Detecting ADHD-Related Text Using Classical Machine Learning and Deep Learning Models

Sara Joshi
Department of Information Technology, MPSTME, NMIMS, Mumbai
sara.joshi14@nmims.in

***Abstract*— Attention Deficit Hyperactivity Disorder (ADHD) is a prevalent yet underdiagnosed neurodevelopmental disorder, often characterized by patterns of inattention, impulsivity, and hyperactivity. Traditional diagnostic methods, though effective, are subjective and time-consuming. With the growth of digital footprints on social media platforms like Reddit, there is increasing potential to leverage user-generated content for automated mental health assessment. This study investigates the efficacy of both classical machine learning and deep learning models in classifying ADHD-related text. Using a balanced dataset of 20,000 Reddit posts, we perform extensive preprocessing, feature engineering, and implement models including SVM, Random Forest, XGBoost, AdaBoost, and LSTM. Our results highlight the linguistic markers of ADHD-related discourse and compare model performance across traditional and deep learning paradigms. Our approach presents a scalable, accessible method for early ADHD screening, reducing reliance on manual clinical assessments.**

Keywords: *ADHD Detection, Machine Learning, Deep Learning, Text Classification, Digital Behavioral Data, Reddit*

I. INTRODUCTION

Attention Deficit Hyperactivity Disorder (ADHD) is a prevalent neurodevelopmental condition affecting approximately 5–7% of children globally [1]. It is marked by persistent patterns of inattention, hyperactivity, and impulsivity, significantly impacting academic performance, social relationships, and emotional well-being. Traditionally, ADHD diagnosis has relied on clinical interviews, behavioral assessments, and standardized questionnaires completed by parents and educators. Although effective, these diagnostic methods are time-consuming, resource-intensive, and often influenced by subjective interpretations, leading to delays in diagnosis and treatment [2].

In recent years, advances in machine learning (ML) and natural language processing (NLP) have opened new avenues for automating mental health assessments using digital behavioral data. Social media platforms like Reddit offer an abundance of user-generated content that may reflect users' cognitive and emotional states in real time. These unstructured texts have demonstrated potential for identifying mental health indicators that might not be captured through traditional clinical methods [3], [4].

While extensive research has been conducted in applying ML and deep learning (DL) for detecting conditions such as depression and anxiety, the automated detection of ADHD—particularly through text-based analysis—remains underexplored [5]. A few recent studies have shown promising results using behavioral data [1], EEG signals [2], and textual features [5], [6] for ADHD identification. However, there is limited focus on leveraging large-scale, real-world text data to capture linguistic markers associated with ADHD.

This study addresses this research gap by developing and evaluating a range of classical ML and DL models for classifying ADHD-related posts from Reddit. The primary contributions of this work include:

1. A systematic linguistic analysis of ADHD-related discussions on Reddit, identifying textual markers correlated with ADHD symptoms.
2. A comparative evaluation of traditional ML algorithms (e.g., Support Vector Machine, Random Forest) and deep learning models (e.g., LSTM) for accurate classification of ADHD-related content [3], [4], [5].
3. The construction of a publicly available annotated dataset to support further research in automated ADHD detection using digital behavioral data.

The overarching aim of this research is to enhance the early screening process for ADHD through scalable, automated methods that reduce reliance on manual clinical assessments. By utilizing real-world social media data and advanced classification techniques, this work aspires to contribute to more accessible and timely ADHD identification, ultimately supporting improved developmental outcomes for affected individuals.

II. LITERATURE REVIEW

Over the past decade, text-based analysis has emerged as a powerful tool for identifying mental health conditions such as depression, anxiety, and bipolar disorder through digital behavioral data. Social media platforms, particularly Twitter and Reddit, have been widely utilized to extract linguistic markers indicative of these conditions. Several studies have demonstrated that classical machine learning models like Support Vector Machines (SVM) and Logistic Regression can achieve classification accuracies ranging from 70% to 85% in distinguishing depressive content from general discourse [1]. More recently, transformer-based architectures such as BERT have shown superior performance by capturing nuanced semantic and contextual features in text, further improving detection accuracy for conditions like depression and anxiety [1].

Despite this progress, the application of text-based methods to ADHD detection remains relatively underexplored. The linguistic markers associated with ADHD differ significantly from those of other mental health conditions. Rather than emotional tone or sentiment, ADHD-related language often involves irregular attention spans, impulsivity, and fragmented or distracted speech patterns—elements that are harder to detect using conventional text analysis techniques [2], [5].

Classical Machine Learning Models in ADHD Research

Classical machine learning techniques have been extensively applied to ADHD detection, primarily in domains utilizing structured data such as demographics, behavioral assessments, and neuropsychological test results. For example, Maniruzzaman et al. [2] applied Logistic Regression and Random Forest algorithms to the National Survey of Children's Health (NSCH), achieving an accuracy of 85.5% in predicting ADHD diagnosis. Another study [3] employed Random Forest and neural network models using Continuous Performance Test (CPT) data, reporting classification accuracies exceeding 87%.

Parameswaran et al. [4] also demonstrated the effectiveness of classical ML models in predicting ADHD using demographic and behavioral features, while Li et al. [5] explored acoustic and textual markers derived from DIVA interviews for adult ADHD screening. However, these approaches primarily focus on structured, numerical inputs, leaving the domain of unstructured, user-generated textual content largely untapped.

Although applications of classical models like SVM and Random Forest to ADHD-related text remain limited, their advantages—such as model interpretability and performance on smaller datasets—make them valuable for initial explorations into ADHD text classification [5]. However, their inability to capture long-range dependencies and complex contextual relationships limits their performance when dealing with natural language, particularly in ADHD-related discourse, which often lacks clear structure.

Deep Learning for Mental Health Text Analysis

Deep learning approaches have significantly advanced the field of mental health detection through text. Architectures like Long Short-Term Memory (LSTM) networks and transformers have demonstrated exceptional capabilities in learning hierarchical, contextual representations from large-scale unstructured text data [1].

Bidirectional LSTM networks have shown state-of-the-art performance in classifying anxiety and depression-related posts on platforms like Reddit, owing to their ability to model temporal sequences and context. Transformer models like BERT further elevate this performance by capturing bidirectional dependencies and deeper semantic relationships within text, enabling more nuanced understanding and better classification outcomes [1].

Despite these strengths, the application of deep learning models to ADHD-specific text remains minimal. The disorder's linguistic footprint is less consistent and often lacks the emotional tone prevalent in depressive or anxious language [5]. ADHD-related posts may exhibit scattered thoughts, impulsive word choices, and non-linear expressions, which can pose significant challenges for both classical and deep learning models without sufficient labeled training data.

Datasets and Limitations

A key limitation in advancing ADHD detection via text lies in the scarcity of large-scale, publicly available annotated datasets. Existing resources, such as the NSCH or CPT datasets, provide valuable structured data but do not encompass the rich linguistic diversity of online user-generated content [2]. In contrast, platforms like Reddit offer vast amounts of real-world textual data, yet ADHD-specific corpora remain largely unavailable [5].

In addition, prior research in this area often suffers from class imbalance issues, where ADHD-related posts are significantly underrepresented compared to non-ADHD content. This imbalance can result in biased learning and reduced model accuracy for minority classes. Furthermore, existing studies frequently overlook the contextual richness and informal linguistic patterns that could serve as valuable indicators of ADHD [5].

To address these gaps, this study constructs a new, large-scale annotated ADHD dataset based on Reddit discussions. It also explores the comparative performance of classical ML models and deep learning architectures on this corpus. By releasing this dataset to the public, the study aims to support further innovation in the automated detection of ADHD through digital behavioral data.

III. PROPOSED WORK

In this paper, we propose the development of a machine learning and deep learning-based framework to identify ADHD-related linguistic patterns in Reddit posts. The goal is to evaluate whether these models can serve as scalable, non-clinical tools for early detection and pre-screening of ADHD.

We will systematically compare the performance of classical machine learning models—Support Vector Machine (SVM), Random Forest (RF), XGBoost, and AdaBoost with a deep learning approach using a Long Short-Term Memory (LSTM) network.
These models were selected for their respective strengths in handling high-dimensional data, interpretability, and sequence-based learning.
Our proposed work includes:

- Investigating the effectiveness of text-based features in distinguishing ADHD from non-ADHD users.
- Identifying which models perform best in capturing the subtle, context-dependent markers of ADHD in user-generated content.
- Addressing challenges of data noise, class imbalance, and generalizability to unseen data.

IV. METHODOLOGY

The methodology for this study involves a series of well-defined steps to preprocess and analyze text data, followed by the application of both classical machine learning models and deep learning techniques for the classification of ADHD-related and non-ADHD Reddit posts. The process encompasses data collection, text preprocessing, feature extraction, and the implementation of various models to compare their performance in distinguishing between ADHD and non-ADHD content.

Source of Data

The dataset used for this study was sourced from Kaggle. It consists of Reddit posts, where two separate datasets were used:

1. ADHD Dataset: This dataset contains posts from Reddit related to ADHD, identified based on the content in the titles and self-texts of posts.

2. Non-ADHD Dataset: This dataset contains Reddit posts that are classified as non-ADHD, based on general content and context.

Description of ADHD vs. Control Class

- ADHD Class: The posts labeled as ADHD were gathered from Reddit users discussing ADHD-related topics, experiences, or symptoms. These posts contain discussions about symptoms, diagnosis, medication, and lifestyle adjustments related to ADHD.

- Control Class (Non-ADHD): The posts in this category are those that do not relate to ADHD. These cover general topics unrelated to the disorder.

Size, Balance, and Preprocessing Steps

- Both datasets were reduced to 10,000 rows for each class (ADHD and Non-ADHD) to ensure balance and maintain computational efficiency.
- The combined dataset (20,000 rows) was shuffled and split into training and testing sets for model training and evaluation.
- Preprocessing steps such as text cleaning, tokenization, and feature extraction were carried out to prepare the text data for model training.

Text Preprocessing

The text data underwent the following cleaning steps:

- Conversion to lowercase.
- Removal of URLs and special characters (e.g., punctuation, digits).
- Removal of excess whitespace and newlines.
- Removal of stopwords using a predefined stopword list (in the case of TF-IDF vectorization).

Tokenization and Padding (For Deep Learning)

- Tokenization: Text data was tokenized into integer sequences, where each word was mapped to a unique integer.
- Padding: The sequences were padded to a uniform length (300 tokens) to ensure compatibility with deep learning models.

Vectorization Method

- TF-IDF: For classical machine learning models, a TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer was used with a vocabulary size of 5,000 words. This method converts the text into a sparse matrix of word features.
- Deep Learning Embedding Layer: For deep learning models, text data was tokenized and padded, followed by embedding each word into a fixed-dimensional vector space (embedding dimension of 64).

Models Used

Classical Models

1. SVM (Support Vector Machine): A linear SVM was used for text classification. It aims to find the optimal hyperplane that separates the two classes (ADHD and Non-ADHD).
2. Random Forest: A random forest classifier with 10 estimators was used to build an ensemble of decision trees to make predictions.
3. XGBoost: XGBoost is an efficient gradient boosting model that was used with 10 estimators to improve classification performance.
4. AdaBoost: The AdaBoost classifier was used to create an ensemble of weak classifiers, iteratively adjusting weights to minimize errors.

Deep Learning Model

LSTM (Long Short-Term Memory): An LSTM-based deep learning model was employed for text classification. It uses a sequence of layers to capture long-term dependencies in sequential data (text in this case). The model consists of:

1. An embedding layer (64-dimensional vectors).
2. A bidirectional LSTM layer to capture both forward and backward dependencies in text.
3. A Dense output layer with a sigmoid activation function for binary classification (ADHD vs. Non-ADHD).

V. EXPERIMENTATION & RESULTS

Evaluation Metrics

The following metrics were used to evaluate the performance of the models:

- Accuracy: The percentage of correct predictions over the total predictions.
- Precision: The proportion of positive predictions that were correct.
- Recall: The proportion of actual positives that were correctly identified.
- F1-Score: The harmonic mean of precision and recall, providing a balance between the two.
- Confusion Matrix: This matrix provides a detailed breakdown of the true positives, true negatives, false positives, and false negatives.

Train/Test Split

The dataset was split into training (80%) and testing (20%) sets. The models were trained on the training set and evaluated on the testing set.

Model	Accuracy	Precision (0)	Precision (1)	Recall (0)	Recall (1)	F1-Score (0)	F1-Score (1)
SVM	88%	0.90	0.87	0.86	0.91	0.88	0.89
Random Forest	95%	0.93	0.96	0.96	0.93	0.95	0.95
XGBoost	93%	0.88	0.99	0.99	0.87	0.93	0.92
AdaBoost	80%	0.71	1.00	1.00	0.61	0.83	0.76
LSTM	96%	0.98	0.97	0.97	0.98	0.98	0.97

Table 1: Performance Table of All Models

Confusion Matrix and Other Visualizations

The confusion matrices for each model were plotted to visually assess the performance of the models. These matrices help us understand how each model is performing in terms of true positives, false positives, true negatives, and false negatives.

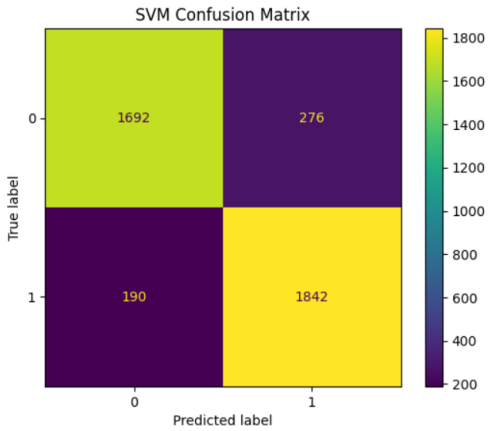

Fig 1.1: SVM Confusion Matrix

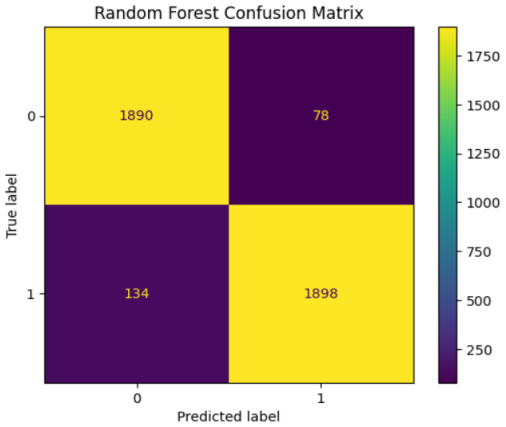

Fig 1.2: Random Forest Confusion Matrix

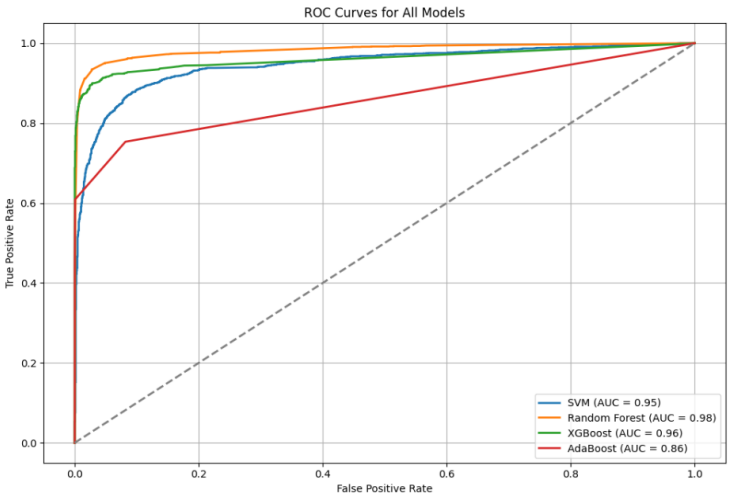

Fig 1.3: ROC Curve for all models

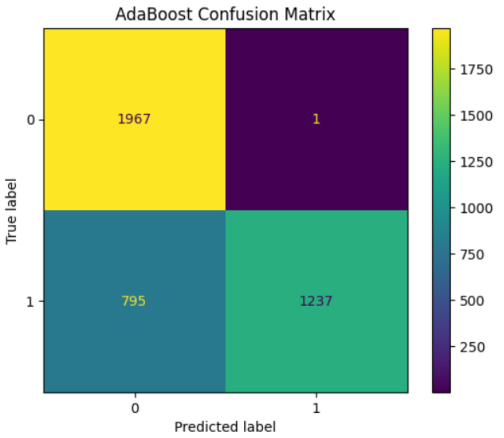

Fig1.4: XGBoost Confusion Matrix

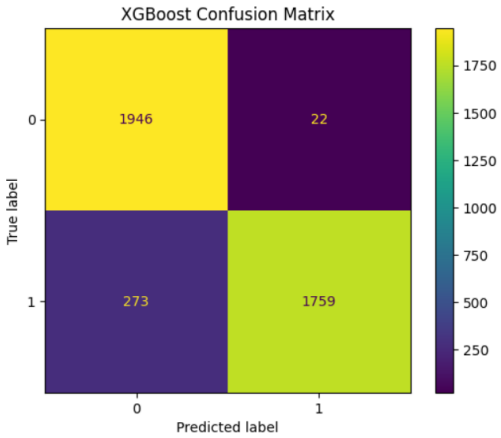

Fig 1.5: XGBoost Confusion Matrix

Best-Performing Model

The **LSTM** model achieved the highest accuracy and F1-score, outperforming the classical models. This indicates that the LSTM model was able to effectively capture the sequential nature of the text data and provide better classification results for both ADHD and Non-ADHD classes.

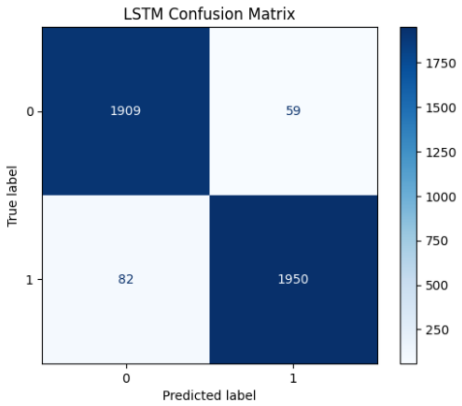

Fig 2.1: LSTM Confusion Matrix

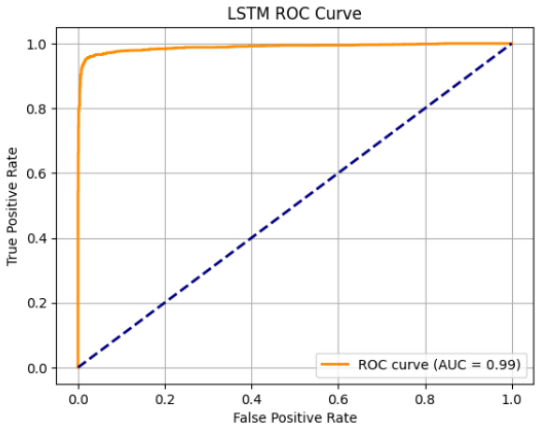

Fig 2.2: SVM Confusion Matrix

VI. DISCUSSION

This section analyzes the comparative performance of deep learning and classical models in classifying ADHD-related posts. Key insights into model behavior, linguistic patterns, and dataset challenges are discussed to better understand the outcomes and areas for improvement.

Interpretation of Results

- LSTM Performance: The LSTM model's strong performance can be attributed to its ability to capture long-term dependencies in the text, making it highly effective for text classification tasks. Its bidirectional nature allows it to understand both forward and backward context in sentences, enhancing its ability to distinguish between ADHD and non-ADHD posts.

- Classical Models: The classical models, while performing well, were slightly less effective than the LSTM model. Random Forest and XGBoost provided strong results but did not capture the sequence and context of the text in the same way as the LSTM. SVM and AdaBoost performed well in certain cases but struggled with class imbalance and noisy data.

Insights into Linguistic Features

- The linguistic features that contributed to the distinction between ADHD and Non-ADHD posts include:
    - ADHD-related vocabulary: Terms related to symptoms, medication, concentration difficulties, etc., were more prominent in ADHD posts.
    - User behavior: Posts about emotional struggles, mental health, and attention-related issues were more frequent in ADHD posts, whereas Non-ADHD posts were more general in nature.

Limitations of Current Model

- Overfitting: The LSTM model showed some signs of overfitting, as evidenced by the drop in validation accuracy in later epochs, even though it still achieved high overall performance. This overfitting likely occurred due to the following reasons based on the implementation:
    - Model Complexity: LSTM models are powerful but also very complex, capable of learning intricate patterns in the data. In this case, the LSTM may have been too complex for the dataset, leading it to memorize the training data rather than generalize to unseen data. The high number of parameters in the model may have caused it to fit the noise in the data, resulting in a drop in validation performance. Simplifying the model or reducing the number of layers/neurons could

mitigate this.

- Lack of Regularization Techniques: The implementation may not have employed sufficient regularization methods such as dropout, L2 regularization, or early stopping. Regularization techniques are essential to prevent the model from overfitting by ensuring it doesn't memorize the training data. Without these techniques, the LSTM model likely overfitted the training data, leading to high performance on training but poor generalization to validation data.

- Insufficient Training Data: The size of the dataset could be another factor contributing to overfitting. While the dataset was balanced, if it was not large enough, the LSTM model might have learned to memorize specific patterns from the limited examples, rather than generalizing well. In such cases, the model could perform well on the training set but fail to generalize to the validation set, leading to overfitting. Increasing the dataset size or augmenting the data could help improve generalization.

- Dataset Imbalance: Although the dataset was balanced, a slight imbalance in the nature of posts could affect model performance. For instance, Non-ADHD posts might include a broader variety of topics, leading to greater variability in classification.

- Reddit-Specific Language: The dataset contains language specific to Reddit, such as abbreviations, internet slang, and informal communication. This language may not generalize well to other platforms or real-world applications.

- Label Noise: Some of the labels may have been noisy, particularly in cases where posts self-reported ADHD without medical confirmation.

VII. FUTURE WORK

To enhance the accuracy and generalizability of ADHD classification models, several avenues can be explored in future research:

- Use of BERT and Transformer-based Models: Incorporating pre-trained transformer architectures like BERT can significantly improve performance by capturing deep contextual information from text. These models are capable of understanding nuanced linguistic patterns that traditional models may overlook.

- Incorporating Metadata: Integrating additional features such as post timestamps, user engagement metrics, or user history can provide richer context, potentially improving model predictions by capturing behavioral and temporal cues.

- Dataset Expansion: Expanding the dataset to include posts from various social media platforms beyond Reddit can help reduce domain bias and improve the model's ability to generalize across different writing styles and contexts.

- Ensemble Approaches: Future studies could explore ensemble methods that combine the strengths of classical machine learning and deep learning models. This hybrid strategy may yield more robust and accurate classification outcomes.

- Hyperparameter Tuning and Cross-Validation: Extensive tuning of model hyperparameters, coupled with k-fold cross-validation, can optimize model performance and mitigate overfitting, especially in complex models like LSTM or transformers.

VIII. CONCLUSION

This study explored the classification of ADHD-related posts on Reddit using both classical machine learning models and a deep learning approach, with Long Short-Term Memory (LSTM) networks emerging as the most effective. The superior performance of the LSTM model highlights the value of leveraging sequential modeling for nuanced text classification tasks in mental health contexts. While classical models such as Random Forest and XGBoost showed competitive results, they lacked the ability to capture the temporal and contextual intricacies of natural language that the LSTM model could. Despite this success, challenges such as overfitting, platform-specific language, and potential label noise underscored the need for further refinement.

Importantly, the findings emphasize the potential of automated systems in supporting early detection and screening of ADHD-related behaviors through social media analysis. Such approaches offer scalable, accessible tools that could complement traditional mental health diagnostics. With thoughtful advancements in model architecture, dataset diversity, and generalizability, this line of research could contribute meaningfully to digital mental health solutions and aid clinicians and researchers in identifying individuals in need of psychological support.

REFERENCES

1. Maniruzzaman, M., Shin, J., & Hasan, M. A. M. (2022). Predicting Children with ADHD Using Behavioral Activity: A Machine Learning Analysis. Applied Sciences, 12(5), 2737. https://doi.org/10.3390/app12052737

2. Chugh, N., Aggarwal, S., & Balyan, A. (2024). The Hybrid Deep Learning Model for Identification of Attention-Deficit/Hyperactivity Disorder Using EEG. Clinical EEG and neuroscience, 55(1), 22–33. https://doi.org/10.1177/15500594231193511

3. S. Parameswaran, S. R. Gowsheeba, E. Praveen and S. R. Vishnuram, "Prediction of Attention Deficit Hyperactivity Disorder Using Machine Learning Models," 2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT), Vellore, India, 2024, pp. 1-6, doi: 10.1109/AIIoT58432.2024.10574571.

4. Li, S., Nair, R., & Naqvi, S. M. (2024). Acoustic and Text Features Analysis for Adult ADHD Screening: A Data-Driven Approach Utilizing DIVA Interview. IEEE journal of translational engineering in health and medicine, 12, 359–370. https://doi.org/10.1109/JTEHM.2024.3369764

5. Nizar Alsharif, Mosleh Hmoud Al-Adhaileh and Saleh Nagi Alsubari et al. ADHD Diagnosis Using Text Features and Predictive Machine Learning and Deep Learning Algorithms. JDR. 2024. Vol. 3(7). DOI: 10.57197/JDR-2024-0082