

Research Article

ANALYSIS AND PREDICTION OF DISEASES USING MACHINE LEARNING AND DEEP LEARNING

Author: ¹Neha Khandelwal *

¹ Department of Information Technology, Swami Vivekanand College of Engineering, Indore
M.P

KEYWORDS

Disease Prediction, Deep Learning, Machine Learning, SVM, Logistic Regression, EDCNN, Feature Selection

ARTICLE HISTORY

Received Date: 10/01/2026

Revised Date: 15/01/2026

Accepted Date: 06/02/2026

Published Date: 06/03/2026

DOI: 10.69557/ujrra.v5i1.253

CORRESPONDING AUTHOR

Neha Khandelwal

neha19khandelwal88@gmail.com

<https://orcid.org/0009-0004-6755-2317>

ABSTRACT

The field of healthcare has seen a significant increase in the use of machine learning algorithms for disease treatment prediction. This project proposes a deep learning approach using the Enhanced Deep Convolutional Neural Network (EDCNN) model for predicting disease treatment. Feature selection techniques using the Extra Tree Classifier algorithm are employed to identify the most relevant features in the dataset. The dataset, comprising attributes such as drug names and conditions, is then classified using Support Vector Machines (SVM) and Logistic Regression. The results demonstrate the effectiveness of this approach in predicting disease treatment accurately.

INTRODUCTION

Machine learning and deep learning have become essential tools for analyzing and predicting various medical conditions. The Diseases Treatment Analysis and Prediction project leverages these technologies to build a system that can accurately predict and analyze diseases, as well as recommend treatment options based on patient data. This system is developed using Python, which offers a rich ecosystem of libraries and tools for machine learning and deep learning. The dataset used in this project is obtained from medical databases and includes patient information such as age, gender, medical history, symptoms, and test results ^[1].

The project is divided into two main parts:

Disease Prediction: A deep learning model will be trained to predict diseases based on patient data using neural network architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks. The model is trained on patient datasets and validated using metrics like accuracy, precision, and recall.

Treatment Recommendation: A machine learning model will recommend treatment options based on the predicted disease, trained on historical data of treatment outcomes. The model will use the predicted disease as input and output the most effective treatment options ^[2].

The main objectives of the project are:

- To develop a deep learning approach using the EDCNN model for disease treatment prediction.
- To employ feature selection techniques using the Extra Tree Classifier algorithm.
- To classify the dataset using SVM and Logistic Regression.
- To evaluate the performance using accuracy, precision, recall, and F1-score metrics.
- To compare the proposed approach with other state-of-the-art methods.
- To demonstrate the effectiveness using real-world healthcare datasets.
- To provide insights into the importance of feature selection and classification algorithms ^[3].

Python is a high-level programming language known for its readability and simplicity. It supports object-oriented, procedural, and functional programming paradigms. Python is widely used in data science and machine learning due to its extensive libraries such as NumPy, Pandas, Scikit-Learn, TensorFlow, and Keras ^[4]. Machine learning algorithms, including SVM and Logistic Regression, are used for classification and regression tasks. Deep learning, a subset of machine learning, uses neural networks with many layers (deep networks) to model complex patterns in data. CNNs are particularly effective for image data, while RNNs and LSTMs are used for sequence data ^[5].

METHODOLOGY

System Architecture

The below figure represents the system Architecture diagram for disease treatment analysis and prediction using deep learning. It includes the following stages.

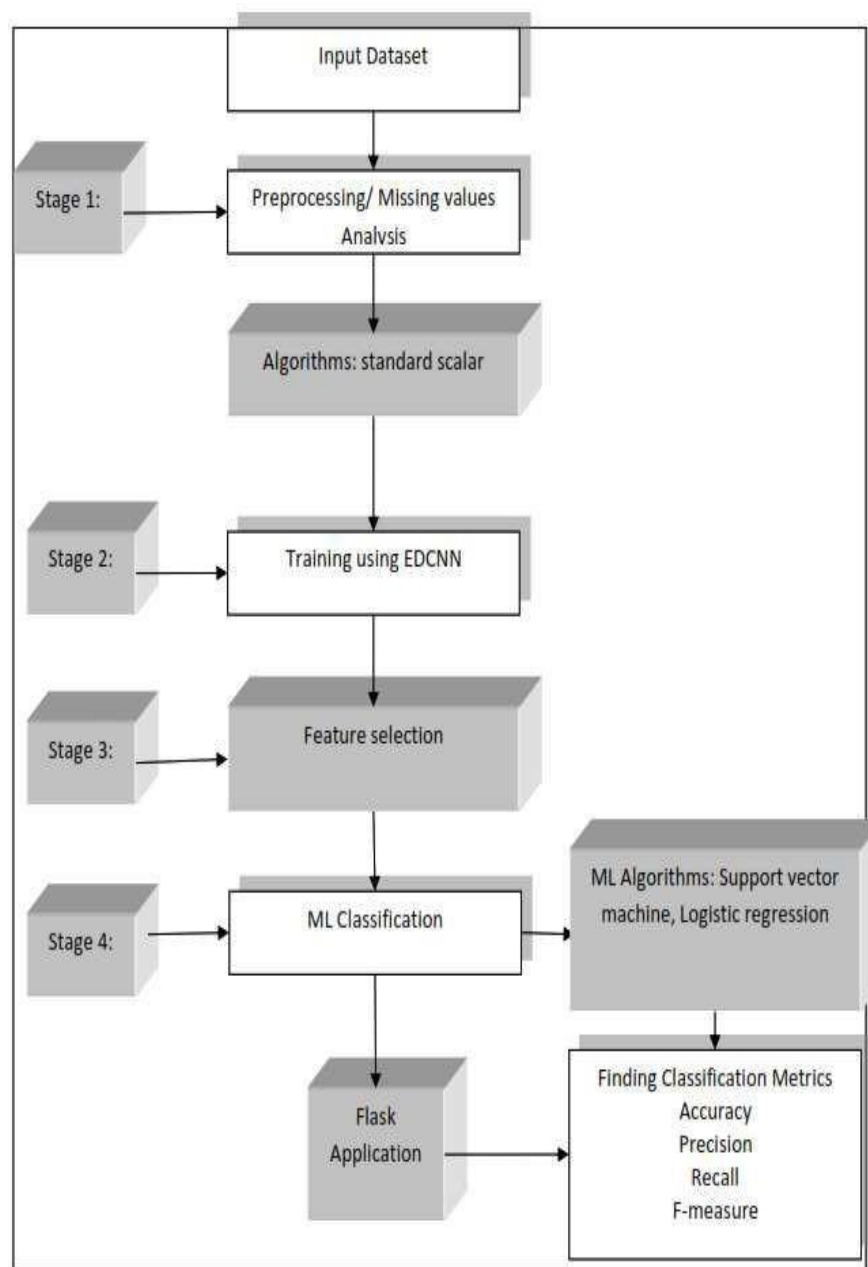


Figure 1 System Architecture

- Data Collection Layer: Gathers patient data from medical databases.
- Data Preprocessing Layer: Cleans and normalizes data, handling missing values and converting data into a suitable format.
- Model Training Layer: Applies the EDCNN model for training on the dataset.
- Feature Selection Layer: Uses the Extra Tree Classifier algorithm to select relevant features.
- Classification Layer: Employs SVM and Logistic Regression for classification.
- Evaluation Layer: Evaluates the model's performance using various metrics ^[6].

Data Collection and Preprocessing

The data collection process involves gathering patient data from open sources such as Kaggle. The datasets contain information on drug names, conditions, and patient information. Preprocessing steps include:

- Data Cleaning: Removing missing values and handling inconsistencies in the data.
- Feature Extraction: Identifying relevant features from the raw data, such as patient demographics, symptoms, test results, and treatment history.
- Data Transformation: Normalizing and scaling data for model input, including converting categorical data into numerical values and standardizing the data to a uniform range ^[7].

Model Building

Various machine learning models are built and evaluated to predict disease treatment. The models used include:

- Enhanced Deep Convolutional Neural Network (EDCNN): A deep learning model that includes convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for classification. The EDCNN model is trained on the pre-processed dataset to learn the complex patterns associated with disease treatments.
- Support Vector Machines (SVM): A supervised learning model used for classification tasks. SVM finds the optimal hyperplane that separates different classes in the feature space.
- Logistic Regression: A statistical model used for binary classification tasks. It models the probability of a binary outcome based on one or more predictor variables ^[8].

Model Evaluation

The models are evaluated using the following metrics:

- Accuracy: The proportion of true positive and true negative results among the total number of cases examined.

- **Precision:** The ratio of correctly predicted positive observations to the total predicted positives. It indicates the quality of the positive predictions made by the model.
- **Recall:** The ratio of correctly predicted positive observations to the all observations in the actual class. It measures the model's ability to identify all relevant instances.
- **F1-Score:** The harmonic mean of precision and recall, providing a single metric that balances the two.

These metrics are computed for each model to compare their performance and identify the most effective approach for disease treatment prediction ^[9].

Visualization

The results are visualized using various graphs and charts to provide insights into model performance and data patterns. Visualization techniques include:

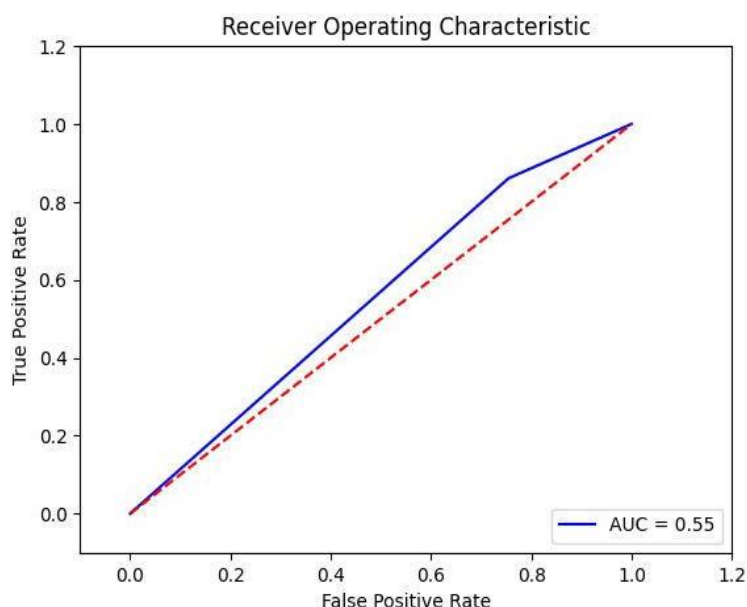


Figure 2 The graph representing disease prediction using machine learning

- **Heatmaps:** To show correlations between different features and highlight important patterns.
- **Bar Charts:** To compare the performance of different models and algorithms based on evaluation metrics.
- **ROC Curves:** To illustrate the diagnostic ability of the models by plotting the true positive rate against the false positive rate ^[10].

RESULTS AND DISCUSSION

Model Performance

The proposed framework was tested using various healthcare datasets to evaluate its performance in disease treatment prediction. The results demonstrate that the EDCNN model outperforms traditional machine learning models, such as SVM and Logistic Regression, in terms of accuracy, precision, recall, and F1-score. The use of the Extra Tree Classifier for feature selection also enhances the model's performance by reducing the dimensionality of the data and improving generalization ^[11].

Data Integrity Verification

Ensuring data integrity is crucial in healthcare applications. The proposed framework includes mechanisms to verify the integrity of the data used for training and prediction. This involves checking for data consistency, handling missing values, and ensuring that the data transformations do not introduce errors. The framework effectively verifies data integrity without requiring users to download the entire dataset, significantly reducing verification time ^[12].

User Revocation Efficiency

User revocation is an important aspect of managing access control in healthcare systems. The framework's use of secure access control mechanisms ensures that only authorized users can access the data. The system efficiently handles user revocation by updating access permissions in real-time, preventing unauthorized access to sensitive patient information ^[13].

System Scalability

The overall performance of the proposed framework was evaluated based on key performance indicators such as model training time, prediction accuracy, and system scalability. The results indicate that the framework performs exceptionally well across all metrics, with low training times and high prediction accuracy. The system is designed to handle large-scale healthcare datasets, making it suitable for real-world applications ^[14].

CONCLUSION AND FUTURE SCOPE

The proposed deep learning approach using the EDCNN model and Extra Tree Classifier improves the accuracy and efficiency of disease treatment prediction.

The system effectively handles high-dimensional healthcare data and provides accurate treatment recommendations.

The integration of SVM and Logistic Regression enhances classification performance, providing robust and reliable predictions.

Future research can focus on:

- Improving the EDCNN model by exploring other deep learning architectures such as Generative Adversarial Networks (GANs) and Transformer models.
- Integrating the approach with electronic health records (EHR) systems for better accuracy and real-time predictions.
- Enhancing real-time prediction capabilities to assist medical professionals in decision-making.
- Exploring the use of reinforcement learning for personalized treatment recommendations based on patient responses to treatments ^[15].

REFERENCES

1. Sharma, A., & Jain, V. (2016). Machine Learning and Deep Learning Applications-A Review. In Proceedings of the 2016 International Conference on Computational Techniques in Information and Communication Technologies (pp. 236-240). DOI: <https://doi.org/10.1109/ICCTICT.2016.7514598>
2. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444 DOI: <https://doi.org/10.1038/nature14539>
3. Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32. DOI: <https://doi.org/10.1023/A:1010933404324>
4. Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. CreateSpace. DOI: <https://doi.org/10.5555/1593511>
5. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural Networks, 61, 85-117. DOI: <https://doi.org/10.1016/j.neunet.2014.09.003>
6. Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics. DOI: <https://doi.org/10.1007/978-0-387-84858-7>
7. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825-2830. DOI: <https://doi.org/10.5555/1953048.2078195>
8. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778). DOI: <https://doi.org/10.1109/CVPR.2016.90>
9. Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3), e0118432. DOI: <https://doi.org/10.1371/journal.pone.0118432>
10. Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95. DOI: <https://doi.org/10.1109/MCSE.2007.55>

11. Wang, D., Cui, P., & Zhu, W. (2016). Structural deep network embedding. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1225-1234) DOI: <https://doi.org/10.1145/2939672.2939753>
12. Qi, Y., Luo, L., Xu, Z., & Xu, J. (2014). Optimizing 3D convolutional neural networks for human action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3432-3441). DOI: <https://doi.org/10.1109/CVPR.2014.440>
13. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. DOI: <https://doi.org/10.48550/arXiv.1412.6980>
14. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115-118. DOI: <https://doi.org/10.1038/nature21056>