articles. Harnessing the power of state-of-the-art natural language processing in analyzing textual content and graph mining algorithms in gauging the structure and dynamics of citation networks, FutureCite uses textual content, citation networks, and metadata for feature extraction. It incorporates very diversified feature resources to help greatly improve accuracy in predictions. Extensive experiments conducted on a comprehensive test publication dataset have proven that the model is quite robust and efficient, yielding very promising results in most evaluation metrics. This high-precision prediction of future citation levels makes a great difference in enhancing researchers' capability to identify impactful literature, hence strategizing better decisions on research funding and publication planning.

Taufik Hidayat et al. [5], propose a hybrid model of machine learning that combines Markov Decision Process, genetic algorithm, and random forest to enhance the live virtual migration process in data centers. Their results indicate that the proposed model supports 99% accuracy, which is considerably more rapid in training time compared to the previous techniques such as K-nearest neighbor, decision tree classification, and neural networks. Key findings are that this model could help in the optimization of VM placement and reducing downtime, hence warranting further research in deep learning approaches applicable in enhancing data center performance.

P.M., C.A.-A, et al. [6], introduces the POLIDriving dataset, an innovative driving dataset dedicated to road traffic safety applications. The dataset was collected from various heterogeneous sources of information: different probes both inside the vehicle and on pedestrians, vehicle scanners, health monitors, weather services, sources of traffic accident

data, road geometrical, among others, with the help of a flexible ensemble of software and hardware tools. The study involves drivers of different ages and gender driving on two routes under different weather conditions. This dataset includes almost 18 h of driving data, over 61,000 observations, and 32 different kinds of data attributes. From this unique set of attributes related to driver–vehicle–road–environmental data, obtained from the POLIDriving dataset, models created with the Gradient Boosting Machine and Multilayer Perceptron achieved an accuracy of 95.6% and 98.6%, respectively. The authors conclude in that the existence of a rich, diverse dataset in POLIDriving significantly promotes research in the prediction of traffic accidents, although they point out some limitations in demographic representation and driving conditions for calling future dataset improvements to action.

V.K., B.I., et al.[\[7\]](#), on the other hand, posits that in high-stakes environments, such as education, structured learning is needed by big language models. They, thus proposed a workshop method for students with varied skills as a guide on how to apply LLMs in various platforms in writing fictional stories, making images, and basic web code. The practical experiment using 150 students to evaluate and host online their projects. An anonymous survey about the effectiveness of the workshop showed that all reviews were positive, citing the success of the workshop in demonstrating LLM capabilities and limitations. Such structured, project-based approaches are concluded in the study as a crucial part of effectively integrating LLMs into education and their creative, practical use.

Janowski, Artur, et al, [\[8\]](#), the article presents the HELIOS concept: Homogeneity Estate Linguistic Intelligence Omniscient Support; it is a new, original solution merging linguistic intelligence with machine learning to be able to better analyze the real estate market. According to them, the conventional approaches used in valuing property have, so far, lacked the precision necessary in defining the homogeneity and similarity among the properties. Using state-of-the-art artificial intelligence, HELIOS advances this to offer more efficient and accurate mass appraisal and property valuation. The results tell us that an ideal definition of homogeneous market areas and comparable properties is the task of AI more so than it is for humans; hence, it is capable of validating data and finding trends. The study identifies a number of social perils but promotes the idea that AI supports real estate analytics and is financed by a grant from the National Science Center. Additional work further aims at confirming the efficiency of HELIOS in performing homogeneous market analyses.

I.B., M.B.B, et al. [\[9\]](#), evaluates the effectiveness of fine-tuning Transformer models (BERT, RoBERTa, DeBERTa, GPT-2) for multi-class classification of hotel reviews outperforms prompt engineering for large language models (LLMs) (ChatGPT, GPT-4). We approach the problem with a multi-task learning framework, accomplishing both sentiment analysis and classification one-shot for the reviews under the rubric of service quality, ambiance, and food. The results show that the fine-tuned models—particularly RoBERTa—outperform classification tasks due to the high level and rich information, entailing faster contextual processing. ChatGPT and GPT-4 are great at sentiment analysis, yet this comes at a price of increased computational requirement, resulting in slower processing. The results indeed point toward the fact that multi-class classification on hospitality review data increases overall performance and efficiency with a fine-tuned model—emphasizing the most uniform model selection in accordance with task requirements. It further emphasizes the potential to perform domain-specific fine-tuning for better results and suggests future research for further optimization of NLP models in the hospitality industry.

Sallam, Malik. et al, [\[10\]](#). This systematic review provides a basis for the potential uses and limitations of ChatGPT in health care education, research, and practice. A systematic search following PRISMA was conducted in PubMed/MEDLINE and Google Scholar. In sum, the number of total eligible records extracted from both databases was 60. Key findings included some benefits identified in 85% of the records, but also pointed out issues in regard to ethics, copyright, transparency, and legal issues, while noting particularly risks related to bias, plagiarism, inaccurate content, and cybersecurity threats. The article notes that, although ChatGPT has huge potential to be an innovation driver in healthcare, it is pertinent for there to be put in place very stringent guidelines to engage with the stakeholders, forestall risks, and assure responsible use.

C.R.H., A.J.F., et al [\[11\]](#), explain large language models related to health care in clinical and surgical applications. Taking into consideration the PRISMA guidelines, the authors found 333 articles in the six databases within the year 2023 that met all the inclusion criteria; of those, 34 were relevant articles, including 14 original research articles, four letters, one interview, and 15 reviews impressively covering such a wide range of medical specialties. In all, major findings in this realm suggest that LLMs could possibly support diagnosis, guide treatment, patient triaging, and physician knowledge augmentation at the clinical setting and in documentation, surgical planning, and intraoperative guidance in surgical settings with promise to improve healthcare delivery but with a big question mark on accuracy, bias, and patient

privacy. Indeed, it is palpably clear that although LLMs are immensely powerfully qualitative, they should supplement—not replace—the judgment of health professionals.

A.Q. and A.M. of MDPI published [\[12\]](#), a study on deep learning techniques for the classification of CVs from Moroccan engineering students at ENSAK, Ibn Tofail University, using GRU, LSTM, and CNN with BERT and Gensim Word2Vec embeddings. In the paper, 867 CVs will be used that are related to five specializations in engineering. In this research, it turned out that models using BERT embeddings have relatively better performance compared to models using Gensim Word2Vec. Again, the closest to perfection accuracy in this case was 0.9351 for CNN-GRU/BERT. Again, it is models based on BERT that set the best precision and recall—0.9411—at CNN-GRU. This could hence show just how deep the competencies of BERT run in the representation of text toward automated classification of CVs and what kind of improvement in recruitment processes may be expected from such leading-edge methods.

N. F., A. H., et al. [\[13\]](#), contributes a deep learning application to layout recognition in historical documents for the enhancement of optical character recognition processes. In this work, among others, fully convolutional neural networks were applied along with background knowledge being integrated into the model, which was tested on five corpora from the 15th and 16th centuries. The methods used included baseline detection and several postprocessing techniques. The findings presented above show that on key findings, standard layouts could be recognized with an accuracy and recall of 99.9%, while complex layouts reached recognition rates as high as 90% by adding background knowledge. The research points to great potentials of layout recognition and OCR integration, where future work will entail optimizing small text detection and finally integrate layout detection with semantic classification in one model.

L. Z., J. M., et al. [\[14\]](#), proposes a workflow for 3D modeling of construction infrastructure using Terrestrial Laser Scanning, UAV photogrammetry, GNSS/IMU, computer vision, and AI algorithms. In the methods applied, LoFTR network for image matching, NeuralRecon for generating consistent 3D point clouds, and RPM-net for the co-registration of point clouds are used. In-vitro validation conducted on a high formwork project established an accuracy with a registration error of 5 cm. The results showed that the geometric accuracy of the generated 3D models compares to TLS models while being more complete and having a photorealistic appearance—important features for professionals to create visual documentation and further analysis.

R.Z., K.W., et al [\[15\]](#), proposed a deep multilabel multilingual document learning method for cross-lingual document retrieval. Their approach uses a six-layer fully connected network for the projection of cross-lingual documents into a shared semantic space, and hence enables the calculation of semantic distance between them. Unlike traditional methods that operate at the word level through cross-lingual comparisons, MDL does so at the document level and infuses multilabel supervision signals from the data itself. This eliminates ambiguities in manually induced labels and therefore enhances the discriminative power of the embeddings. It is also efficient in training since each language is trained separately. The experiments conducted on Wikipedia data consisting of 800k entries in four languages demonstrate that MDL is more than 30% better than state-of-the-art methods at document retrieval tasks. The authors have provided that in future, improvements can be made by incorporating cross-lingual knowledge bases and soft target supervisory signals to promote document features further. This work was supported by the National Natural Science Foundation of China and in part by the Development Project of Jilin Province of China.

X.J., L.L.O., et al. [\[16\]](#), is on early detection of gout flares through nurses' chief complaint notes in the Emergency Department using Natural Language Processing. In this paper, some NLP techniques are reviewed, such as classic, sparse text representations, like tf-idf, or dense encodings from medical domain-specific LLMs, including RoBERTa-large-PM-M3-Voc and BioGPT. The paper considers severe data imbalances with the use of methods such as oversampling, class weights, and focal loss. Results indicated that traditional tf-idf representations reached an F1 score above 0.75, while RoBERTa-large-PM-M3-Voc and BioGPT models reached a maximum F1 score of 0.8 and 0.85 respectively for the 2019 and 2020 datasets. Discriminative models performed better; however, very good results were also obtained when combining generative models as feature extractors with support vector machines. Results show an efficient application of LLMs, particularly domain-specific LLMs, in medical text processing and for the detection of GF. This might translate into the early diagnosis and follow-up care of patients with gout. No funding from external sources was received for this research, and the data used were publicly available from PhysioNet.

Z.S. and V.B. et al. [\[17\]](#), apply large language models—GPT models—to the static code analysis of the CWE-653 vulnerability at the front end of applications, particularly Angular. They applied the GPT API with a set of prompts for the identification and ranking of sensitive code segments and their protection level using few-shot examples and chain-of-thought techniques to preprocess and analyze static code. The proposed methodology was tested on

open-source projects; GPT-4 scored 88.76% in detecting vulnerabilities, outperforming GPT-3.5. The results demonstrate that LLMs have huge potential for improving code quality and vulnerability detection. That definitely demands more research and application of AI in software engineering. This work was partially supported by the Ministry of Innovation and Technology NRD Office and the Artificial Intelligence National Laboratory Program.

Bhavani Malisetty and A. J. Pereira [18], provide a performance observation for quantized LLMs on generation and analysis tasks pertaining to IoT privacy policy texts. For their experiments, the authors contrast 4-bit, 5-bit, and 8-bit quantized versions of the Llama 2 model against a baseline, namely, a non-quantized Llama 2 model. Results show that quantized models are very close in performance to the base model, indicating how they will be able to handle IoT privacy policy language on consumer-grade devices with minimal performance loss. Future work could be fine-tuning, quantization-aware training, and newer model evaluation.

Anjia Ye , et al. [19], present a hybrid methodology that combines large language models with human input, a process in a position to assist with the streamlining of systematic reviews. Specifically, their approach incorporates a semiautomated LLM-assisted workflow called Gemini-Pro, wherein LLMs summarize research articles by extracting key information to facilitate more rapid human decision-making for inclusion and exclusion decisions. The approach is intended to combine human expertise with LLM capabilities to reduce errors and increase accuracy. In a case study, it was shown that this hybrid approach detected 1.53% more misclassified articles than a human-only process, while on the remaining articles, human decisions were matched. Results show that such workflow may increase efficiency and accuracy in SRs by solving problems such as temporal lag and cognitive load.

Jonghyeon Yang [20], address the limitations of current GeoQA systems by using semantic similarity to analyze location-related questions from the MS MARCO dataset. They cluster semantically similar questions with an embedding-based topic modeling approach, extracting latent geographic topics in the process. This approach will help avoid the failures of traditional semantic parsing methods and really make sense of user interests within the geographic domain. The study indicates that the embedding-based topic modeling is pretty powerful for the discovery of coherent geographic topics. It further assists GeoQA systems in bringing a system closer to satisfying user requirements. However, this study points out that the dataset can be outdated and biased. Thus, further studies are warranted using more recent and diversified datasets.

2.2 Discussion

From the literature summary , we decided to use the LLMs(Large Language Model) for our project . We decided to implement an LLM in our project, instead of traditional machine learning or deep learning or RNNs etc., since these models are proven to understand and generate human languages accurately and contextually relevant. These large language models are pre-trained on enormous data volumes, which enables their performance in complex tasks on human languages, like sentiment analysis or text classification, much more powerfully and efficiently than the traditional models. This in turn improves performance on tasks that require fine-grained human language understanding, making it most appropriate for the goals of our project.

CHAPTER-3

PROPOSED WORK, IMPLEMENTATION, INTERFACES AND COMMUNICATION

The proposed work includes several stages, including data collection and preprocessing, model selection, model training and finetuning, model evaluation, to achieve the specific goals outlined below.

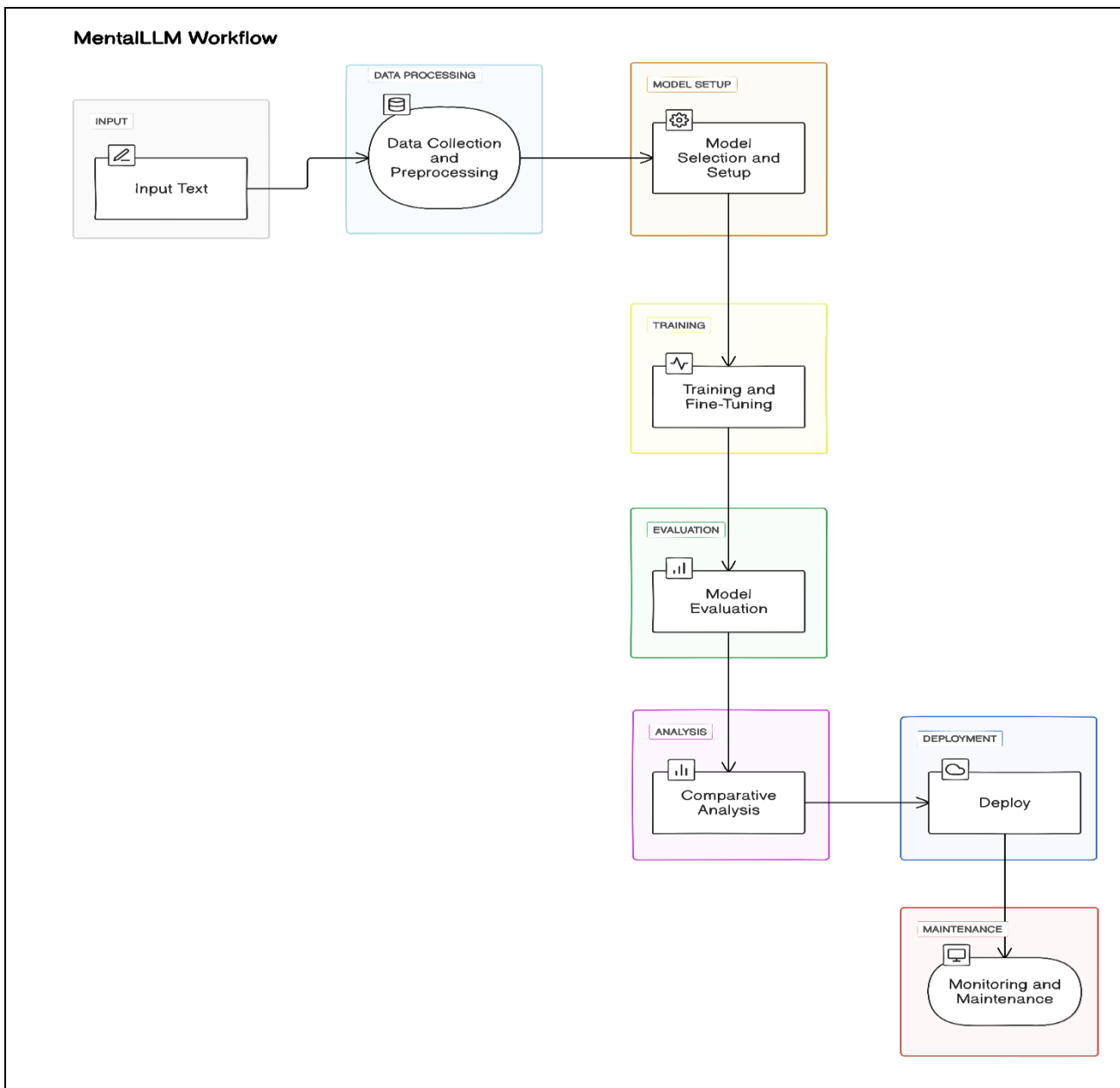


Figure 3.1 Architecture Diagram

3.1 DATA COLLECTION

Four data sets are mostly used in analysis related to mental health. Reddit is the source of these high quality and readable available data sets. we used those that have been annotated or supervised by human experts.

- **Dreaddit:** This dataset collected posts via Reddit PRAW API from Jan 1, 2017 to Nov 19, 2018, which contains ten subreddits in the five domains (abuse, social, anxiety, PTSD, and financial) and includes 2929 users' posts. Multiple human annotators rated whether sentence segments showed the stress of the poster, and the annotations were aggregated to generate final labels.
- **DepSeverity:** This dataset leveraged the same posts collected in Dreaddit, but with a different focus on depression. Two annotators followed DSM-5. (The DSM-5: Classification and criteria changes) and categorized posts into four levels of depression: minimal, mild, moderate, and severe.
- **SDCNL:** This dataset also collected posts from Python Reddit API, including Suicide Watch and Depression from 1723 users. Through manual annotation, they labelled whether each post showed suicidal thoughts.
- **CSSRS-Suicide:** This dataset contains posts from 15 mental health-related subreddits from 2181 users between 2005 and 2016. Four practicing psychiatrists followed Columbia Suicide Severity Rating Scale (C-SSRS) guidelines to manually annotate 500 users on suicide risks in five levels: supportive, indicator, ideation, behavior, and attempt.

We selected DepSeverity as our dataset from among these datasets. As our project tries to categorize depression severity, is particularly relevant to the DepSeverity dataset because it addresses depression and its severity levels. This dataset's alignment with our project objective ensures that the data we use is directly applicable to our study.

Initially our dataset was imbalanced as shown in the graph below:

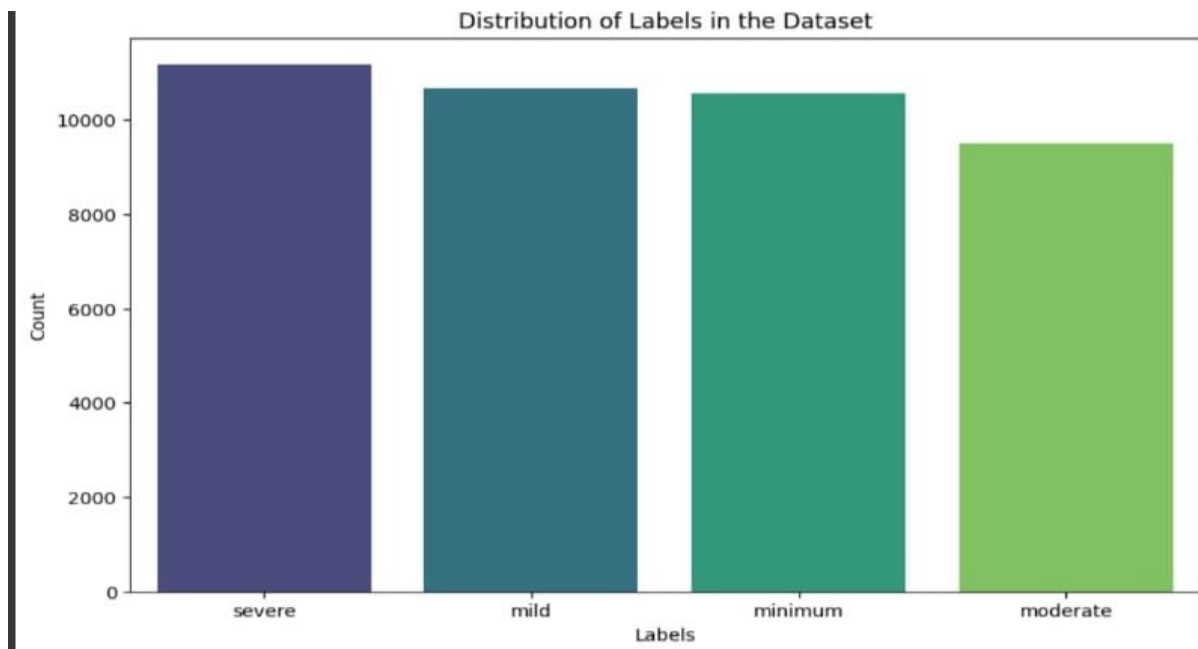


Figure 3.2. Imbalanced Distribution of Labels in the dataset

We made the dataset balanced for the further process. Our dataset consists of four labels: minimum, mild, moderate, severe. The Distribution of Labels are given below:

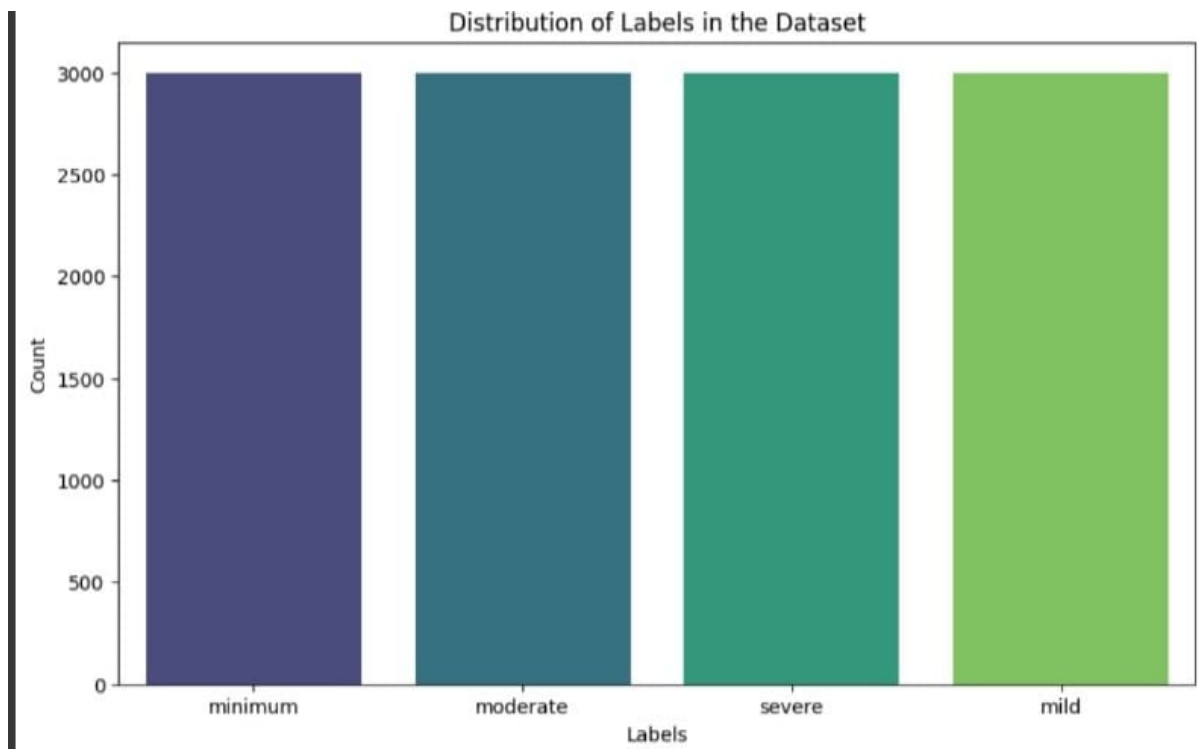


Figure 3.3 Balanced Distribution of Labels in the dataset

3.2 DATA PRE-PROCESSING

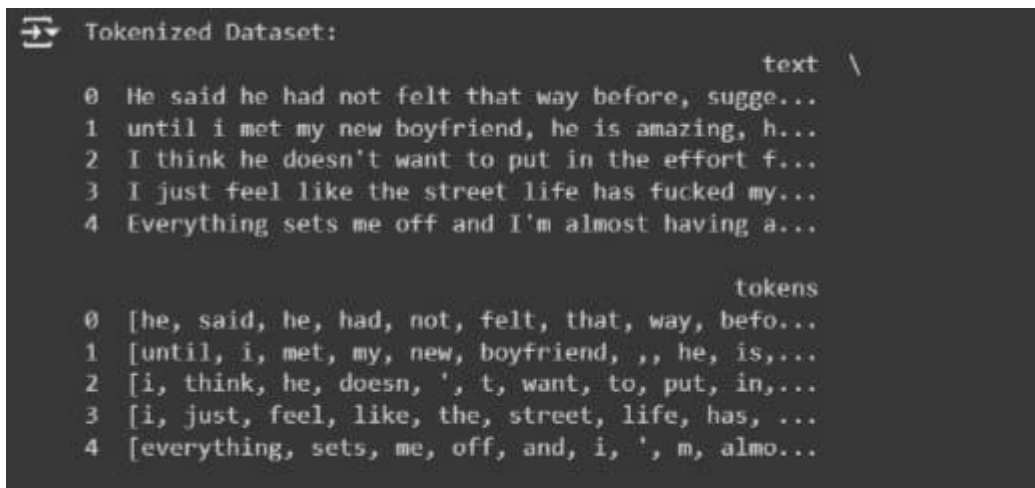
The process of cleaning and preparing raw data into machine understandable format is known as pre-processing. Pre-processing will change the data into a format that makes it easier for the model to understand and process. This is an important step because the model's performance can be greatly affected by the pre-processed data.

The pre-processing steps which we have used for Bert model are: tokenization, special token addition, padding, truncation, and attention masks.

3.2.1 Tokenization

Tokenization is the process of turning text data into that format of data that the model can understand.

While the simplest forms of token-based embedding layers deal with raw text, the BERT model deals with numerical tokens or, in general, representations of text. After tokenization, our dataset text broke down into units that are then translated into numerical IDs understandable by the model.



```
Tokenized Dataset:

text \
0 He said he had not felt that way before, sugge...
1 until i met my new boyfriend, he is amazing, h...
2 I think he doesn't want to put in the effort f...
3 I just feel like the street life has fucked my...
4 Everything sets me off and I'm almost having a...

tokens
0 [he, said, he, had, not, felt, that, way, befo...
1 [until, i, met, my, new, boyfriend, ,, he, is,...
2 [i, think, he, doesn, ', t, want, to, put, in,...
3 [i, just, feel, like, the, street, life, has, ...
4 [everything, sets, me, off, and, i, ', m, almo...
```

Figure 3.4. Before tokenization and After tokenization

3.2.2 Adding Special Tokens

Adding the special tokens like [CLS] and [SEP].

- We used the [CLS] (classification token), to the start of the input, to compile data for classification tasks.
- We have used separator token [SEP] is used to divide the sentences.


```

Token Lengths:
      text  token_length
0  He said he had not felt that way before, sugge...      155
1  until i met my new boyfriend, he is amazing, h...      313
2  I think he doesn't want to put in the effort f...      146
3  I just feel like the street life has fucked my...       92
4  Everything sets me off and I'm almost having a...      151
Length After Adding Special Tokens:
      text \
0  He said he had not felt that way before, sugge...
1  until i met my new boyfriend, he is amazing, h...
2  I think he doesn't want to put in the effort f...
3  I just feel like the street life has fucked my...
4  Everything sets me off and I'm almost having a...

      input_ids_length_with_special_tokens
0                      157
1                      315
2                      148
3                       94
4                      153

```

Figure 3.5. Length of the sentence before and after adding the CLS and SEP tokens.

3.2.3 Padding

BERT models are designed to work on fixed-size input. Padding guarantees that shorter sequences have a special [PAD] token filling them up to the wanted length, thus allowing batch processing. According to the input size, different lengths were taken care by padding. After padding the length of the sentences in the dataset were updated to max_length.

```

Padded Input IDs Lengths:
      text  padded_input_ids_length
0  He said he had not felt that way before, sugge...      512
1  until i met my new boyfriend, he is amazing, h...      512
2  I think he doesn't want to put in the effort f...      512
3  I just feel like the street life has fucked my...      512
4  Everything sets me off and I'm almost having a...      512

```

Figure 3.6. Length of the sentence after padding

3.2.4 Truncation

Sequences that are longer than the model's maximum length are shortened by truncation.

The maximum input length for BERT is 512 tokens. This will be considered as the threshold, the text that is longer than this threshold will be trimmed. This will make sure that the input size will be within the model size.

```

➡ After Padding and Truncation:
                                input_ids_with_special_tokens
0  [101, 2002, 2056, 2002, 2018, 2025, 2371, 2008...
1  [101, 2127, 1045, 2777, 2026, 2047, 6898, 1010...
2  [101, 1045, 2228, 2002, 2987, 1005, 1056, 2215...
3  [101, 1045, 2074, 2514, 2066, 1996, 2395, 2166...
4  [101, 2673, 4520, 2033, 2125, 1998, 1045, 1005...

```

Figure 3.7. Before and after truncation

3.2.5 Attention Mask

The tokens which are used classify the padding text and the real text is shown by the Attention Mask. The attention mask facilitates the model's ability to distinguish between padding tokens and real tokens. This is significant because the model's predictions shouldn't be affected by padding tokens. The attention mask directs the attention mechanism of the model by giving padding tokens a value of 0 and actual tokens a value of 1.

```

                                attention_mask
0  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
1  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
2  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
3  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
4  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...

```

Figure 3.8. Attention Mask

The other steps like Stemming, Lemmatization, Removing of Stop Words and Punctuation are not included in our pre-processing because of following reasons:

Stemming and Lemmatization: Not needed because BERT's tokenizer handles word variations through subwords.

Stop Words: Not removed because BERT can derive meaningful context from them.

Punctuation and Special Characters: Retained because BERT can utilize them to understand text structure.

3.3 MODEL SELECTION

For project, we planned to examine different Large Language Models (LLMs), including Bert, Llama3-70b-8192, Mistral-8X7b-32768, Gemma-7b-it.

BERT (Bidirectional Encoder Representations from Transformers)

BERT uses a transformer architecture. It specifically uses the encoder part of the transformer. This model reads the entire text at once. AS the name Bidirectional Encoder Representations from Transformers itself suggests that BERT uses a bidirectional approach, it considers the text from both the directions i.e., left and right of a word. BERT is specifically known as Masked Language Modeling. It was pre-trained on the large corpus of text including the entire Wikipedia and Books Corpus. BERT is known for two special techniques named Masked Language Modeling (MLM) and Next Sentence Prediction (NSP).

SPECIFICATIONS OF BERT:

Architecture specifications:

Bert is transformer based. It contains 12 layers for BERT-base and 24 layers for BERT-large. Each layer has a hidden size of 768 for BERT-base, 1024 for BERT-large. Attention Heads are 12 for BERT-base, 16 for BERT-large. Total Parameters are Approximately 110 million for BERT-base, 340 million for BERT-large.

Training specifications:

Masked Language Modeling (MLM): Randomly masks some tokens from the input, and the objective is to predict the original tokens.

Next Sentence Prediction (NSP): A binary classification task to predict whether a given sentence B is the actual next sentence of a given sentence A.

Tokenization specifications:

Tokenizer is a wordpiece tokenizer, vocabulary size would be 30,000 tokens.

ALGORITHM FOR BERT:

Initialize Model

- **Input:** Pre-trained model weights.
- **Output:** BERT model initialized with Transformer architecture.

Tokenize Input Text

- **Input:** Raw text.
- **Output:** List of tokens.
- **Steps:**
 - i. Split text into words or subwords using a tokenizer.

Create Input Embeddings

- **Input:** List of tokens.
- **Output:** Input embeddings.
- **Steps:**
 - i. **Token Embeddings:** Convert tokens into token embeddings.
 - ii. **Segment Embeddings:** Create segment embeddings for sentence pairs.
 - iii. **Positional Embeddings:** Create positional embeddings for token positions.
 - iv. **Combine:** Sum token embeddings, segment embeddings, and positional embeddings.

Process through Transformer Layers

- **Input:** Input embeddings.
- **Output:** Contextualized embeddings.
- **Steps:**
 - i. For each Transformer encoder layer:
 - **Self-Attention:**
 - Compute attention scores for each token.
 - Update token embeddings based on attention scores.
 - **Feed-Forward Neural Network:**
 - Pass updated embeddings through a feed-forward network.
 - Apply layer normalization and residual connections.

Pre-Training Tasks

- **Masked Language Modeling (MLM)**
 - **Input:** Input embeddings with some tokens masked.

- **Output:** Predicted tokens for masked positions.
- **Steps:**
 - Predict masked tokens based on context from other tokens.

- **Next Sentence Prediction (NSP)**

- **Input:** Sentence pairs.
- **Output:** Prediction of whether the second sentence follows the first.
- **Steps:**
 - Determine if the second sentence logically follows the first.

Fine-Tuning for Specific Tasks

- **Input:** Task-specific data (e.g., labeled text).
- **Output:** Fine-tuned model.
- **Steps:**
 - i. Add task-specific output layer.
 - ii. Train model on task-specific data.
 - iii. Update model weights using backpropagation and optimization.

Make Predictions

- **Input:** New input text.
- **Output:** Task-specific predictions (e.g., classification, question answering).
- **Steps:**
 - i. Tokenize and create embeddings for new input text.
 - ii. Process through the Transformer layers.
 - Extract output embeddings (e.g., [CLS] token for classification).
 - iii. Generate predictions based on output embeddings.

FINE-TUNING OF BERT

Additional training of the pre-trained model on the specific data of any particular downstream task is referred to as fine-tuning BERT. This includes the first step of preparation of input data

in the required format and labeling according to the tasks; this may be classification, as in the case of text classification or question answering. A task-specific layer is then added to the model, for instance a softmax layer in case of classification, and trained. The input text will first be tokenized into token IDs by BERT's tokenizer. Then the model initialized with pre-trained weights will be trained on the task specific data using some task specific loss function and update both the original layers of BERT and the new added layer. This is normally done with the Adam optimizer and weight decay. Then, it checks the model's performance on a validation set; after that, one is able to further tune parameters such as learning rate or batch size. Fine-tuning complete, utilize the rich, contextual embeddings learned during pretraining to make predictions on new, unseen data that are tailored for the needs of this downstream task.

Table 3.1: Fine-Tuning of Parameters

PARAMETERS	SPECIFICATIONS
Batch at each epoch	32,16,8
Learning Rate	$5e^{-5}, 1e^{-5}$
Weight_decay	None,0.01
Loss Function	Cross Validation

Llama 3-70b-8192

LLaMA stands for Large Language Model Meta AI. LLaMA model is a transformer based with large number of parameters. In the name, 70b denotes that the model has 70 billion parameters and 8192 indicates the token limit that the maximum sequence of word that model can process.

By default, it will be LLaMA 3-70b-8192 with its impressive 70 billion parameters, in such a way that very complicated patterns of the language are captured and high accuracy is attained for many NLP tasks. Equipped with its 8,192 tokens, it can handle and memorize context through long sequences of text. This model would, therefore, be very suitable for applications requiring long contexts, such as the generation of long-form content and creating detailed text analyses. It also ensures high-quality output and state-of-the-art performance on a diverse range of complex language understanding and generation tasks via advanced performance coupled with versatility based on state-of-the-art transformer-based architecture.

MISTRAL-8X7b-32768

Mistral is a model designed for the specific NLP tasks. In the name, 8X7b suggests that the each component or sub-model is comprised of 7 billion parameters which is summing up to 56 billion parameters and 32768 is the token limit which is suitable for tasks requiring extended context understanding.

Mistral-8X7b-32768 has a 32,768-token limit, so it is quite appropriate for handling longer contexts, such as tasks involving long documents or detailed dialogues. It has a capacity of 56 billion parameters, hence high learning and generalization ability, which enables it to ensure accuracy with nuance across a large bank of NLP tasks. It is an all-versatile model; due to its prowess, it's going to handle advanced applications in content generation, translation, and summarization, while the state-of-the-art capabilities would ensure superior performance and deep contextual comprehension.

Gemma-7b-it

Gemma model also utilizes transformer architecture with 7 billion parameters. IT suggests that the model is specifically pre-trained for Italian language.

These models represent advancements in NLP, each designed for specific types of language understanding and generation tasks, leveraging their architectures and training paradigms to handle complex and diverse applications in AI and machine learning. We used all the above mentioned models in our project for the text classification.

Gemma-7b-it is chosen because of specialized pre-training in the Italian language, and 7 billion parameters give a high capacity for language understanding and generation. In addition, it provides high performance due to its transformer architecture in treating complex tasks of text classification. It is focused on the Italian language, hence understanding the language more precisely and subtly. This makes it very effective in projects dealing with Italian text. You will be using Gemma-7b-it among other models including Mistral-8X7b-32768. This set of capabilities is designed against different language needs and task complexities, hence increasing the effectiveness of your text classification project.

3.4 FINE-TUNING

Fine-tuning is a very important step in our project where we are using the pre-trained BERT model to a specific task, such as predicting mental health conditions from textual data. During fine-tuning, the model is trained on a depseverity dataset to improve its performance for the

target task. In this section, we will detail the process of fine-tuning the BERT model on our ‘depseverity’ dataset, which contains text data labelled with different levels of depression severity.

The fine-tuning process involves setting training parameters and running the training loop. We used the Trainer class from the transformers library to simplify this process.

The parameters which we used in the fine-tuning process are as follows:

- num_train_epochs
- per_device_train_batch_size
- per_device_eval_batch_size
- weight_decay
- learning_rate
- logging_dir
- logging_steps
- evaluation_strategy
- save_strategy
- load_best_model_at_end
- fp16=True

Here, using these parameters, getting the output is totally a trial and error method. The whole parameters which we changed and how we used is mentioned below.

Firstly, we used the Hyperparameters which are given below:

Table 3.2. Hyperparameters & Specifications

num_train_epochs	3
train_batch_size	32
Eval_batch_size	32

By using this parameters whose number of epochs are 3, training and evaluation size are 32, warmup steps are 500, weight decay is 0.01, We got the Training Loss starting from 7.9 which led to the drastic output’s which we didn’t expected. The figure given below is the result obtained by using above Hyperparameters mentioned in **Table 3.2**

Step	Training Loss
10	7.979200
20	7.964600
30	7.896400
40	7.772100
50	7.638400
60	7.496000

Figure 3.9. Training Loss using different Hyperparameters

Then, we decided to change the batch size from 32 to 8, and also we included the parameter weight decay. Here, we also we observed the training loss increasing but not with the values as before. This time the values started from 0.97 and increased till 1.20. And here, we also observed that validation loss decreasing and accuracy is increasing.

Table 3.3. Adjusted Hyperparameters for Improved Accuracy

num_train_epochs	3
train_batch_size	8
eval_batch_size	8
weight_decay	0.01

Epoch	Training Loss	Validation Loss	Accuracy
1	0.976600	1.492620	0.747896
[10335/10335 2:18:53, Epoch 3/3]			
Epoch	Training Loss	Validation Loss	Accuracy
1	0.976600	1.492620	0.747896
2	1.003600	1.273983	0.792888
3	1.205700	1.242453	0.818723

Figure 3.10. Training Loss, Validation Loss, Accuracy using different Hyperparameters

Then, We increased the number of epochs from 3 to 21, and also included the parameter fp16. This time, training loss got decreased but the validation loss increased.

Table 3.4. Hyperparameters for Extended Epoch Training

num_train_epochs	21
train_batch_size	8
eval_batch_size	8
weight_decay	0.01
Learning_rate	$5e^{-5}$
evaluation_strategy	epoch
save_strategy	epoch
Fp16	True

Epoch	Training Loss	Validation Loss	Accuracy
1	0.654500	0.574931	0.727771
2	0.497700	0.551117	0.741474
3	0.443900	0.610832	0.740560
4	0.288200	0.667096	0.742540
5	0.296400	0.976641	0.745585
6	0.160100	1.100528	0.732643
7	0.177100	1.099890	0.720311
8	0.193600	1.186748	0.716809
9	0.144900	1.359595	0.713307
10	0.118300	1.422682	0.736145
11	0.146000	1.434882	0.711328
12	0.109500	1.572946	0.716809
13	0.126400	1.602374	0.720311

Figure 3.11. Training Loss, Validation Loss, Accuracy using different Hyperparameters

And then, we decreased the number of epochs from 21 to 19. Also, increased the batch_size from 8 to 16 and decreased the learning rate from $5e^{-5}$ to $1e^{-5}$ as slower learning will lead to good results. Here, we got the output accuracy as 81% and the validation loss is stopped at 1.2 which is the least one according to the previous one. So, we finalize this model to be used in our further process.

Table 3.5. Model Training Hyperparameters and Settings

num_train_epochs	19
train_batch_size	16
eval_batch_size	16
weight_decay	0.01
Learning_rate	1e-5
evaluation_strategy	epoch
save_strategy	epoch
Fp16	True

Epoch	Training Loss	Validation Loss	Accuracy
1	0.625300	0.620051	0.743229
2	0.588000	0.503625	0.791146
3	0.312600	0.457605	0.825521
4	0.307800	0.478218	0.828646
5	0.216000	0.579617	0.827604
6	0.174300	0.716938	0.822917
7	0.087300	0.827269	0.829167
8	0.078700	0.925423	0.827604
9	0.014700	1.029447	0.826562
10	0.029100	1.110884	0.824479
11	0.001900	1.116342	0.833854
12	0.046900	1.156017	0.831771
13	0.045100	1.217961	0.833333
14	0.001400	1.253067	0.829688
15	0.064100	1.285063	0.827083
16	0.038600	1.281457	0.832292
17	0.004200	1.273108	0.831250
18	0.000200	1.295811	0.832812

Figure 3.12. Training Loss, Validation Loss, Accuracy using different Hyperparameters

Here, we saved output dir to save the model checkpoints, logs which will be helpful to monitor the training and to resume if it is interrupted. Number of training epochs are for complete passes through the training dataset. It also prevents from the overfitting. Per_device_train_batch we used the batch size as 32,16,8. Among these three sizes we got the optimum output for batch size 16. batch size will help in speed up training. Similarly, for eval batch size too. Weight decay is used to overcome the problem of overfitting. We have reduced the overfitting by using

the L2 regularization, which is also known as ridge regularization. It will prevent the overfitting by adding the penalty to the large weights.

Learning rate is the step which updates the weights. Smaller learning rate results in good results. Logging_dir is a directory to save the logs. It will helps in tracking the training process. Logging steps will provides insights into the training process and helps detect issues early. Evaluation stratergy is to monitor the validation performance and detect overfitting.

Save_stratergy is to save the model at the particular point. We saved the model at each epoch.

Load_best_model_at_end ensures that the best-performing model is used for further tasks, avoiding models that might have overfitted later. Fp16 Speeds up training and reduces memory usage without significantly affecting model accuracy.

Table 3.6. Parameter Specifications and Optimal Values

PARAMETERS	SPECIFICATION	OPTIMUM
Batch at each epoch	32,16,8	16
Learning Rate	5e-5,1e-5	1e-5
Weight_decay	None,0.01	0.01
Loss Function	Cross Validation	Cross Validation

3.5 Process of Integration and Interface

In this project, with the help of MERN, we developed a web application using React for the front end and Fast API for the back end. With the help of these two technologies, we were able to deliver a seamless and efficient web application with a great user experience.

CHAPTER-4

RESULTS

4.1 CONFIGURATION

We have used the free version of Google Colab for preprocessing the dataset and finetuning the model. The run type in the Google Colab is Python3 and the Hardware accelerator is T4 GPU.

4.2 EVALUATION PARAMETERS

Evaluation metrics measure how well and how efficiently the model performs, including accuracy, precision, recall and F1-score.

These parameters including confusion matrix, precision, recall, F1-score, and accuracy—are so vital in your project, such that completeness of all details regarding model performance is adhered to. They point exactly to which levels of depressions are posting correct predictions and which usually might get misclassified. They help to bring improvement from the source, so that your model may be perfectly accurate and reliable, especially in reports to be used in sensitive applications for predicting mental health conditions.

Let us look into the metrics that are used in the BERT model evaluation.

4.3 ACCURACY

Accuracy is nothing but the ratio of correctly predicted values to the total values or instances. It is used to measure the performance for a classification model. In other words, it measures up to what level the model predictions are correct.

Purpose: Gives an overview of how the model is performing by telling how many of the predicted instances are correct against the total instances.

The formula used to calculate the accuracy is:

Accuracy=Number of Correct Predictions/Total Number of Predictions

(or)

Accuracy=(TP+TN)/(TP+TN+FP+FN)

where:

- TP (True Positives): The number of positive values that are correctly predicted by the model.
- TN (True Negatives): The number of negative values that are correctly predicted by the model.
- FP (False Positives): The number of negative values that are incorrectly predicted as positive by the model.
- FN (False Negatives): The number of positive values incorrectly predicted as negative by the model.

```

Accuracy: 0.8276
Precision: [0.76781609 0.90675991 0.89721254 0.72821577]
Recall: [0.70464135 0.84199134 0.98282443 0.76304348]
F1 Score: [0.73487349 0.87317621 0.93806922 0.74522293]

```

Figure 4.1. Evaluation Metrics of BERT model

CONFUSION MATRIX:

Purpose: Have an absolute view of how good or bad your classification model's performance is by contrasting the true labels against the predicted labels.

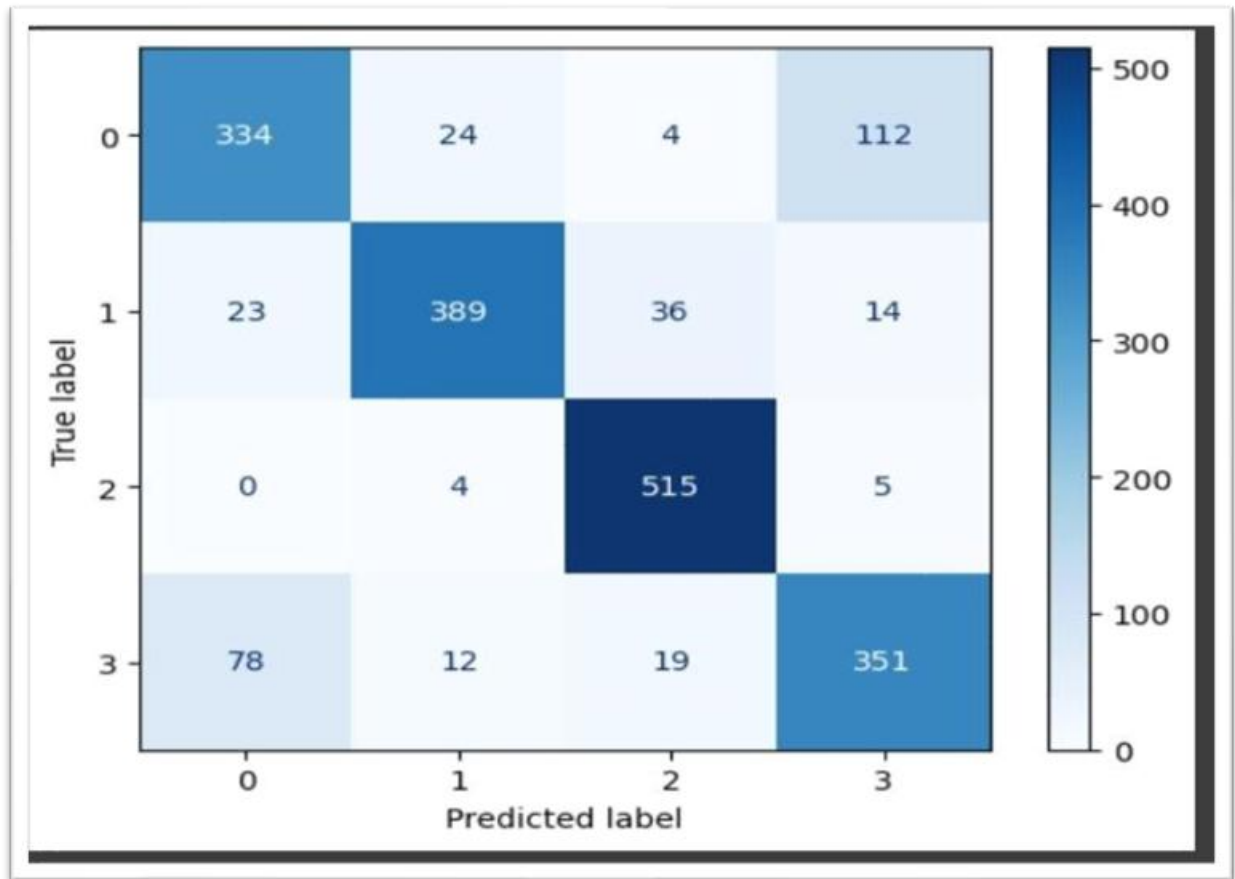


FIGURE 4.2 CONFUSION MATRIX

a)PRECISION:

The ratio of correctly predicted positive observations to the total predicted positive values is called as Precision.

Purpose: It tells how much of the Predicted Positive Cases were actual Positive Cases. High Precision means Low False Positives.

The formula used to calculate the Precision is:

$$\text{Precision} = (\text{TP})/(\text{TP}+\text{FP})$$

b)RECALL:

It is the ratio of correctly predicted positive observations to the all observations in the actual class.

Purpose: It tells how many of the Actual Positive Cases were correctly predicted by the model. High Recall means Low False Negatives.

The formula used to calculate the Recall is:

$$\text{Recall} = (\text{TP})/(\text{TP}+\text{FN})$$

c)F1-SCORE:

It is the weighted average of Precision and Recall. It considers both false positive and false negatives.

Purpose: It is a trade-off between precision and recall. Moreover, it gives only a single number for both false positives and negatives, therefore useful in case of class imbalance.

The formula used to calculate the F1-Score is:

$$\text{F1-Score} = (2 * (\text{Precision} * \text{Recall})) / (\text{Precision} + \text{Recall})$$

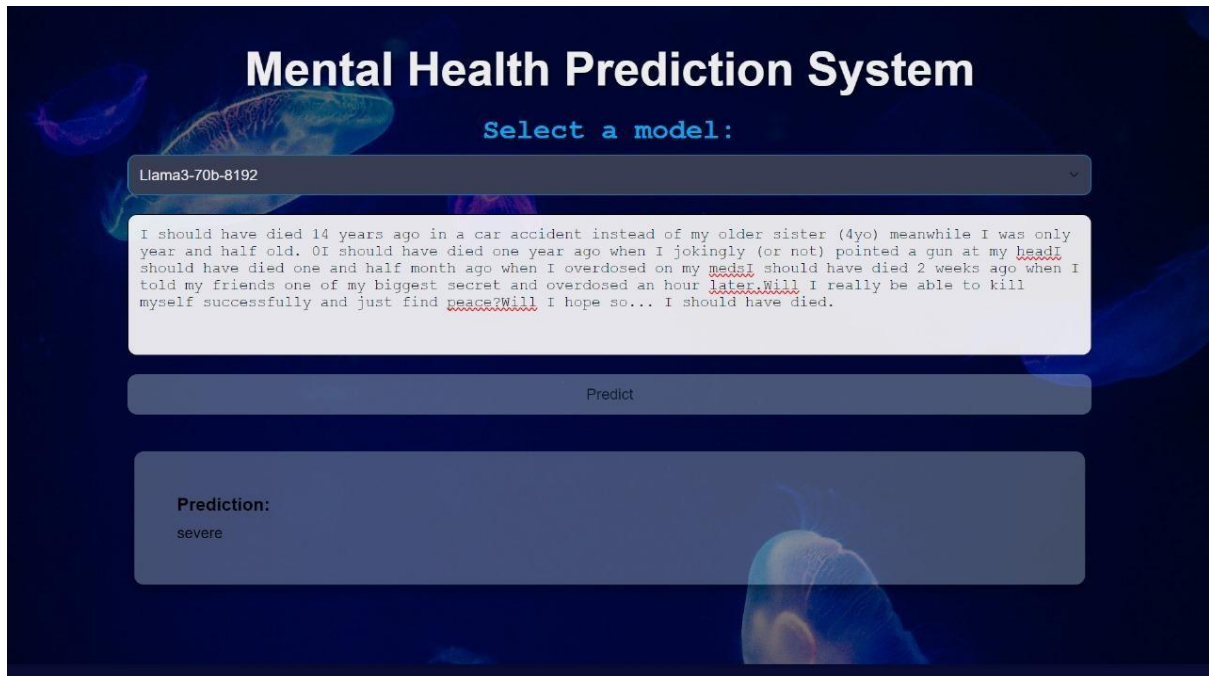
4.4 QUALITATIVE ANALYSIS

Now let's look into the predictions of our API models and fine-tuned BERT model on mental health predictions which gives us the labels mild, minimum, moderate, severe.

Let's take few examples:

Ex 1 of input text:

I should have died 14 years ago in a car accident instead of my older sister (4yo) meanwhile I was only year and half old. 0I should have died one year ago when I jokingly (or not) pointed a gun at my head I should have died one and half month ago when I overdosed on my meds. I should have died 2 weeks ago when I told my friends one of my biggest secret and overdosed an hour later. Will I really be able to kill myself successfully and just find peace? Will I hope so... I should have died.



The screenshot shows a web interface titled "Mental Health Prediction System". Below the title is a "Select a model:" dropdown menu with "Llama3-70b-8192" selected. A text input field contains the same text as in the previous block. Below the input field is a "Predict" button. At the bottom, a "Prediction:" label is followed by the word "severe". The background of the interface features a dark blue gradient with faint, glowing jellyfish-like shapes.

Figure 4.3. PREDICTION OF LLAMA on Severe Class

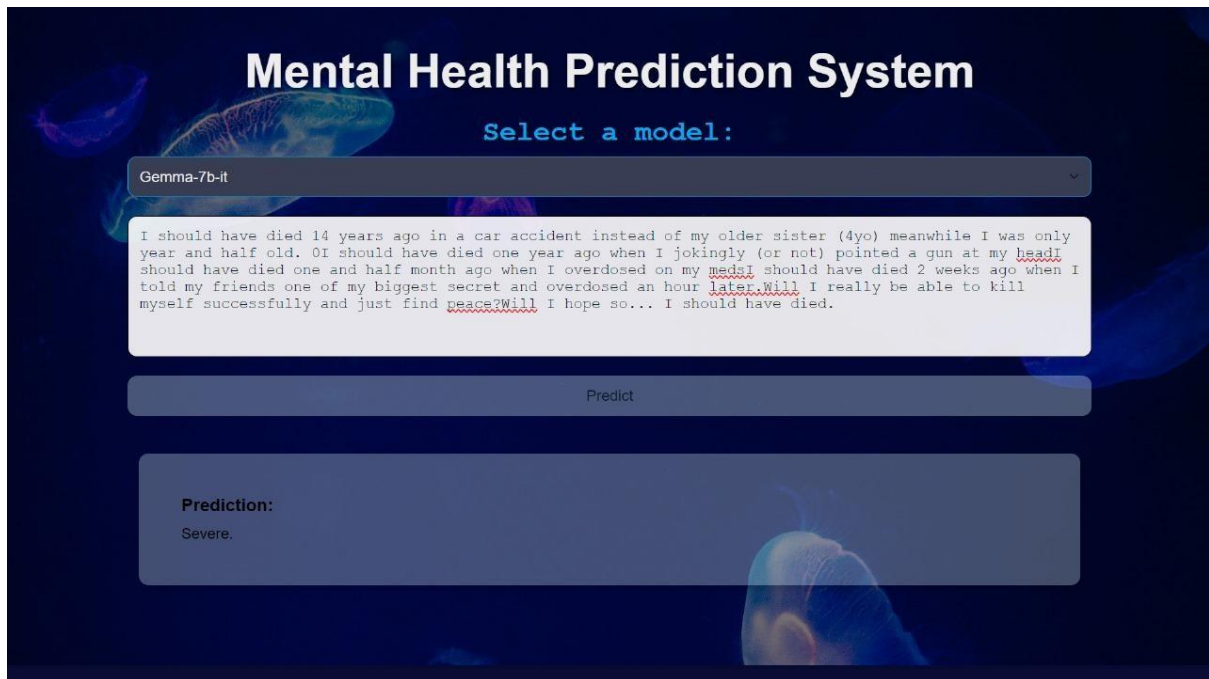


Figure 4.4. PREDICTION OF GEMMA on Severe Class

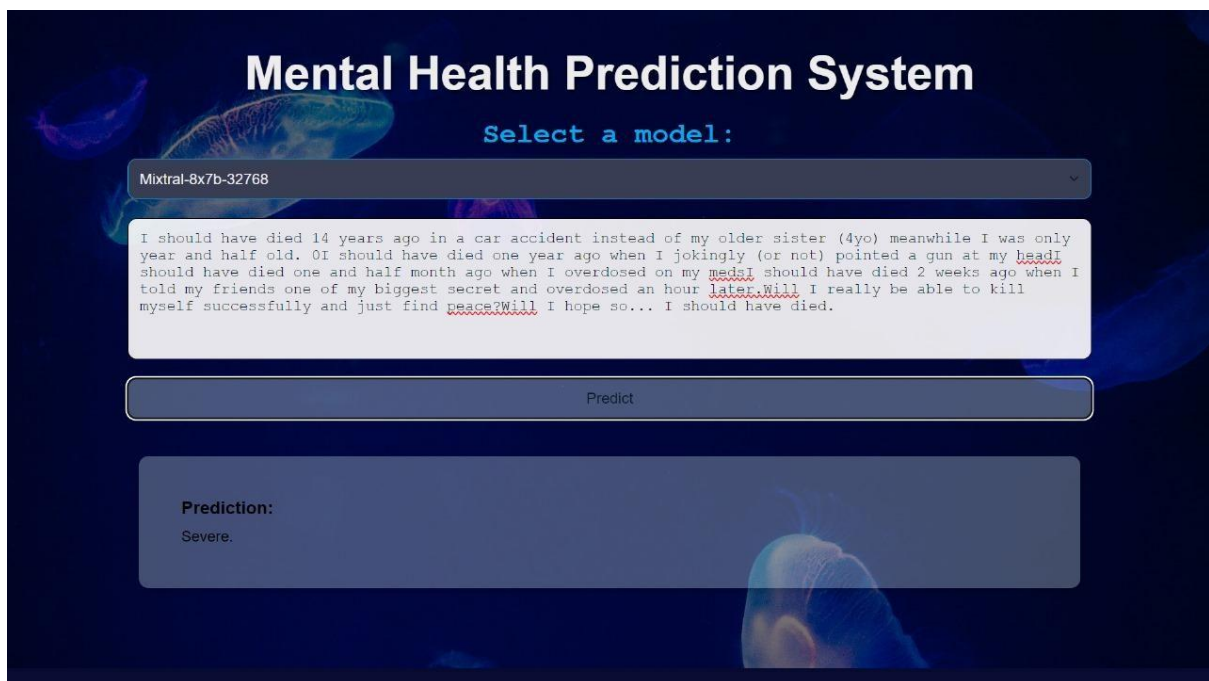


Figure 4.5. Prediction of Mistral on Severe Class

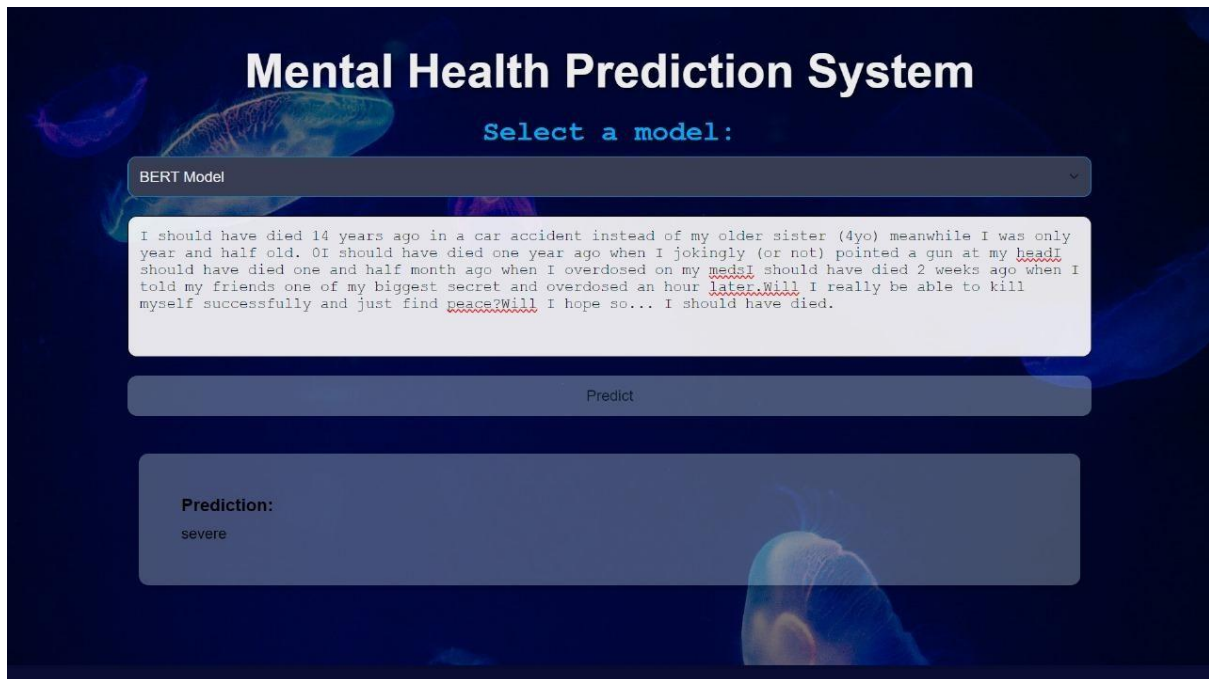


Figure 4.6. PREDICTION OF BERT on Severe Class

Ex 2 of input text :

I've never spoken to anyone about my anxiety but I'm pretty sure I have generalized anxiety disorder. When I was young I used to be very bright and would take charge of projects and doing assignments. As time went on I became lazier but still fairly on top of things. When I went into college I suffered and things never clicked. Doing even the most simple of tasks or assignments were just so difficult for me.

Mental Health Prediction System

Select a model:

Llama3-70b-8192

I've never spoken to anyone about my anxiety but I'm pretty sure I have generalized anxiety disorder. When I was young I used to be very bright and would take charge of projects and doing assignments. As time went on I became lazier but still fairly on top of things. When I went into college I suffered and things never clicked. Doing even the most simple of tasks or assignments were just so difficult for ~~me~~ my brain chemical thing like low dopamine

Predict

Prediction:
moderate

Figure 4.7. PREDICTION OF LLAMA on Moderate Class

Mental Health Prediction System

Select a model:

Gemma-7b-it

I've never spoken to anyone about my anxiety but I'm pretty sure I have generalized anxiety disorder. When I was young I used to be very bright and would take charge of projects and doing assignments. As time went on I became lazier but still fairly on top of things. When I went into college I suffered and things never clicked. Doing even the most simple of tasks or assignments were just so difficult for ~~me~~ my brain chemical thing like low dopamine

Predict

Prediction:
Moderate

Figure 4.8. PREDICTION OF GEMMA on Moderate Class

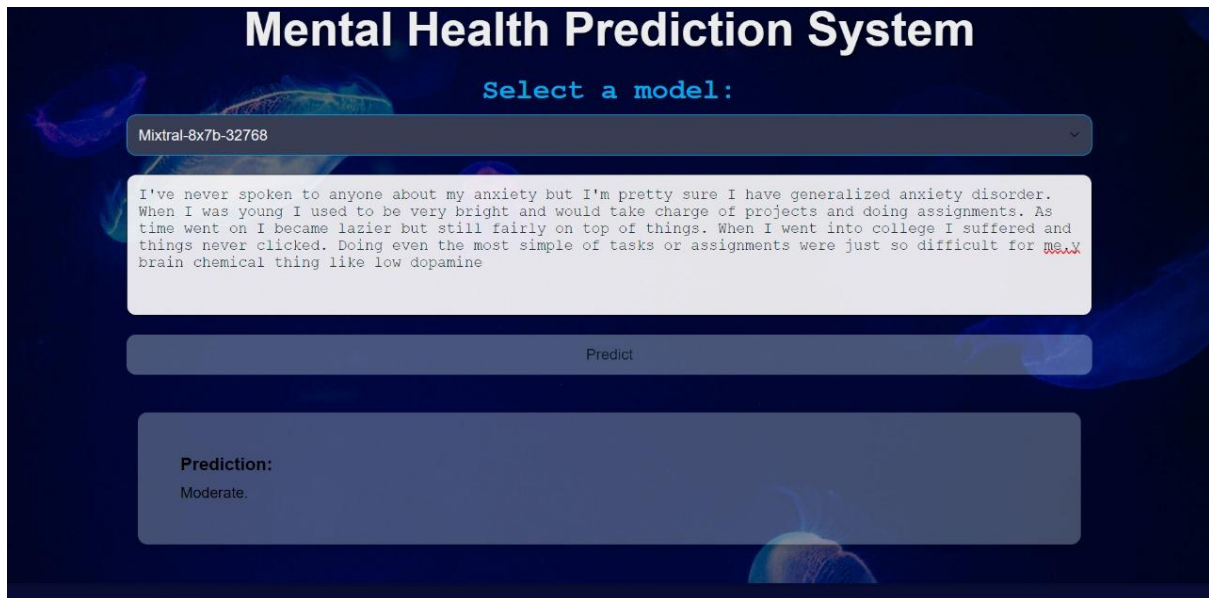


Figure 4.9: PREDICTION OF MISTRAL on Moderate Class

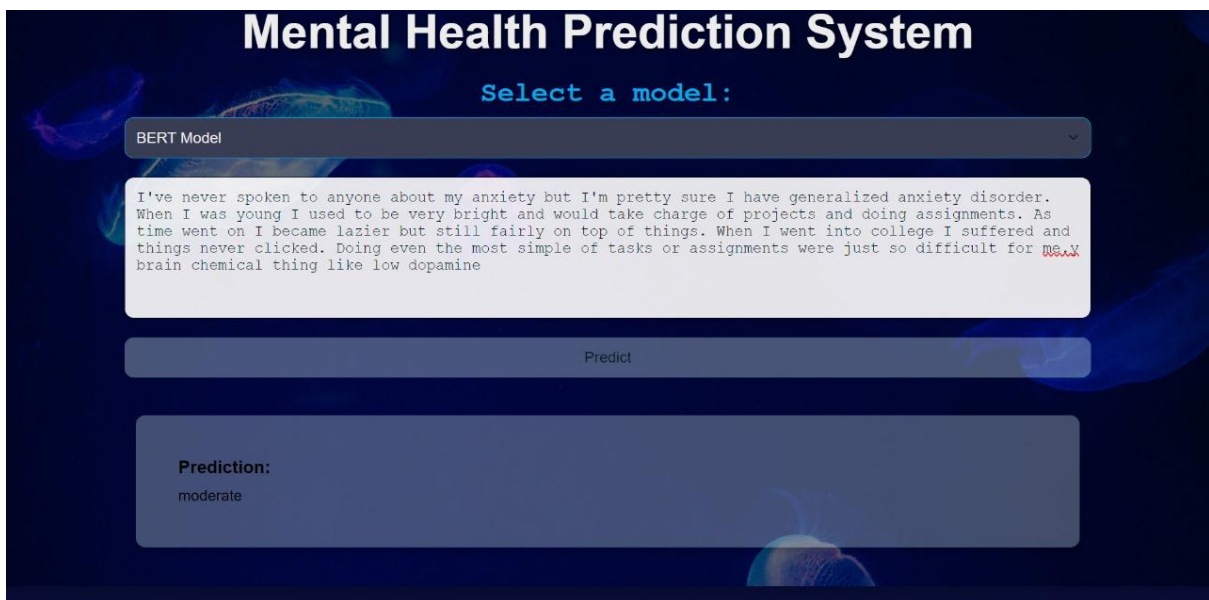


Figure 4.10. PREDICTION OF BERT MODEL on Moderate Class

4.5 QUANTITATIVE ANALYSIS

The dataset we took for the BERT Model basically contains 4 labels they are, **mild**, **minimal**, **moderate**, **severe**. It is one of the Classification based problem dataset, which is multi classified dataset.

We named this each label with Classes. For **mild** label we assigned **Class 0**, **minimal** label we assigned **Class 1**, **moderate** label we assigned **Class 2**, **severe** label we assigned **Class 3**.

By different evaluation parameters we had calculated Precision, F1-score, Recall each label by using the BERT model. Let's see the value of each label that is predicted by the BERT model.

	precision	recall	f1-score	support
Class 0	0.77	0.70	0.73	474
Class 1	0.91	0.84	0.87	462
Class 2	0.90	0.98	0.94	524
Class 3	0.73	0.76	0.75	460
accuracy			0.83	1920
macro avg	0.83	0.82	0.82	1920
weighted avg	0.83	0.83	0.83	1920

FIGURE 4.11. CLASSIFICATION REPORT

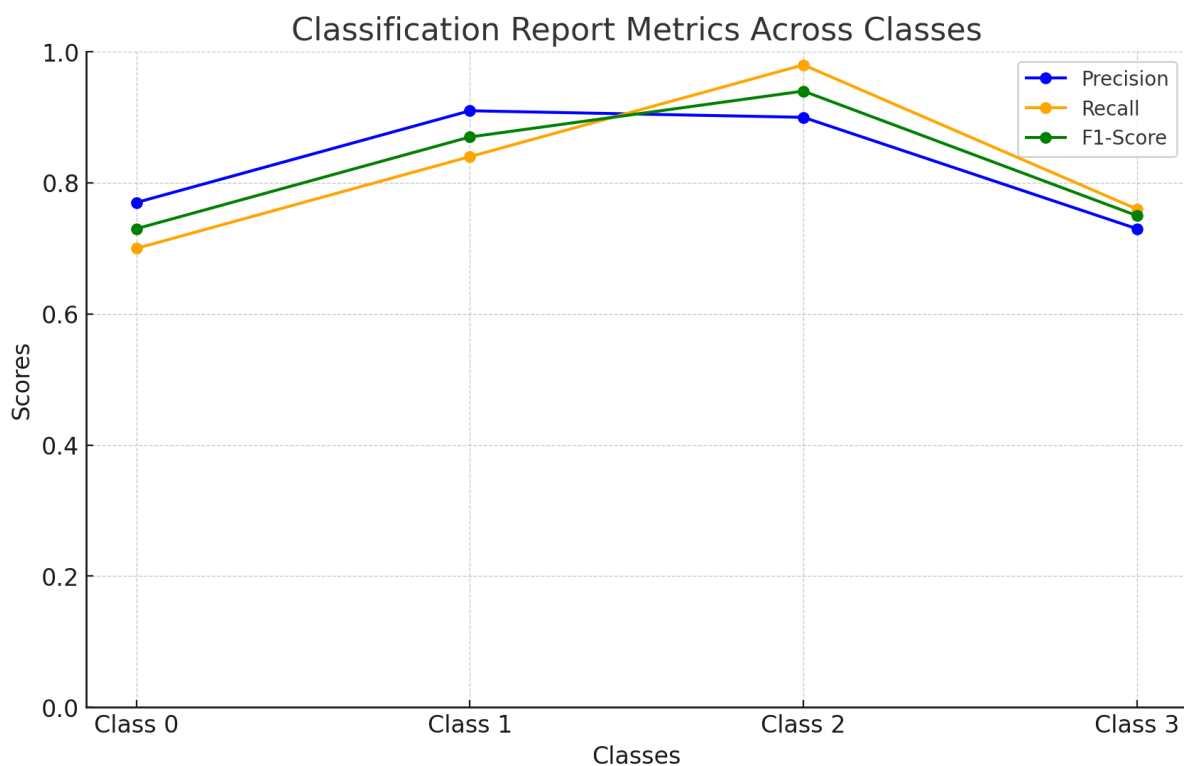


Figure 4.12. Classification Reports Metrics Across Classes

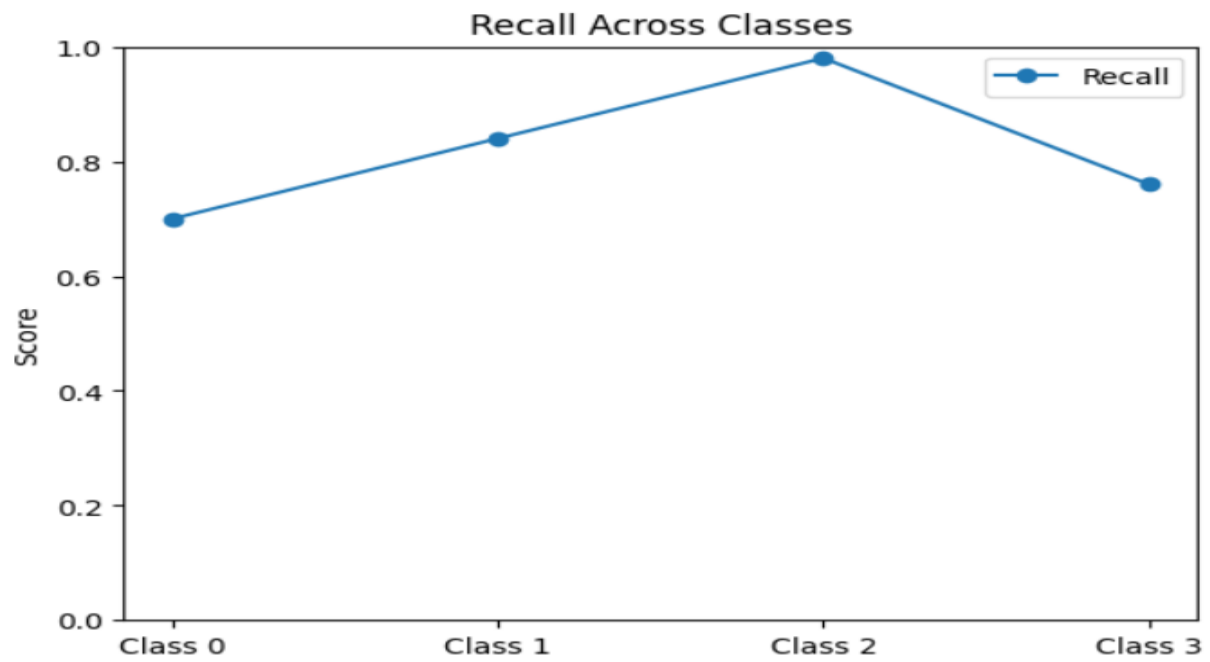


Figure 4.13. Recall Across Classes

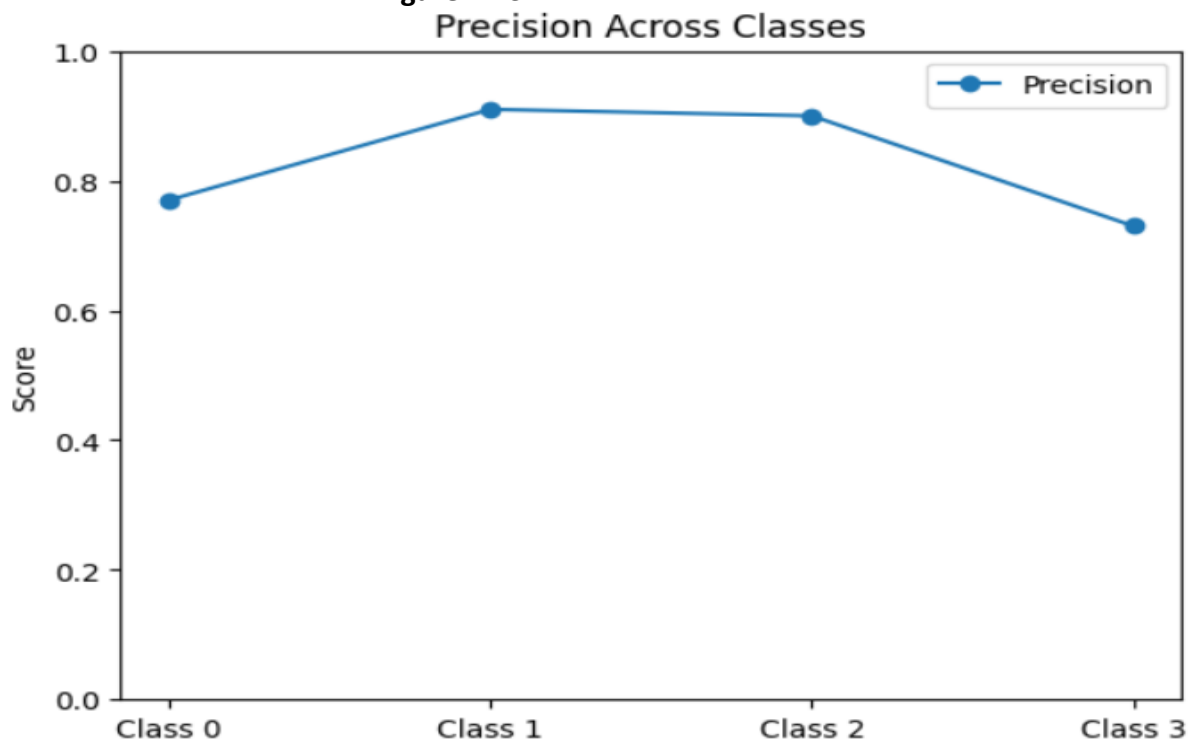


Figure 4.14. Precision Across Classes

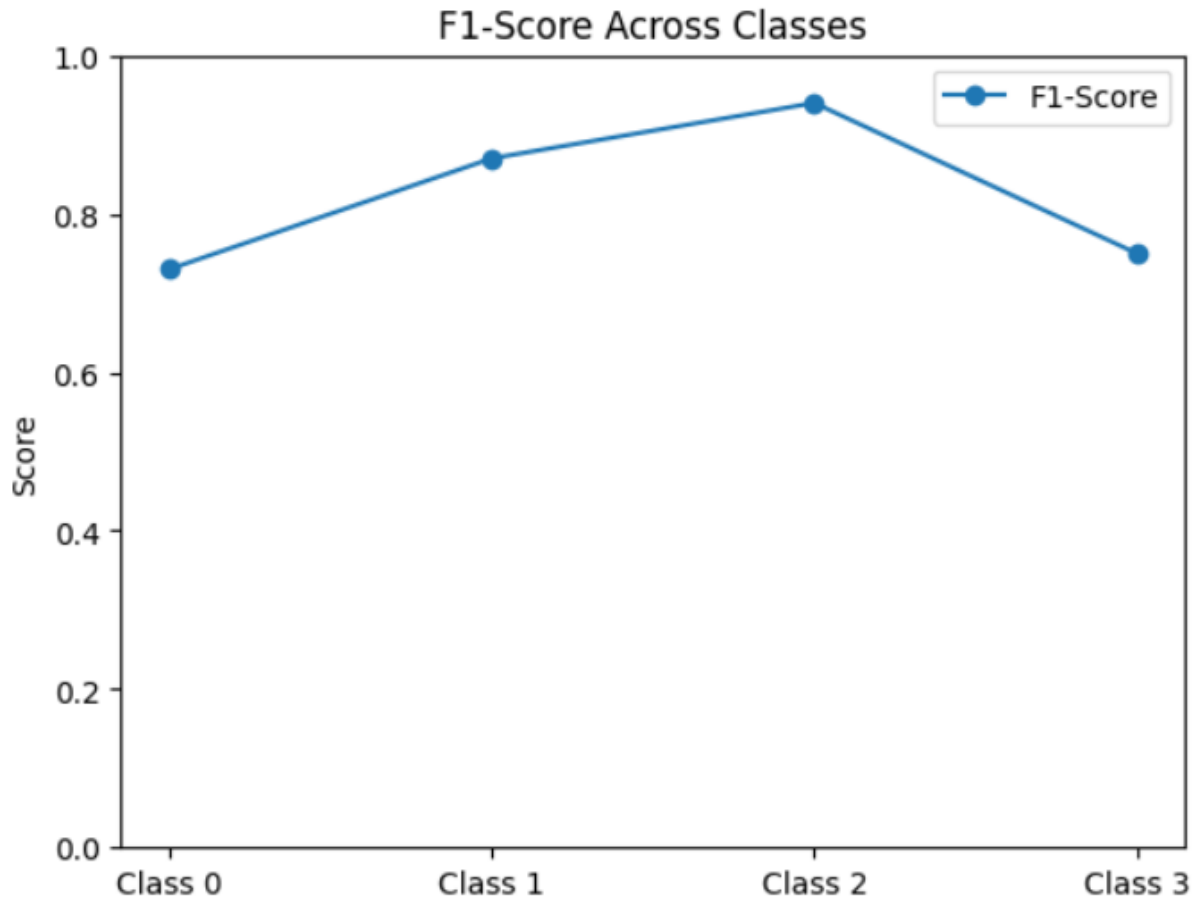


Figure 4.15. F1-Score Across Classes

4.6 SUMMARY OF RESULTS (QUALITATIVE ANALYSIS & QUANTITATIVE ANALYSIS)

Of all these choices, the one that is probably the best applicable and most befitting this text is Gemma's prediction, because it offers a full assessment; a rating of severity; supportive guidance; empathy in offering judgment; and encouragement to find a solution.

It takes seriously, into account the triggering event concerning the woodworking competition, manifestation of symptoms of PTSD, and the aftereffect of emotions from the bad dream. It classifies the mental state as one of moderate depression, rated 7/10, entailing serious emotional distress.

It encourages one to seek the assistance of friends, family, or even professional help from a mental health expert. It is advice meant to encourage people to develop into a more pragmatic

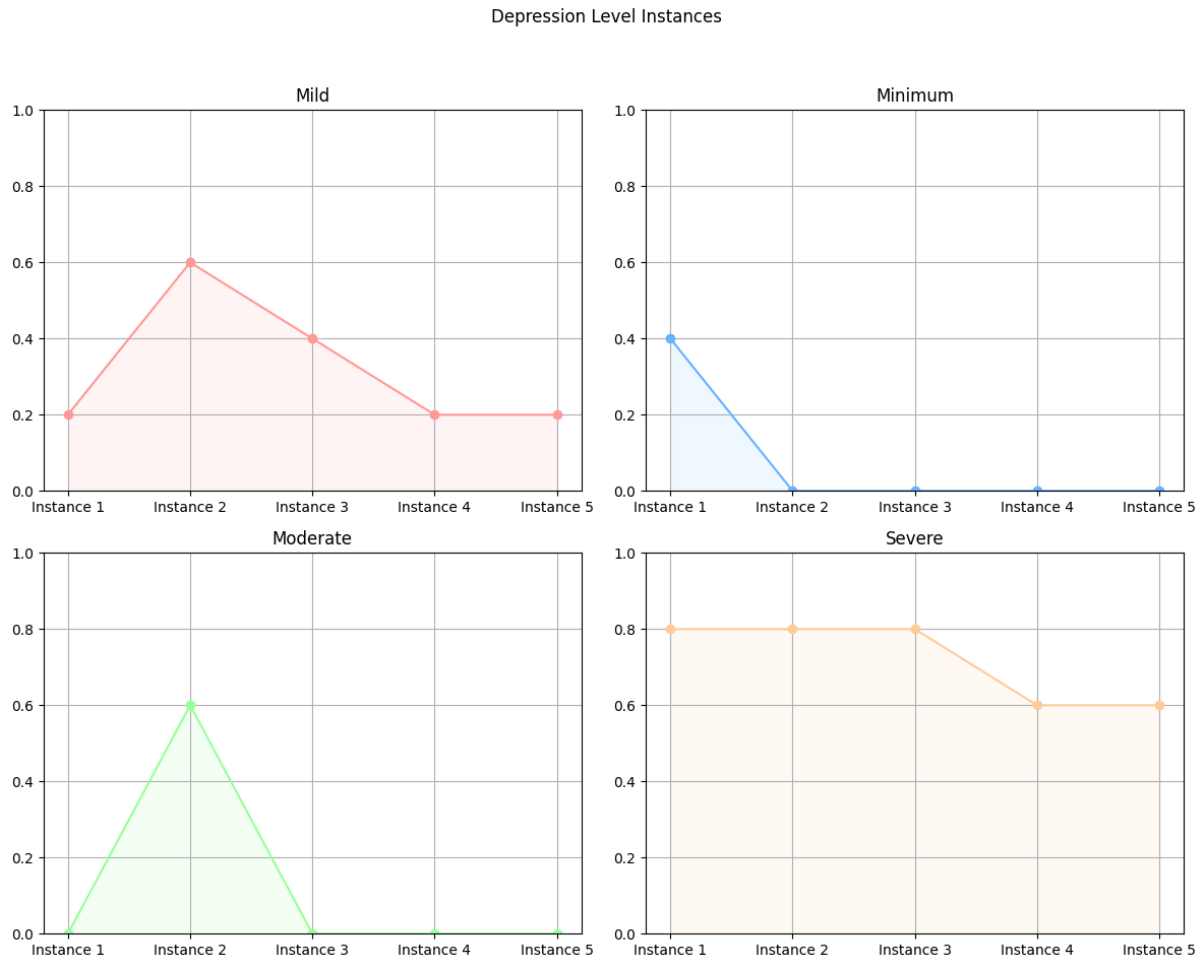


Figure 4.16 .CONFIDENCE SCORE OF EACH CLASS

stride of efforts toward the management of emotional distress and the seeking of proper care and assistance.

The empathetic tone that Gemma communicates supports the emotions of a person, adds value to the goals, and responds to complex emotional reactions. This equally weighted approach to understanding and advice makes a prediction by Gemma most fitting for contemporary mental health challenges.

We got the accuracy as 82% after fine-tuning the BERT model using our pre-processed Dataset. The BERT model gave the correct predictions for the given input text.

CHAPTER-5

CONCLUSION & FUTURE SCOPE

This project deals with predicting the severity of depression by advanced NLP and deep learning models, LLMs, especially a fine-tuned BERT model, to rate mental health. The project classifies text data based on four severity levels: minimum, mild, moderate, and severe, using a fine-tuned BERT model with the Groq API to boost its performance and further scale up. This provides a mental health professional with a tool that is invaluable, allowing for timely diagnosis and intervention.

It employs machine learning and deep learning models, including autoencoders, attention mechanisms, and graph neural networks, to increase the prediction capabilities by addressing many aspects associated with mental health conditions. Here it can be well anticipated that, except for some problems that are likely to be encountered in the context of data privacy, complexity of mental health data, and ethical considerations, the prospect of considering LLMs to predict mental health is promising, and this approach of the project brings optimism about the improvement of outcomes for patients and assisting health professionals in making the diagnosis and treatment of mental disorders better.

Opening the model code on Hugging Face allows collaboration and innovation within the community to jointly improve the potential of the technology in predicting mental health events. This, in general, helps build strong predictive models that reduce the occurrence and advanced features of mental health conditions and help in improving the mental health care system.

References

1. Kuchin, Yan, Ravil Mukhamediev, Nadiya Yunicheva, Adilkhan Symagulov, Kirill Abramov, Elena Mukhamedieva, Elena Zaitseva, and Vitaly Levashenko. 2023. "Application of Machine Learning Methods to Assess Filtration Properties of Host Rocks of Uranium Deposits in Kazakhstan" *Applied Sciences* Vol. 13, no. 19, pp. 10958, 2023
2. Li, Jue, and Chang Wu., "Deep Learning and Text Mining: Classifying and Extracting Key Information from Construction Accident Narratives", *Applied Sciences*, Vol. 13, no. 19, pp.10599, 2023.
3. Sharma, Dilip Kumar, Bhuvanesh Singh, Saurabh Agarwal, Lalit Garg, Cheonshik Kim, and Ki-Hyun Jung. 2023. "A Survey of Detection and Mitigation for Fake Images on Social Media Platforms" *Applied Sciences* Vol. 13, no. 19, pp.10980,2023.
4. Thafar, Maha A., Mashael M. Alsulami, and Somayah Albaradei. 2024. "FutureCite: Predicting Research Articles' Impact Using Machine Learning and Text and Graph Mining Techniques" *Mathematical and Computational Applications* Vol. 29, no. 4, pp. 59,2024.
5. Hidayat, Taufik, Kalamullah Ramli, Nadia Thereza, Amarudin Daulay, Rushendra Rushendra, and Rahutomo Mahardiko. 2024. "Machine Learning to Estimate Workload and Balance Resources with Live Migration and VM Placement" *Informatics* Vol. 11, no. 3, pp. 50,2024.
6. Marcillo, Pablo, Cristian Arciniegas-Ayala, Ángel Leonardo Valdivieso Caraguay, Sandra Sanchez-Gordon, and Myriam Hernández-Álvarez. 2024. "POLIDriving: A Public-Access Driving Dataset for Road Traffic Safety Analysis" *Applied Sciences* vol. 14, no. 14, pp. 6300,2024.
7. Kozov, Vasil, Boyana Ivanova, Kamelia Shoylekova, and Magdalena Andreeva. 2024. "Analyzing the Impact of a Structured LLM Workshop in Different Education Levels" *Applied Sciences* 14, no. 14: 6280,2024.

8. Janowski, Artur, and Malgorzata Renigier-Bilozor. 2024. "HELIOS Approach: Utilizing AI and LLM for Enhanced Homogeneity Identification in Real Estate Market Analysis" *Applied Sciences* vol. 14, no. 14, pp. 6135,2024.
9. Botunac, Ive, Marija Brkić Bakarić, and Maja Matetić. 2024. "Comparing Fine-Tuning and Prompt Engineering for Multi-Class Classification in Hospitality Review Analysis" *Applied Sciences* vol. 14, no. 14, pp. 6254,2024.
10. Sallam, Malik. 2023. "ChatGPT Utility in Healthcare Education, Research, and Practice: Systematic Review on the Promising Perspectives and Valid Concerns" *Healthcare* vol. 11, no. 6, pp.887,2024.
11. Pressman, Sophia M., Sahar Borna, Cesar A. Gomez-Cabello, Syed Ali Haider, Clifton R. Haider, and Antonio Jorge Forte. 2024. "Clinical and Surgical Applications of Large Language Models: A Systematic Review" *Journal of Clinical Medicine* Vol. 13, no. 11, pp. 3041,2024.
12. Qostal, Aniss, Aniss Moumen, and Younes Lakhri. 2024. "CVs Classification Using Neural Network Approaches Combined with BERT and Gensim: CVs of Moroccan Engineering Students" *Data* Vol. 9, no. 6,pp. 74,2024.
13. Fischer, Norbert, Alexander Hartelt, and Frank Puppe. 2023. "Line-Level Layout Recognition of Historical Documents with Background Knowledge" *Algorithms* Vol. 16, no. 3, pp. 136,2023.
14. Zhao, Linlin, Huirong Zhang, and Jasper Mbachu. 2023. "Multi-Sensor Data Fusion for 3D Reconstruction of Complex Structures: A Case Study on a Real High Formwork Project" *Remote Sensing* Vol. 15, no. 5, pp. 1264,2023.
15. Feng, Kai, Lan Huang, Hao Xu, Kangping Wang, Wei Wei, and Rui Zhang. 2022. "Deep Multilabel Multilingual Document Learning for Cross-Lingual Document Retrieval" *Entropy* Vol. 24, no. 7,pp. 943,2022.
16. Oliveira, Lucas Lopes, Xiaorui Jiang, Aryalakshmi Nellippillipathil Babu, Poonam Karajagi, and Alireza Daneshkhah. 2024. "Effective Natural Language Processing Algorithms for Early Alerts of Gout Flares from Chief Complaints" *Forecasting* Vol. 6, no. 1,pp. 224-238,2024.

17. Szabó, Zoltán, and Vilmos Bilicki. 2023. "A New Approach to Web Application Security: Utilizing GPT Language Models for Source Code Inspection" *Future Internet* Vol.15, no. 10, pp. 326,2023.
18. Malisetty, Bhavani, and Alfredo J. Perez. 2024. "Evaluating Quantized Llama 2 Models for IoT Privacy Policy Language Generation" *Future Internet* Vol. 16, no. 7,pp.224 ,2024.
19. Ye, Anjia, Ananda Maiti, Matthew Schmidt, and Scott J. Pedersen. 2024. "A Hybrid Semi-Automated Workflow for Systematic and Literature Review Processes with Large Language Model Analysis" *Future Internet* Vol. 16, no. 5,pp. 167,2024.
20. Yang, Jonghyeon, Hanme Jang, and Kiyun Yu. 2023. "Analyzing Geographic Questions Using Embedding-based Topic Modeling" *ISPRS International Journal of Geo-Information* Vol. 12, no. 2,pp.52,2023.