Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 5 (2023)

Optimizing Machine Learning Models in Human Activity Recognition with the WISDM Dataset

[1]M.Sridevi,[2]Dr. Selvi.V

[1]Research Scholar, Department of Computer science,
Mother Teresa Women's University,
Kodaikanal.

[2]Assistant Professor, Department of Computer science,
Mother Teresa Women's University,
Kodaikanal.

Abstract:-Human Activity Recognition (HAR) is a technology with significant potential, utilizing data from various devices like smartphones and cameras. It finds applications in daily activities such as driving, cleaning, and gaming, involving fundamental movements like standing, sitting, jogging, and typing. Accurate identification of these actions is crucial for effective human-computer interaction systems. This work incorporates a HAR module to extract valuable insights from data signals. It employs Machine Learning (ML) models to automatically detect human activities using raw data from Internet of Things wearable sensors, including innovative combinations like Adagrad and ELU within the MLP algorithm. The ML models' performance is evaluated using statistical metrics such as accuracy, precision, recall, and f1-score, and comparisons are made with existing models.

Keywords*: Artificial Intelligence, AI, Teaching, Learning, Education.*

1. Introduction

The utilization of wearable sensors for physiological monitoring in the elderly has demonstrated significant potential in improving well-being and preventing adverse health events (S. Perez-Gamboa et al, 2022). This research focuses on addressing the challenges posed by irregular measurements in data from these sensors, with the aim of creating a robust HAR system using ML models and raw sensor data, without the need for pre- or post-processing. Recent advancements in machine learning algorithms have made their implementation in data analysis increasingly viable. While wearable sensors have made remote monitoring of individuals, even those with serious illnesses like Parkinson's disease, feasible, cameras have emerged as alternative HAR receivers, enabling the extraction of actions and movements from video sequences. However, the accuracy of HAR and state detection can be significantly affected when combining cameras with wearable sensors. In this context, video signals are not as desirable as those from wearables.

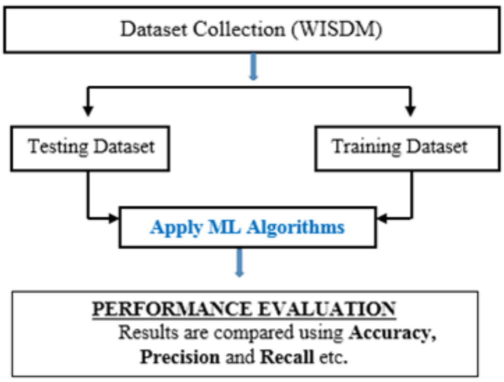

Figure 1: Outline for proposed model

Recent advancements in information technology have simplified the collection and preservation of routine medical data, facilitating medical decision-making and diagnosis. The study leverages the Activity and Biometrics Dataset from cell phones and wearables to simulate a wide range of human behaviors using various machine learning algorithms on a pre-trained dataset (X. Liu et al., 2016).The research challenge revolves around developing an AI model for HAR using IoT sensor data, addressing the detection of human activities and evaluating the effectiveness of machine learning techniques. HAR is a critical aspect of computer vision with applications in robotics, video games, human-robot interactions, therapy, sports, health monitoring, and video surveillance. Recognizing actions performed by individuals, particularly the elderly, based on raw data from IoT wearable sensors is the primary focus of this research.

Contribution: In the context of HAR using the WISDM dataset, the selection of optimizer-activation function combinations is pivotal for maximizing accuracy and generalization. While conventional combinations like Adam with ReLU or RMSprop with Leaky ReLU are reliable starting points, innovation can be introduced through adaptive optimizers tailored to time-series data, like Adam with GRU layers for temporal understanding. Moreover, the utilization of custom activation functions that capture unique patterns within WISDM's sensor data, beyond standard options, can enhance recognition performance. Additionally, incorporating transfer learning by pretraining on related datasets and fine-tuning with specific optimizers and activation functions may yield novel insights for HAR. This adaptability and domain-specific customization within deep learning models are instrumental in pushing the boundaries of accuracy and robustness when classifying human activities with the WISDM dataset.

2. Literature Review

In their study, Ahmed, Rafiq, and Islam (2020) devised a hybrid strategy that combines filter and wrapper methods to select the most relevant features for their Human Activity Recognition (HAR) system. This approach effectively identified optimal features, with the integration of both filter and wrapper methods. They applied Support Vector Machines (SVM) to HAR without feature selection, achieving an impressive average accuracy of 96%, surpassing the accuracy achieved without feature selection by 6%, which yielded a 94% accuracy rate.

In their 2022 study, K. Butchi Raju and his team utilized a comprehensive dataset sourced from various medical sensors, encompassing glucose level sensors, respiration rate sensors, temperature sensors, oxygen level sensors, EMG sensors, EEG sensors, and ECG sensors. Their research, titled "Smart Heart Disease Prediction System with IoT and Fog Computing Sectors Enabled by Cascaded Deep Learning Model," harnessed the GSO-CCNN (Genetic Search Optimization with Convolutional Cascaded Neural Networks) approach, resulting in an impressive success rate of 94%.

In 2020, KUN XIA et al. developed a human activity identification system, achieving an impressive accuracy of 95.85% using the WISDM dataset and the LSTM-CNN architecture. Their model's high identification accuracy is dependent on utilizing a comprehensive set of parameters.

Muhammad Farhan and colleagues (2021) introduced a novel KNN-SVM approach to identify human activities in the complex multi-process physical activities dataset from UCI. Their model involves the continuous training of two efficient classifiers, KNN and SVM, to differentiate among various human activities. These classifiers demonstrated efficiency levels ranging from 85% to 87%. By using majority decision in the classification process, their system achieved classification efficiency comparable to existing literature.

3. Methods

This section discusses the importance of using machine learning (ML) to create intelligent systems for recognizing human behaviors from IoT sensor data. ML models, primarily classification strategies, are explored for their role in improving prediction accuracy and detection efficiency. The section emphasizes the use of the WISDM sensor dataset for input and data preprocessing to enhance the categorization system's performance. ML techniques are then employed to classify activities, and their performance is compared to identify the most effective classifiers.

3.1. Machine learning for Multiclass Human Activity Classification

Machine learning (ML) is the study of how computers can learn from data, mainly focusing on algorithms and their implementation in learning. It involves gleaning new insights from a set of features, allowing expert systems to provide solutions based on historical data. Supervised machine learning, a common approach, treats activity categorization as a standard classification problem. This method uses labeled training data to create a mathematical link between features and class labels. Various supervised learning techniques are applied to categorize an IoT sensor dataset into different human activities, ensuring accurate classifications with systematic learning algorithms (Bhattacharya, D et al., 2022).

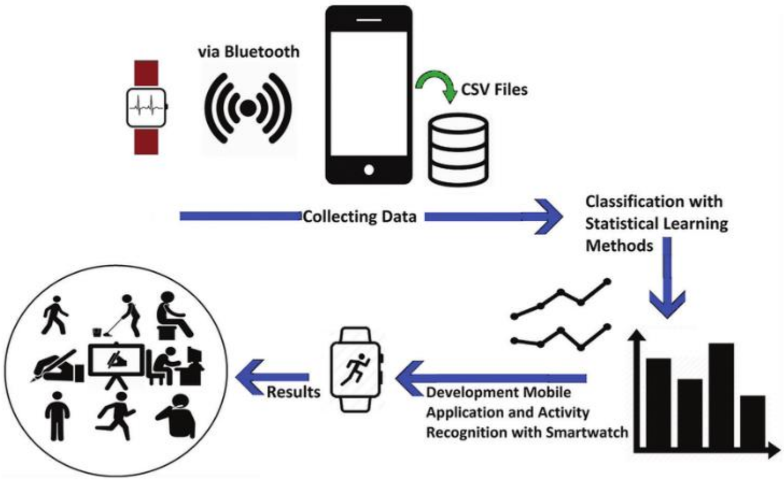

Figure 2: Functioning of supervised classification (L. Minh Dang et al., 2020)

This research utilizes different machine learning classification methods, showcasing their performance with various features.

i. Softmax Regression (SR)

In multiclass classification, the goal is to categorize instances into one of multiple activity classes (e.g., walking, jogging, sitting, and standing). Softmax Regression, also known as Multinomial Logistic Regression, is a suitable approach. The Softmax Regression model predicts the probability of an instance belonging to each class. For K classes, it computes K probabilities using a linear equation followed by the softmax function:

$$P(Y = k) = \frac{1}{1 + e^{(\beta_0 + \beta_1 . X_1 + \beta_2 . X_2 + \cdots + \beta_n . X_n)}} for\ k\ =\ 1,2, \dots . K$$

Where:

- $P(Y = k)$is the probability of the instance belonging to class
- X represents the input features from the WISDM dataset.
- $\beta_1, \beta_2 \dots \beta_n$are the coefficients associated with each class.

The Softmax function guarantees that probabilities add up to 1, making it ideal for multiclass problems. In training, the model fine-tunes the β coefficients to minimize a chosen loss function, such as Cross-Entropy Loss, utilizing labeled data from the WISDM dataset. The objective is to make precise predictions for the correct class of each instance (Bhattacharya, D et al., 2022).

ii. Random Forest (RF)

RF is an ensemble learning method for multiclass classification tasks with the WISDM dataset.

- ***Data Preparation:*** Features are extracted from the WISDM dataset to represent instances.

- ***Bootstrap Sampling:*** Random Forest creates multiple bootstrap samples (subsets) from the dataset with replacement.
- ***Decision Tree Construction:*** In each bootstrap sample, a decision tree is built, which divides the data into subsets by selecting the most informative features. This process can be illustrated by:

$$I(D,f) = {}_{i=1}^{c}p(i|t).I(p(i|t))$$

Where

- $I(D,f)$ is the impurity of dataset D after splitting it by feature f.
- c is the number of classes.
- $p(i|t)$ is the proportion of instances of class i at node t.
- $I(p(i|t))$ is the impurity of class i at node t.

Voting: Each decision tree predicts the class probabilities for the instance, and the final class is determined by majority voting across all trees. For multiclass classification, this can be represented as:

$$Y = arg\ max_{k}{}_{i=1}^{N}I(y_i\text{=k})$$

Where:

- Y is the predicted class label.
- k is each possible class label.
- N is the number of decision trees in the Random Forest.

The class label with the most votes from the decision trees is the final predicted class for the instance. This is a simplified representation of how Random Forest combines decision trees and majority voting to classify human activities using the WISDM dataset in a multiclass classification context (Bhattacharya, D et al., 2022).

iii. Support Vector Machine (SVM)

SVM, a classification algorithm, is applied for categorizing human activities into multiple classes within the WISDM dataset.

- ***Data Preparation:*** Features are extracted from the WISDM dataset to represent instances.
- ***Mathematical Representation:*** SVM aims to find the hyperplane that best separates the different classes. This hyperplane is represented as:

$$W \cdot X + b = 0$$

Where:

- W is the weight vector.
- X represents the input features from the WISDM dataset.
- b is the bias term.

a. Class Label Prediction: To predict the class label for a new instance:

$$Y = arg\ max_k(W_k \cdot X + \ b_k)$$

- Y is the predicted class label.
- W_k is the weight vector for class
- b_k is the bias term for class

b. **Maximal Margin:** SVM seeks to maximize the margin between the hyperplane and the nearest data points (support vectors). The margin can be calculated as:

$$Margin = \frac{2}{||W||}$$

Where

- $||W||$ is the Euclidean norm of the weight vector W.

SVM strives to optimize the margin while minimizing classification errors, resulting in the identification of the optimal hyperplane. SVM has the capability to work with nonlinear data by transforming it into a higher-dimensional space through the use of a kernel function. Common kernel functions encompass the polynomial kernel and the radial basis function (RBF) kernel. This is a simplified representation of how SVM works for multiclass human activity classification with the WISDM dataset. It optimizes the hyperplane to maximize the margin between classes, and the class label for a new instance is determined based on the side of the hyperplane it falls on. The use of kernel functions allows SVM to handle nonlinear data effectively(Jamil, H., et al, 2023).

iv. Multilayer Perceptron (MLP):

MLP is suitable for multiclass classification tasks, such as identifying human activities in the WISDM dataset(Kumar, P, et al , 2023).

1. *Data Preparation:* Features are extracted from the WISDM dataset to represent instances. Each instance is represented as a feature vector X.
2. *Mathematical Representation:* In an MLP, each layer consists of neurons or nodes. The computation in each neuron in a hidden or output layer can be represented as follows:

- For a hidden layer:

$$Z_j = {}_{i=1}^{n}X_i.W_{ij} + b_j$$
$$A_j = \sigma(Z_j)$$

- For the output layer in multiclass classification:

$$Z_k = {}_{j=1}^{m}A_i \, . \, V_{jk} + c$$

$$\widehat{Y_k} = \sigma(Z_k)$$

- Z_j is the weighted sum of inputs for neuron j in the hidden layer.
- A_j is the activation of neuron j in the hidden layer.
- n is the number of input features.
- W_{ij} is the weight connecting feature
- b_j is the bias for neuron j in the hidden layer.
- Z_k is the weighted sum of inputs for neuron k in the output layer.
- $\widehat{Y_k}$ is the predicted output (probability) for class k.
- m is the number of neurons in the hidden layer.
- V_{jk} is the weight connecting neuron j to class k in the output layer.
- c_k is the bias for class k in the output layer.
- $\sigma(Z)$represents the activation function (e.g., sigmoid or softmax) applied element-wise to Z.

1. *Activation Functions:* Common activation functions include the sigmoid function for the hidden layer and the softmax function for the output layer in multiclass classification.
2. *Loss Function:* To train the MLP, a loss function, such as the cross-entropy loss, is used to measure the difference between predicted class probabilities and actual class labels.

$$L\,(Y, \hat{Y}) = -\, {}_{k=1}^{K}Y_k.\,\log\,(\widehat{Y_k})$$

Where

- $L\,(Y, \hat{Y})$ is the loss.

- Y_k is the true class label (one-hot encoded) for class k.
- $\widehat{Y_k}$ is the predicted probability for class k.

3. *Training (Backpropagation):* The backpropagation algorithm serves to update the weights and biases of the neural network with the aim of minimizing the loss. This process entails the computation of gradients that reflect the loss concerning the network's parameters, followed by their adjustment through the utilization of gradient descent or alternative optimization methods.MLP is a versatile architecture for multiclass classification that can learn complex patterns in data. It uses feedforward propagation and backpropagation to make predictions and update model parameters during training. It is widely used in applications like human activity classification using datasets such as WISDM.

4. Results

This section discusses the real-world applications of the proposed model, covering software configuration, hardware setup, and the decision-making process. The comprehensive evaluation is performed on a computing device running macOS with specific hardware specifications, including a 64-bit architecture, a Radeon Pro GPU, 16GB RAM, and an Intel 8-core i7 processor. The environment used for executing Python 3.8 programs is Anaconda.

Table.1. Important libraries(S. Nooruddin, et al, 2022)

Libraries	**Version Details**
Keras	2.3.2
Numpy	1.21.7
Scikit-learn	0.21.4
Pandas	0.25.2
Joblib	0.13.3
Matplotlib	3.1.2
seaborn	0.9.0

4.1. Dataset Description

The dataset contains responses from 5,418 individuals who participated in six different three-minute tests for Human Activity Reconstruction (HAR). It was collected using a custom application designed to work on both smartphones and smartwatches running Android 6.0 and Android Wear 1.5, respectively, and involves various activities such as Walking, Jogging, Upstairs, Downstairs, Sitting, and Standing. This dataset is valuable for human activity recognition research (I. M. Nasir, et al, 2021).

4.2. Performance Metrics and Evaluation

In order to improve its capability of recognizing human behaviors, the proposed architecture makes use of algorithms for machine learning.

Table 2. The total number of data sets that were used in the training and testing procedure

S.No	Total Number of tweets	Training Data	Testing Data
01	5418	4334	1084

The proposed architecture incorporates a number of different machine learning algorithms, and the chaotic technique is used to tune the parameters of the hybrid system. The particulars of the tuned parameter were discussed in the section that came before this one. Additionally, the proposed architecture was validated by employing the dataset on which it was demonstrated that the proposed method successfully classified the appropriate categories. Using the metrics that are outlined in the table that follows, calculations are carried out in order to conduct an analysis of how well the proposed architecture performs.

Table.3. Performance Metrics

SL.NO	Performance Metrics	Mathematical Expression
01	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
02	Sensitivity or recall	$\frac{TP}{TP+FN}$ x100
03	Specificity	$\frac{TN}{TN+FP}$
04	Precision	$\frac{TN}{TP+FP}$
05	F1-Score	$2.\frac{Precison * Recall}{Precision + Recall}$

TP is True Positive Values, TN is True Negative Values, FP is False Positive and FN is False Negative values (S. Mekruksavanich et al, 2021)

```
TRAIN.head(5)
```

	0	1	2	3	4	5	6	7	8	9	...	554	555
0	0.288585	-0.020294	-0.132905	-0.995279	-0.983111	-0.913526	-0.995112	-0.983185	-0.923527	-0.934724	...	-0.112754	0.030400
1	0.278419	-0.016411	-0.123520	-0.998245	-0.975300	-0.960322	-0.998807	-0.974914	-0.957686	-0.943068	...	0.053477	-0.007435
2	0.279653	-0.019467	-0.113462	-0.995380	-0.967187	-0.978944	-0.996520	-0.963668	-0.977469	-0.938692	...	-0.118559	0.177899
3	0.279174	-0.026201	-0.123283	-0.996091	-0.983403	-0.990675	-0.997099	-0.982750	-0.989302	-0.938692	...	-0.036788	-0.012892
4	0.276629	-0.016570	-0.115362	-0.998139	-0.980817	-0.990482	-0.998321	-0.979672	-0.990441	-0.942469	...	0.123320	0.122542

5 rows × 564 columns

Figure 3: Dataset Sample

In data analysis, it's vital to ensure the dataset's quality and reliability through data cleaning. This process involves addressing missing values and eliminating duplicate entries. Missing data should be appropriately filled, removed, or interpolated, depending on the context. Duplicates can lead to biased results, so it's crucial to detect and remove them. Visual aids like flowcharts can help illustrate the steps involved in data cleaning, contributing to transparent and reproducible data analysis.

	0	1	2	3	4	5	6	7	8	9	...	554	555
0	0.257178	-0.023285	-0.014654	-0.938404	-0.920091	-0.667683	-0.952501	-0.925249	-0.674302	-0.894088	...	0.006462	0.162920
1	0.286027	-0.013163	-0.119083	-0.975415	-0.967458	-0.944958	-0.986799	-0.968401	-0.945823	-0.894088	...	-0.083495	0.017500
2	0.275485	-0.026050	-0.118152	-0.993819	-0.969926	-0.962748	-0.994403	-0.970735	-0.963483	-0.939260	...	-0.034956	0.202302
3	0.270298	-0.032614	-0.117520	-0.994743	-0.973268	-0.967091	-0.995274	-0.974471	-0.968897	-0.938610	...	-0.017067	0.154438
4	0.274833	-0.027848	-0.129527	-0.993852	-0.967445	-0.978295	-0.994111	-0.965953	-0.977346	-0.938610	...	-0.002223	-0.040046

5 rows × 564 columns

Figure 4:After Pre-Processing

Exploring the dataset: The exploratory data that we record from each subject can be of assistance to us when performing analysis on our dataset. For instance, we can investigate the frequency with which particular categories (subjects) appear in a dataset(Dang, L.M, et al, 2018).

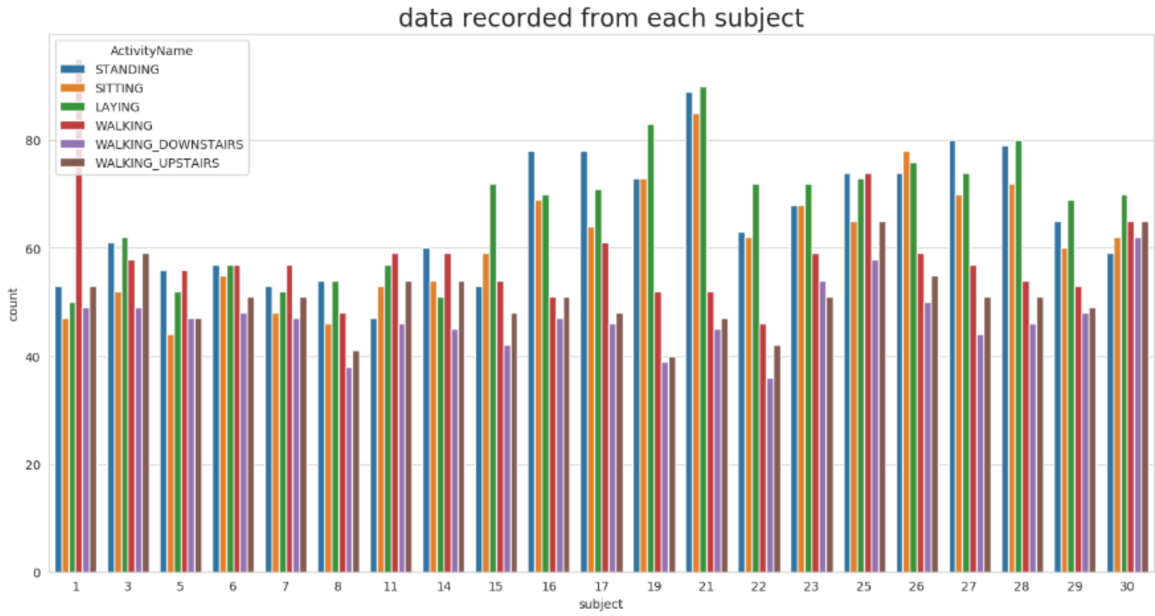

Figure 5:Number of occurrences on each classes in dataset

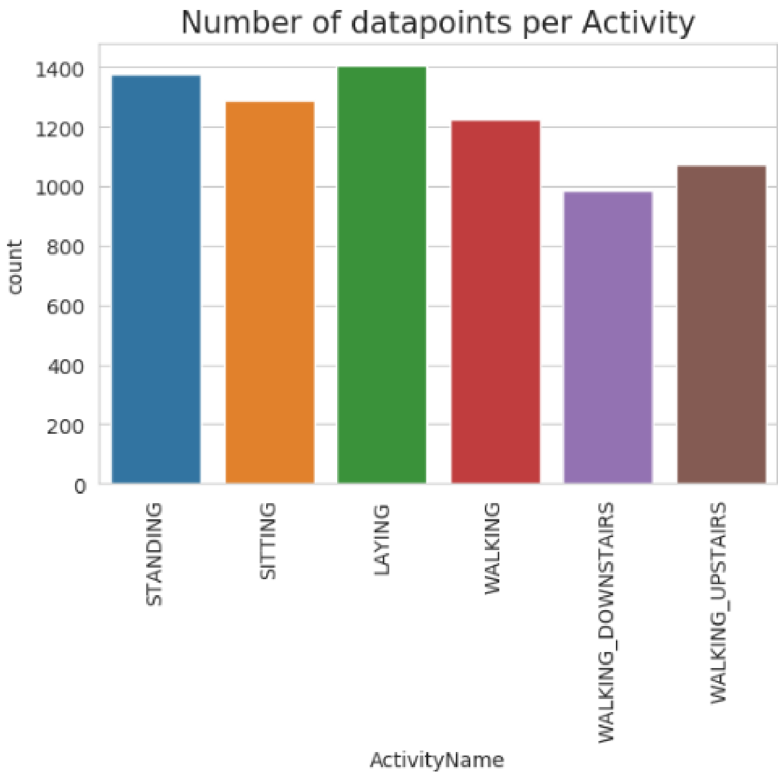

Figure 6: Balanced dataset

The figure above depicts a well-balanced dataset, where classes or categories have even distribution. This balance is essential in data analysis and machine learning to prevent biases and improve the accuracy of predictions. Balancing techniques may be applied to achieve this balance, enhancing the reliability of results.

Data Labeling: In this research, data training leverages the built-in functions available on various devices. For evaluation, twenty percent of the collected data is used. Following this, machine learning techniques are applied for data categorization.

Classification: The performance of these classification algorithms depends on the nature of the data, the quality of the features, and the choice of hyperparameters. Model evaluation is critical, and various metrics like accuracy, precision, recall and F1-Score.

Table.4. Performance analysis of Different ML Algorithms

Algorithm	Accuracy	Sensitivity	Specificity	Precision	F1-Score
SR	81.7	81.5	74.8	81.4	77.6
RF	83.2	78.3	75.1	75	83.2
SVM	86.3	85.4	81	88.1	85.5
MLP	92.4	92.3	93.5	92.5	92.8

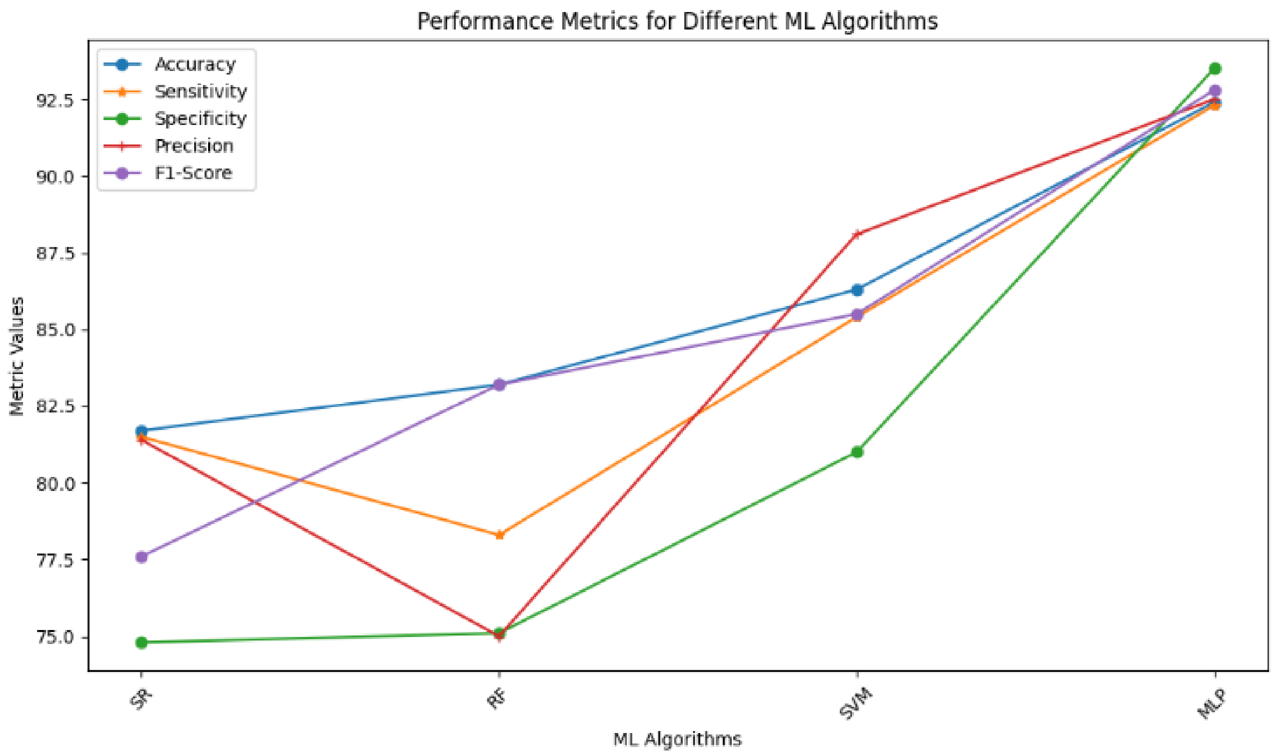

Figure 7: Performance analysis of Different ML Algorithms

The content includes a table and figure representing the performance analysis of various machine learning algorithms. The table summarizes key performance metrics such as accuracy, sensitivity, specificity, precision, and F1-Score for different algorithms labeled "SR," "RF," "SVM," and "MLP." The figure visually illustrates these metrics, aiding in the comparison of algorithm performance. It offers a quick and clear overview of each algorithm's effectiveness in the analysis. The "MLP" algorithm outperformed others with the highest accuracy (92.4%), sensitivity (92.3%), precision (92.5%), and F1-Score (92.8%). Meanwhile, "SVM" had the highest specificity at 81%.

Table.5. Performance Metrics for Different MLP Algorithms with various condition

Algorithm	Accuracy	Sensitivity	Specificity	Precision	F1-Score
Adam - ReLU	88.6	81.1	82.4	88	86.4
RMSprop - Leaky ReLU	80.9	78.1	75.1	83	73.7
SGD- Sigmoid	84.2	85.5	92.6	88	85.7
Adam - Tanh	89	92	89.2	92.4	92.2
Adagrad - ELU	95	95	93.2	93.4	94.2

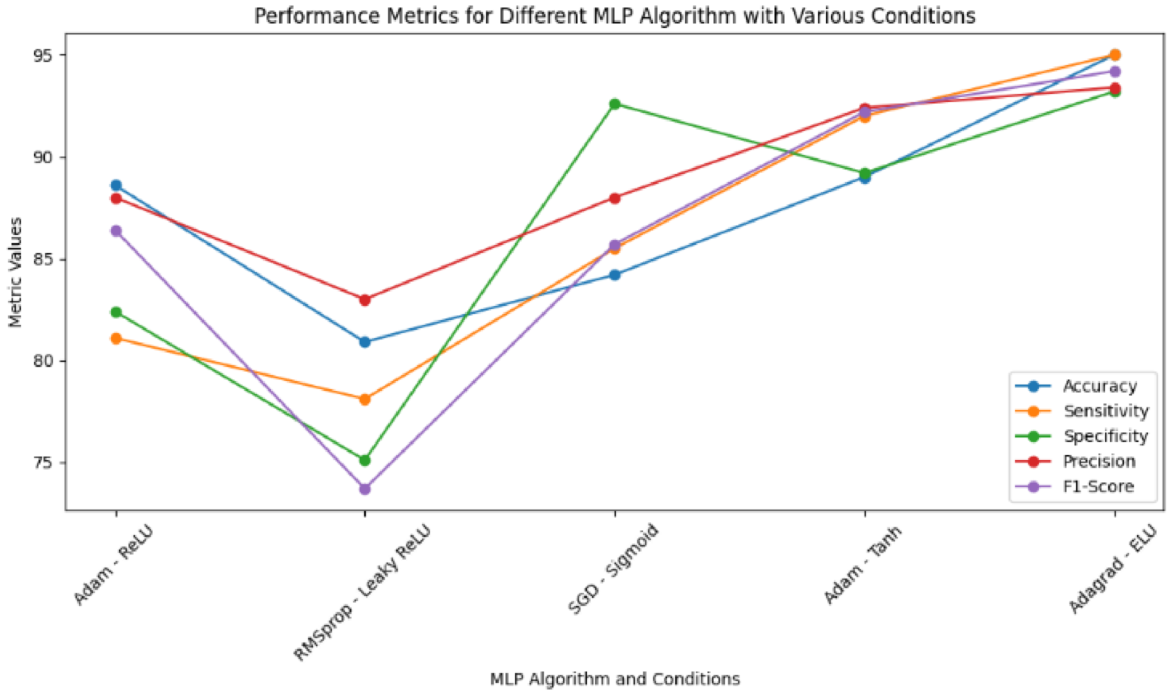

Fig.8. Performance analysis of MLP Algorithms with various condition

The above table and figure that together detail the performance metrics for MLP algorithms under various conditions. Notably, the "Adagrad - ELU" configuration demonstrates the highest performance, achieving an accuracy of 95% and strong metrics across the board. The figure visually represents these metrics, providing a quick comparative overview of how different MLP algorithms perform under varying conditions. This comprehensive presentation aids in the evaluation and selection of the most effective MLP algorithm for specific tasks and conditions.

Conclusion

In conclusion, this research employed deep learning techniques to analyze sentiments associated with COVID-19 vaccinations using Twitter data. The results revealed a balanced distribution of sentiments, with a significant proportion categorized as neutral. Among the deep learning models evaluated, the VAE-GANs model exhibited the highest accuracy of 92.59%, outperforming the LSTM, Bi-LSTM, and CNN models. The precision and recall values further supported the VAE-GANs model's effectiveness in accurately classifying neutral sentiments. This research contributes valuable insights into public opinions on COVID-19 vaccines and holds implications for policymakers and healthcare professionals. The conclusions can guide focused communication plans and address issues with vaccine uptake and reluctance. Future studies may explore the temporal dynamics of sentiments and investigate factors influencing public sentiments to enhance vaccination campaigns and public health interventions. The method was chosen because of its superior performance in dealing with the dataset's complexity and aligning with the research objectives, as demonstrated by comparative analysis and a literature assessment, making it the best choice. To enhance sentiment analysis for COVID-19 vaccine-related tweets, employ data augmentation, fine-tune hyperparameters, use ensemble learning, pre-train on COVID-19 data, address class imbalance, and consider multi-modal analysis along with external knowledge integration for a comprehensive approach.

Refrences

[1] S. Perez-Gamboa, Q. Sun and A. Ghasemkhani, "A Wireless Sensor Based Multi-layer Hybrid Deep Learning Model for Highly Correlated Human Activity Recognition," 2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS), Debrecen, Hungary, 2022, pp. 235-238.

[2] X. Liu, S. Agarwal, C. Ding and Q. Yu, "An LDA-SVM Active Learning Framework for Web Service Classification," 2016 IEEE International Conference on Web Services (ICWS), 2016, pp. 49-56.

[3] N. Ahmed, J.I. Rafiq, M.R. Islam Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model Sensors, 20 (1) (2020).

[4] K. Butchi Raju, Suresh Dara, Ankit Vidyarthi, V. MNSSVKR Gupta, Baseem Khan, "Smart Heart Disease Prediction System with IoT and Fog Computing Sectors Enabled by Cascaded Deep Learning Model", Computational Intelligence and Neuroscience, vol. 2022, Article ID 1070697, 22 pages, 2022.

[5] K. Xia, J. Huang and H. Wang, "LSTM-CNN Architecture for Human Activity Recognition," in IEEE Access, vol. 8, pp. 56855-56866, 2020, doi: 10.1109/ACCESS.2020.2982225.

[6] Javed, A.R., Saadia, A., Mughal, H. et al. Artificial Intelligence for Cognitive Health Assessment: State-of-the-Art, Open Challenges and Future Directions. Cogn Comput (2023).

[7] Bhattacharya, D.; Sharma, D.; Kim, W.; Ijaz, M.F.; Singh, P.K. Ensem-HAR: An Ensemble Deep Learning Model for Smartphone Sensor-Based Human Activity Recognition for Measurement of Elderly Health Monitoring. Biosensors 2022, 12, 393.

[8] Jamil, H., Ali, K.M. & Kim, DH. Federated recognition mechanism based on enhanced temporal-spatial learning using mobile edge sensors for firefighters. fire ecol 19, 44 (2023).

[9] Kumar, P., Suresh, S. Deep-HAR: an ensemble deep learning model for recognizing the simple, complex, and heterogeneous human activities. Multimed Tools Appl 82, 30435–30462 (2023).

[10] S. Nooruddin, M. M. Islam and F. Karray, "TinyHAR: Benchmarking Human Activity Recognition Systems in Resource Constrained Devices," 2022 IEEE 8th World Forum on Internet of Things (WF-IoT), Yokohama, Japan, 2022, pp. 1-8.

[11] I. M. Nasir, M. Raza, J. H. Shah, M. Attique Khan and A. Rehman, "Human Action Recognition using Machine Learning in Uncontrolled Environment," 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA), Riyadh, Saudi Arabia, 2021, pp. 182-187.

[12] S. Mekruksavanich and A. Jitpattanakul, "Recognition of Real-life Activities with Smartphone Sensors using Deep Learning Approaches," 2021 IEEE 12th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2021, pp. 243-246.

[13] Dang, L.M.; Hassan, S.I.; Im, S.; Lee, J.; Lee, S.; Moon, H. Deep Learning Based Computer Generated Face Identification Using Convolutional Neural Network. Appl. Sci. 2018, 8, 2610.