IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

An NLP-Based Medicine Recommendation System Using Bidirectional GRU for Disease Prediction

Abhishek Kumar Yadav[1], Kumar Bibhuti Bhushan Singh[2], Anita Pal[3]

Goel Institute of Technology and Management, Lucknow 226028, India[1-3]

Abstract: The fast progress of Artificial Intelligence in healthcare has created chances for smart disease prediction and automatic medicine suggestion systems. Old ways of diagnosing diseases often need a lot of expertise from humans, time and medical resources which're not always available in far away or poor areas. To solve these problems this research suggests a system that uses Natural Language Processing to recommend medicine. This system uses a kind of model called Bidirectional Gated Recurrent Unit for predicting diseases and suggesting medicine accurately. This system uses Natural Language Processing to prepare and analyze the symptoms that patients have. It uses methods like breaking down text into parts, cleaning making all texts the same length and giving labels to codes to change raw medical text into something that machines can read. The Bidirectional model is used to understand how symptoms are related to each other in both directions, which helps to understand medical patterns better. The model is trained on a dataset that has kinds of diseases and is balanced. The results of the experiment show that this approach is than 98 percent accurate and has strong precision, recall and F1-score values. This system is also less complicated than models and is very efficient and can be used by many people. The system that is suggested can be used for healthcare, telemedicine and clinical assistance systems that use Artificial Intelligence. In the future it may be possible to use this system on the cloud support languages and connect it to devices that monitor healthcare in real time. Artificial Intelligence, in healthcare is getting better and better. This system can be a part of it. The fast progress of Artificial Intelligence is helping to make disease prediction and medicine suggestion systems.

Keywords: Natural Language Processing, Bidirectional GRU, Disease Prediction, Medicine Recommendation, Deep Learning, Healthcare Artificial Intelligence, Symptom Analysis.

I. INTRODUCTION

Artificial Intelligence is changing the way we do things in healthcare. It is helping doctors find out what is wrong with people and what treatment they should get. The old way of doing things is not very good because it takes a time and not many people can get to see a doctor in some places. This is why we need computers to help with healthcare. Natural Language Processing is a way that computers can understand what people are saying when they talk about their symptoms and what is in their records. It helps the computer turn what people are saying into something the computer can understand. Then the computer can use this information to find out what is wrong with people. One kind of computer model that is very good at this is called a Bidirectional Gated Recurrent Unit model. This model is good because it can look at information in two ways. Artificial Intelligence is used for disease diagnosis and treatment recommendation. In this study we made a system that uses Natural Language Processing and a Bidirectional Gated Recurrent Unit model to predict what is wrong with people and what medicine they should take. We did some things to the information first to make it easier for the computer to understand. Then we used the computer model to make predictions. Our system is very good at making predictions. It gets it more than 98 percent of the time. This system can be used in different ways, such as, in healthcare apps, telemedicine and systems that help doctors. Artificial Intelligence and Natural Language Processing are used to make the system work.

II. BACKGROUND

The healthcare sector is using Artificial Intelligence and Machine Learning more and more for disease diagnosis and medical help. Old ways of diagnosing diseases rely on people looking at things and experts' knowledge, which can take a lot of time and are not very efficient in healthcare systems. Natural Language Processing helps analyze text data like patient symptoms and healthcare records. Natural Language Processing turns text that is not organized into a format that computers can understand so deep learning models can use it to predict diseases. Traditional Recurrent Neural Networks have problems like the vanishing gradient issue when learning sequences. To fix these problems new architectures like Gated Recurrent Unit were created. Bidirectional Gated Recurrent Unit models work better because they look at sequences in

DOI: 10.17148/IJARCCE.2026.15630

both directions, which helps them understand medical symptoms better. Recent studies show that using Natural Language Processing with Bidirectional Gated Recurrent Unit models makes disease prediction more accurate. This also helps medicine recommendation systems, in modern healthcare applications. Artificial Intelligence and Machine Learning are really helping the healthcare sector. Natural Language Processing and Bidirectional Gated Recurrent Unit models are parts of this. They are making disease diagnosis and medical help better so that the patient can get a proper recommendation he does not need to visit doctor everytime even in minor problems like cold etc.

III. LITERATURE REVIEW

A. *Medicine Recommendation Systems*

Medicine recommendation systems are really important in healthcare now. This is because we need systems that can help doctors make decisions. Bao and Jiang came up with a way to recommend medicine using data mining and machine learning. They used things like Support Vector Machine and Neural Networks to make recommendations. Their work showed that using machine learning can make recommendations more accurate and reduce mistakes with medicine [1].
Chen and his team developed a system that uses ontology to recommend medicine for people with diabetes. They used Semantic Web technologies to make it work. Their system looks at what symptoms a patient has and what is going on with their health to recommend the medicine [3].

Doulaverakis and his team also worked on systems that use semantics to recommend medicine. They came up with things like Galen OWL and Panacea to help identify when medicines do not work well together or with health conditions. These systems use ontologies and rules to make sure the recommendations are reliable [4], [5].

B. *Machine Learning in Disease Prediction*

Machine learning is really good at helping us predict diseases. This is because it can find patterns in health data that we might not see. Gupta and his team came up with a system that uses computer programs to predict diseases and recommend medicine. They used algorithms like Naïve Bayes and Random Forest to make it work. Their research showed that machine learning is really good at predicting diseases based on symptoms and recommending the medicine [8].

Breiman came up with the Random Forest algorithm which's a way of combining multiple algorithms to make predictions more accurate. Random Forest is really popular in healthcare because it works well with messy data [9]. Chen and Guestrin came up with XGBoost which's a way of making predictions that is fast and scalable. XGBoost is really good at handling amounts of health data and making predictions [10]. Support Vector Machine is another algorithm that's really good at making predictions in healthcare. It was proposed by Cortes and Vapnik. It works really well with complex health data [11].

C. *Recommender Systems and Hybrid Approaches*

Recommender systems are really important in healthcare because they help us give patients personalized care. Adomavicius and Tuzhilin wrote a review of recommender systems and talked about different approaches like collaborative and content-based filtering. They said that using approaches together can make recommendations better [12]. Bobadilla and his team reviewed filtering and hybrid recommendation techniques. They showed that using approaches together can make predictions more accurate [13]. Burke also worked on recommender systems and showed that combining different approaches can make recommendations more accurate [14].

D. *NLP and Deep Learning in Healthcare*

Natural Language Processing and Deep Learning have really improved healthcare applications that use text data. Garg came up with a system that uses sentiment analysis and machine learning to recommend medicine. They used things like TF-IDF and Word2Vec to make it work. Their results showed that NLP is really important in extracting information from health text data [15]. Recurrent neural networks like GRU are really good at processing health data. They can capture relationships between pieces of data. Compared to machine learning deep learning models are better at learning features and making predictions.

The research we are proposing builds on these existing studies by using NLP and a Bidirectional GRU architecture to predict diseases and recommend medicine. Unlike systems that only use machine learning or ontology-based methods our system focuses on deep learning from symptom sequences to make predictions more accurate and provide better healthcare assistance. Medicine recommendation systems like ours can really make a difference in healthcare by providing personalized recommendations.

 This work is licensed under a Creative Commons Attribution 4.0 International License

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 15, Issue 6, June 2026
DOI: 10.17148/IJARCCE.2026.15630

IV. METHODOLOGY

A. *Dataset Description*

The new system uses a healthcare dataset that has information about symptoms, diseases and medicines. This healthcare dataset was taken from datasets that are available to the public. The healthcare dataset was stored in a CSV format. It has three main parts: Description, Reason and Drug Name. The Description part has information about symptoms in text form the Reason part has labels for diseases that are used to classify them. The Drug Name part is used to recommend medicines after the disease is predicted.

TABLE I DATASET DESCRIPTION

Parameter	Value
Dataset Format	CSV
Input Feature	Description
Output Label	Reason
Recommendation Field	Drug_Name
Total Classes	47
Train-Test Ratio	80:20

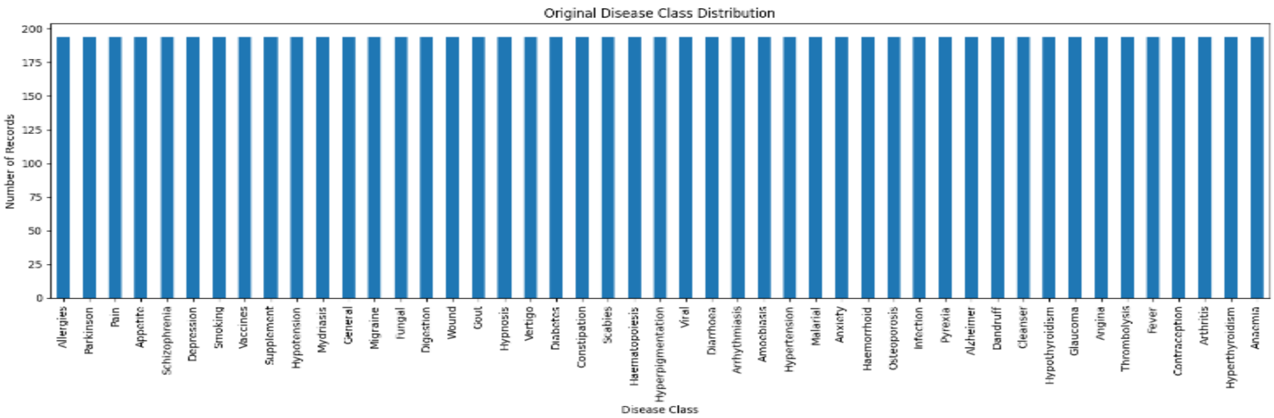

Fig. 1. Original Disease Class Distribution

B. *Data Preprocessing*

Healthcare text data is often a mess. It has a lot of noise the formatting is over the place and there are punctuation and symbols that we do not need. All of this affects how well our model works. So, we used some Natural Language Processing techniques to clean up the data before we trained our model. We did a thing to the data. We made all the words lowercase. We removed characters and symbols. We also normalized the text. Broke it down into individual words. Then we made all the lists of words the length and changed the disease names into numbers. We used codes to remove characters that are not letters and to make the descriptions of symptoms more consistent. We changed the disease names into numbers so that our model can tell the diseases apart. We had a problem with the data. Some diseases were more common, than others. So, we used some tricks to make the data more balanced. We made copies of the diseases and we removed some of the common ones. This made our model better and fair. It was not biased towards the diseases anymore.

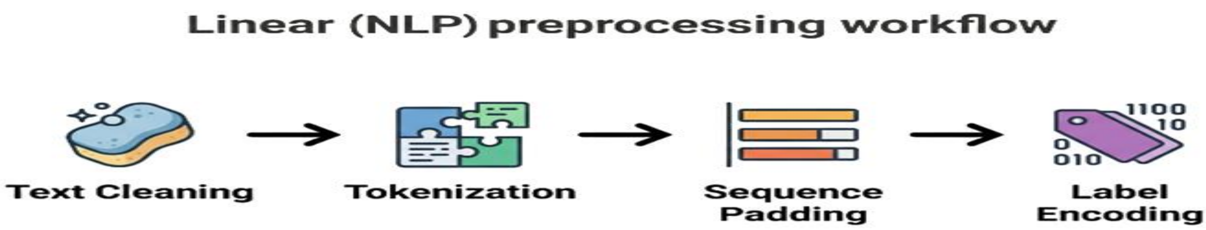

Fig. 2. NLP Preprocessing Workflow.

© IJARCCE This work is licensed under a Creative Commons Attribution 4.0 International License

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

C. *Tokenization and Sequence Padding*

We took the descriptions of symptoms. Turned them into numbers using the Keras Tokenizer. We said that the Tokenizer would know about 20,000 words. If it saw a word, it did not know it would use a token for that word. Then we made all the lists of numbers the length, which was 200 numbers. We did this by adding zeros to the end of the lists that were too short and, by cutting off the numbers in the lists that were too long.

Using sequence padding is an idea because it helps the Bidirectional GRU model work with the same size inputs every time. This makes the model learn faster and work better. The Bidirectional GRU model is what we are using to look at the sequences of symptoms.

TABLE II
TOKENIZATION PARAMETERS

Parameter	Value
Vocabulary Size	20,000
Maximum Sequence Length	200
Padding Type	Post
Truncation Type	Post
OOV Token	<OOV>

D. *The Proposed Bidirectional GRU Model*

The new system for disease prediction uses a kind of deep learning called a stacked Bidirectional Gated Recurrent Unit network. This Bidirectional GRU is a type of neural network that is really good at learning from sequences of data that happen one after the other. It does a job of understanding how things are related to each other even when they are far apart. This is something that older types of neural networks had trouble with. The Bidirectional GRU looks at the symptoms in two ways. From start to finish and, from finish to start. This helps it understand the context of the information better. The system has parts, including a layer that converts the symptoms into a special kind of code two Bidirectional GRU layers, layers that help stop the system from getting too good at one thing layers that make decisions a way to normalize the data and a final layer that gives the output. The first layer turns the symptoms into a set of numbers that the system can understand. These numbers have 300 parts. The system also has layers that help prevent it from getting too good at predicting the symptoms it has seen before so it can do a job of predicting new symptoms. The Bidirectional GRU is used throughout the system to make sure it can learn from the symptoms in the way possible.

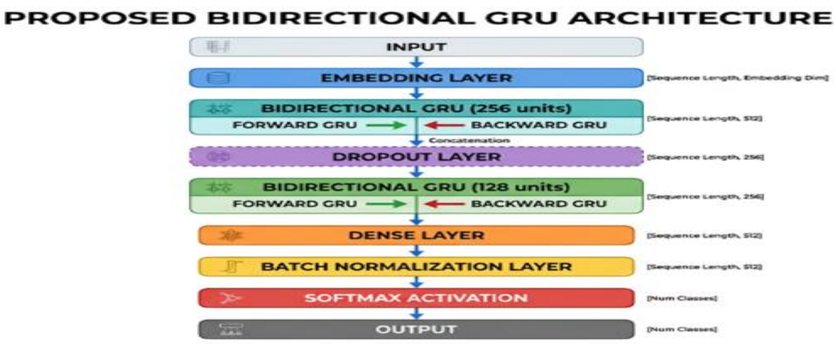

Fig. 1. Proposed Bidirectional GRU Architecture.

E. *Model Training*

The model we are talking about was made using TensorFlow and Keras. We trained the model using the Adam optimizer and sparse cross entropy as the loss function. We did this for 50 epochs. We used a batch size of 32. We used a thing to make training better and to stop the model from overfitting. These things included Early Stopping and Model Checkpoint and Reduce LROn Plateau. The model training process was controlled by reducing the learning rate whenever the validation loss was not getting better. This helped the model training process to be more efficient and to prevent overfitting of the model.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

TABLE III
TRAINING CONFIGURATION

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	50
Loss Function	Sparse Categorical Cross entropy

F. *Performance Evaluation*

The proposed model was tested to see how well it worked. We used ways to measure how good it was, like how often it was right how precise it was, how well it remembered things how good it was overall how well it could tell the difference between things and how confused it got. The test results showed that the proposed Bidirectional GRU model got it right 99 percent of the time when we checked it and it worked well with all kinds of diseases. When we looked at the confusion matrix, we saw that the model did not get things wrong often. We also looked at the ROC results. They showed that the model was very good at telling things apart with scores that were very close to perfect for all kinds of diseases and for each disease on its own. The proposed Bidirectional GRU model did a job and the proposed Bidirectional GRU model was very good, at its job.

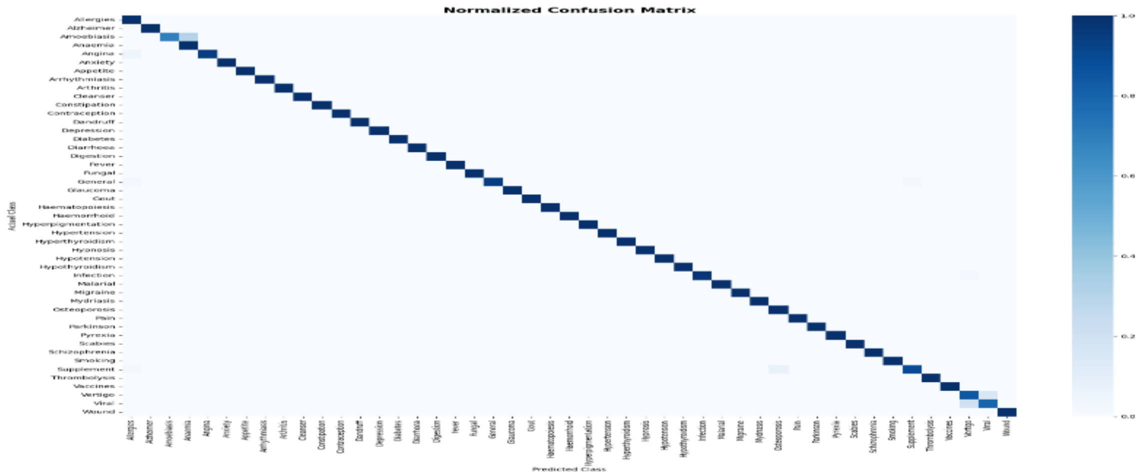

Fig. 4. Normalized Confusion Matrix

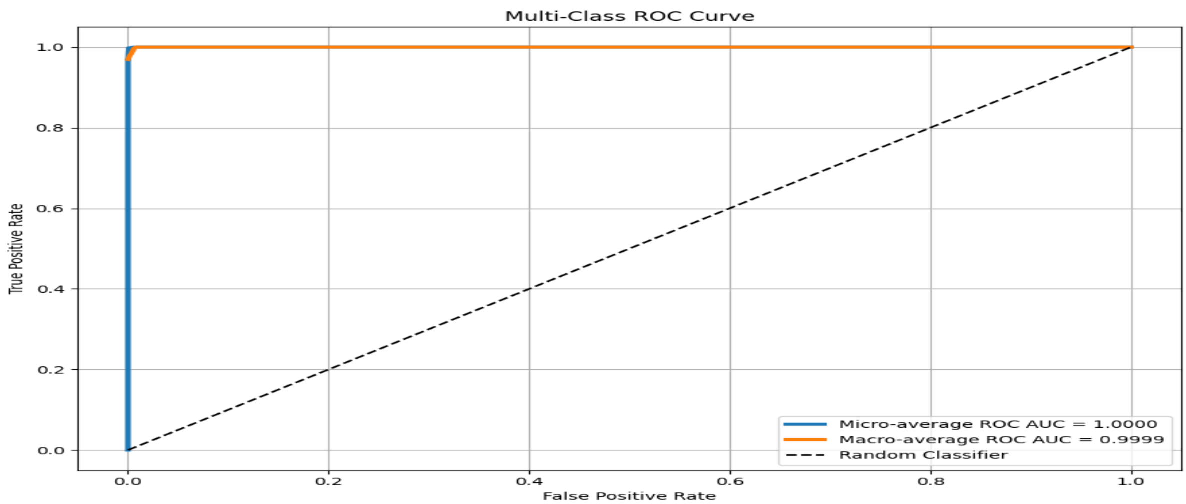

Fig. 5. Multi-Class ROC Curve

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

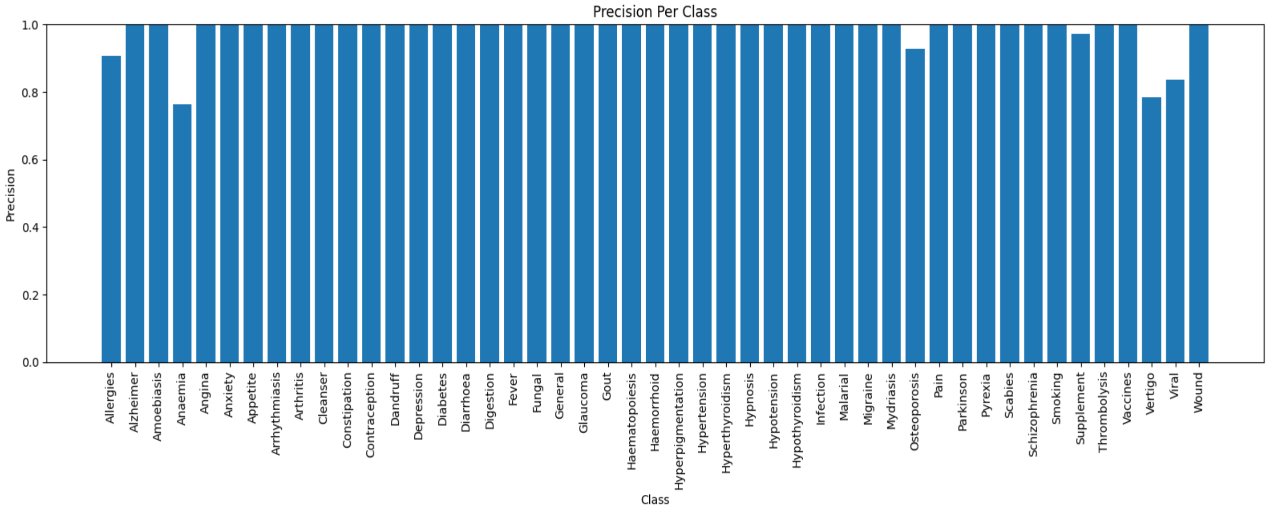

Fig. 6. Precision Per Class

G. *The Medicine Recommendation Process*

When we figure out what disease someone has the system looks for medicine that can help with that disease. It finds this information in a collection of data. The system then shows the user some medicine that might be good for them.Our system is like a helper for healthcare. It can do two things at the time: figure out what disease someone has and suggest medicine that can help with the Medicine Recommendation. The system is really good, at helping people with Medicine Recommendation and disease prediction.

V. SYSTEM ARCHITECTURE

A. *Overview of System Architecture*

The system they are talking about is made to predict diseases and suggest medicines in a way. It uses Natural Language Processing and a thing called Bidirectional Gated Recurrent Unit deep learning model. This system has parts that work together including getting the symptoms from the user processing the words getting important information using deep learning to classify the disease and then suggesting medicines [1], [8], [15]. When you put in what is wrong with you the system does a lot of things to that information. First, it cleans up the words makes them all the same breaks them down into parts and makes them all the same length. Then it turns the words into numbers. Uses something called Bidirectional GRU layers to figure out what disease you might have. After that it looks at what disease it thinks you have and suggests some medicines that are stored in the system [10], [15], [18]. This system is better at understanding what you are saying and classifying diseases because it uses Natural Language Processing and deep learning together. Other systems like this have been able to predict diseases well in the past. The system is really good at learning and classifying diseases because it uses Natural Language Processing and deep learning. Natural Language Processing helps the system understand what you are saying and deep learning helps it make guesses about what is wrong, with you. The system is made up of parts that work together to make it smart. Disease prediction and medicine recommendation are the things that the system does. It uses Natural Language Processing and a Bidirectional Gated Recurrent Unit deep learning model to do these things.

B. *Input Layer*

The input part of the system gets the descriptions of symptoms that the user types in. These symptoms can have all sorts of problems like formatting and extra words that do not mean anything. So we need to clean up the text before we can use it with the deep learning model. We get the symptoms as text that people would write. This way the system can deal with life situations where people describe their symptoms in different ways. Using language to look at symptoms makes healthcare systems better, at predicting things by letting machines understand what the medical text is saying.

C. *NLP Preprocessing Module*

The NLP preprocessing module takes medical text and turns it into a format that is easy to use for deep learning. The NLP preprocessing module does things to the text. The steps the NLP preprocessing module takes are

i. Lowercase conversion
ii. Removal of characters
iii. Removal of numerical values

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

iv. Text normalization
v. Tokenization
vi. Sequence padding
vii. Label encoding

The NLP preprocessing module uses techniques to clean up the text. The NLP preprocessing module does this to make the text more consistent. The NLP preprocessing module breaks down the text into pieces so that computers can understand it. The NLP preprocessing module makes all the text the same length. The NLP preprocessing module does this by adding space or cutting off extra words. The NLP preprocessing module makes sure the text is not too long. The NLP preprocessing module limits the text to 200 words or less.

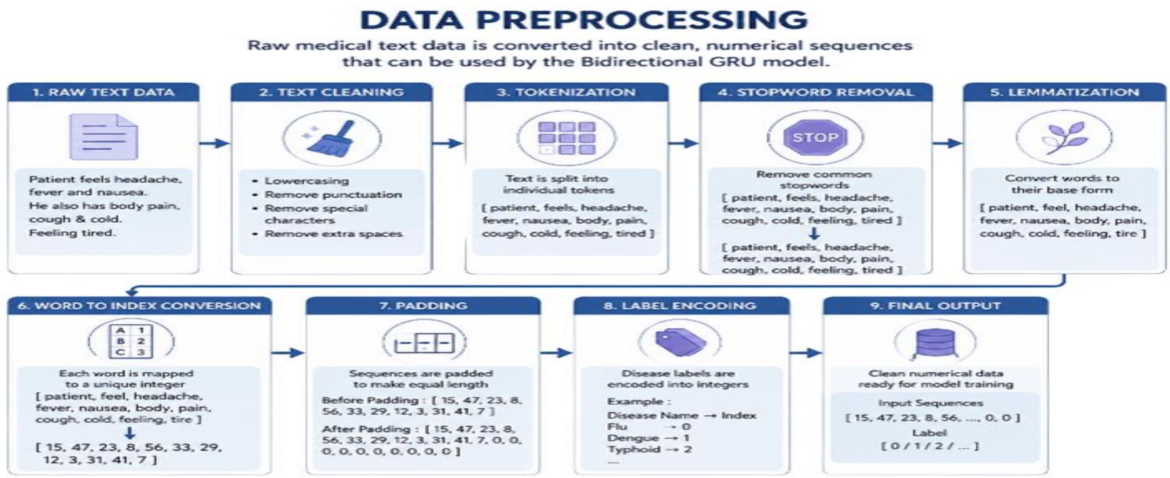

Fig. 7. data Preprocessing

D. *Embedding Layer*

The embedding layer takes the list of symptoms. Turns them into something the computer can understand. It is better than using a list of words because it helps the computer see how the words are related to each other. The model we are talking about uses a number, 300 to make sure the symptom words are represented in a way that the computer can work with. This means that symptoms that are similar will be represented in a way so the model can find connections between them that are not obvious. Embedding layers are used a lot in healthcare applications that involve natural language processing because they help the computer learn more, about the data and find patterns in the sequence of words [15] [22].

E. *Bidirectional GRU Layer*

The Bidirectional GRU layer is what we are focusing on here. This Bidirectional GRU layer is a kind of network. It is really good at dealing with data that comes in a sequence. At the time it helps with the problem of vanishing gradients. The Bidirectional GRU layer is unique. It looks at sequences of symptoms in two directions: forwards and backwards. This helps the model understand how things are connected. The Bidirectional GRU layer is also very good at predicting diseases. Our system has two Bidirectional GRU layers stacked on top of each other.

i. The first Bidirectional GRU layer has 256 units.
ii. The second Bidirectional GRU layer has 128 units.

Dropout layers were introduced between GRU layers to reduce overfitting and improve model generalization capability.

F. *Dense and Output Layers*

The extracted sequential features from the Bidirectional GRU layers are passed to fully connected dense layers for classification.

The architecture contains:

i. Dense Layer (256 Units)
ii. Batch Normalization Layer
iii. Dense Layer (128 Units)
iv. SoftMax Output Layer

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

Batch Normalization stabilizes training and improves convergence speed. The SoftMax activation function generates probability distributions across all disease classes and predicts the most probable disease category. The final output consists of predicted disease labels associated with user symptoms.

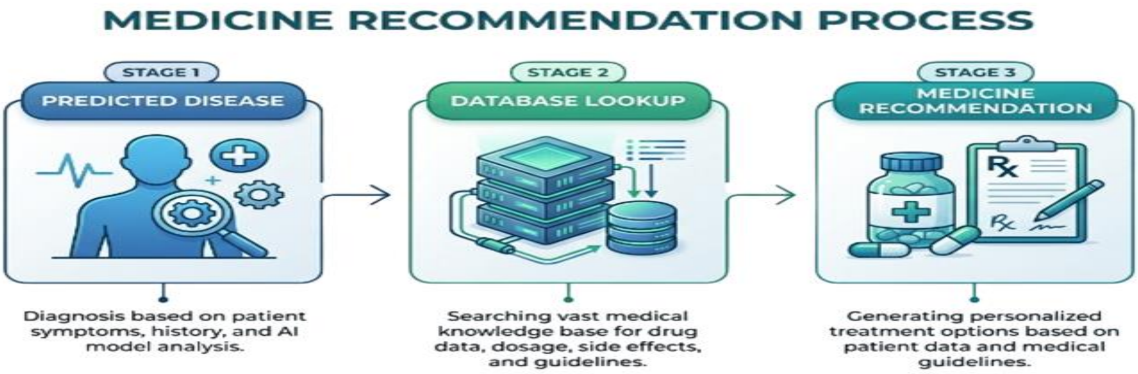

Fig. 8. Medicine Recommendation Process

G. *Performance Monitoring and Evaluation Module*

The proposed architecture also includes a performance evaluation module for analyzing classification results.
The evaluation process uses:

i. Accuracy
ii. Precision
iii. Recall
iv. F1-Score
v. ROC-AUC Analysis
vi. Confusion Matrix
vii. Confidence Score Analysis

The model achieved approximately 99% validation accuracy with strong ROC-AUC performance and very low misclassification rates.

H. *Summary of System Architecture*

The proposed system architecture integrates NLP preprocessing, Bidirectional GRU sequential learning, and intelligent medicine recommendation into a unified healthcare framework. The architecture effectively processes textual symptom data, learns contextual medical patterns, predicts diseases with high accuracy, and provides reliable medicine recommendations.

The combination of NLP techniques, deep sequential learning, and recommendation mechanisms makes the proposed system suitable for intelligent healthcare applications, telemedicine platforms, and AI-based clinical support systems.

VI. RESULTS AND DISCUSSION

A. *Experimental Setup*

We set up the Medicine Recommendation System using Python. We found some libraries like TensorFlow, Keras and Scikit-learn to be useful. We used a healthcare dataset that had kinds of diseases. 47 To be exact.

This dataset was split into two parts: one for training the system and one for testing it. The training part was bigger with 80 percent of the data and the testing part was smaller with 20 percent of the data.

We trained the Bidirectional GRU model using the Adam optimizer. We chose crossentropy as the loss function. To make the Medicine Recommendation System work better we used some techniques. We stopped training. Reduced the learning rate when it plateaus. We also used dropout regularization.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

TABLE IV EXPERIMENTAL CONFIGURATION

Parameter	Value
Deep Learning Framework	TensorFlow / Keras
Optimizer	Adam
Epochs	50
Batch Size	32
Learning Rate	0.001
Loss Function	Sparse Categorical Crossentropy
Total Disease Classes	47

B. *Training and Validation Performance*

The Bidirectional GRU model we are talking about did well when it was being trained. The accuracy of the training and the validation got better and better as time went on. At the time the training loss and the validation loss went down a lot. The Bidirectional GRU model was able to get 99 percent validation accuracy which is very good. It also had low validation loss.

This shows that the Bidirectional GRU model is very good at learning features and it can be used in different situations. Using Bidirectional GRU layers helped the model understand symptoms in the order and it made the model better at figuring out what disease someone has.

The training graphs also show that we were able to stop the model from overfitting. We did this by using dropout layers and making sure the dataset was balanced. We also used some techniques to optimize the model and this helped a lot. The Bidirectional GRU model was able to learn from the data without overfitting which is very important for a model, like this.

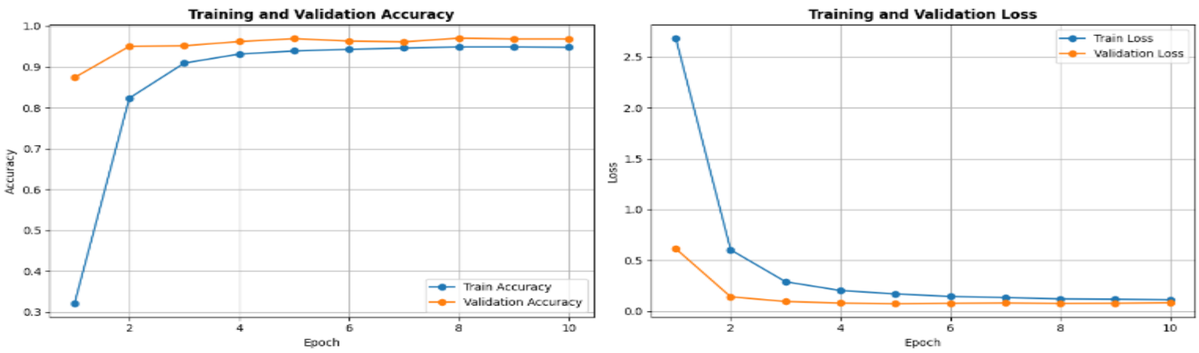

Fig. 9. Training and Validation Accuracy and Loss

C. *Confusion Matrix Analysis*

Confusion matrix analysis was performed to evaluate class-wise disease prediction performance. Both standard and normalized confusion matrices were generated for detailed classification analysis.

The normalized confusion matrix demonstrated that the majority of disease classes were classified correctly with very low misclassification rates. Only a few disease categories showed minor classification overlap due to similarity in symptom descriptions. The results indicate that the proposed Bidirectional GRU model effectively captures contextual relationships between symptoms and diseases

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

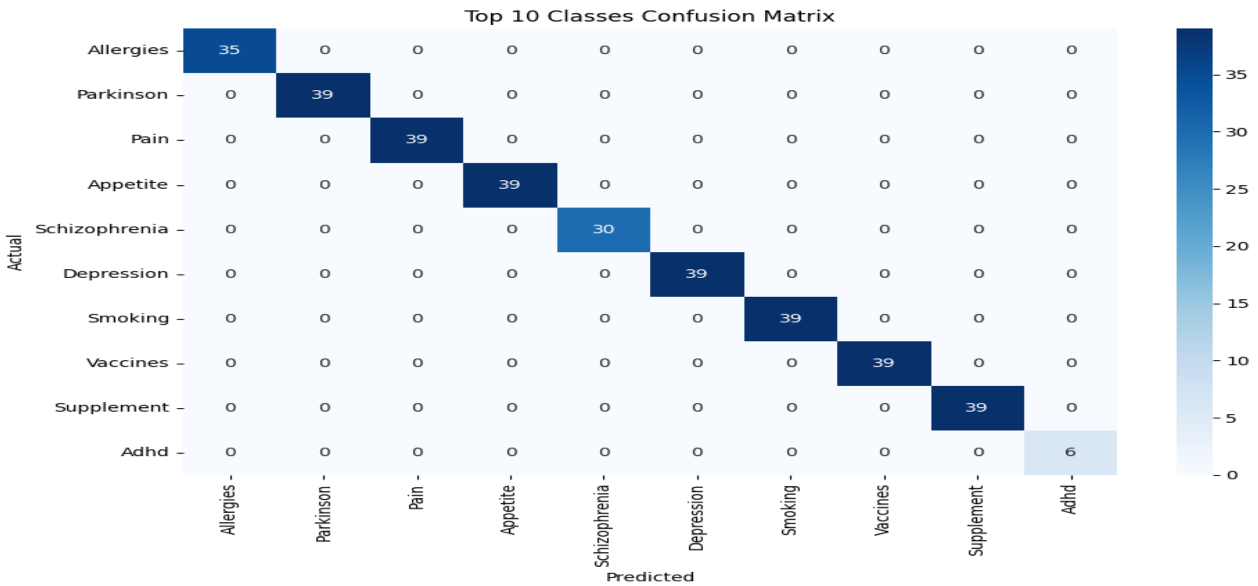

Fig. 10. Confusion Matrix of top 10 classes

D. *ROC-AUC Analysis*

We did a Receiver Operating Characteristic analysis to see how well our model can tell things apart. We made ROC curves for classes using two kinds of calculations: micro-average and macro-average. Our model did well on the ROC-AUC test getting scores very close to 1.0. This means it is very good at telling the difference between kinds of diseases. The ROC results show that our model is good at figuring out which disease is which when the symptoms are complicated. The model had ROC-AUC scores, for all the diseases, which is great.

E. *Precision, Recall and F1-Score Analysis*

The model we proposed was tested thoroughly using Precision, Recall and F1-Score. The results showed that it performed well in disease classes. When precision is high it means our model makes wrong predictions. High recall values show that it can detect diseases well. The F1-score also turned out to be good which means our model balances precision and recall. The results of the overall classification show that the Bidirectional GRU model we proposed can predict diseases and recommend medicine reliably.

TABLE V PERFORMANCE METRICS

Metric	Value
Accuracy	98–99%
Precision	High
Recall	High
F1-Score	High
ROC-AUC	Near 1.0

F. *Comparison with Existing Models*

The performance of the proposed Bidirectional GRU model was compared with traditional machine learning and deep learning approaches such as Support Vector Machine (SVM), Random Forest, and conventional recurrent neural network models.

The proposed model achieved superior performance because Bidirectional GRU captures sequential contextual dependencies more effectively than traditional classifiers. Additionally, NLP preprocessing and balanced dataset generation contributed to improved classification accuracy and reduced prediction bias.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 6, June 2026

DOI: 10.17148/IJARCCE.2026.15630

TABLE VI COMPARISON WITH EXISTING MODELS

Model	Accuracy
Naïve Bayes	Lower
Random Forest	Moderate
Support Vector Machine	Moderate
Traditional RNN	High
Proposed Bi-GRU Model	Highest

G. *Discussion*

The experimental results demonstrate that the proposed NLP-Based Medicine Recommendation System using Bidirectional GRU provides highly accurate disease prediction and efficient medicine recommendation performance.

The integration of NLP preprocessing with Bidirectional GRU sequential learning significantly improved contextual understanding of symptom descriptions. Dataset balancing techniques also improved classification fairness across minority disease classes.

The model achieved strong generalization capability with very low misclassification rates and high ROC-AUC performance. These results indicate that the proposed framework can be effectively used in intelligent healthcare systems, telemedicine applications, and AI-based clinical decision support platforms.

Although the system achieved excellent prediction accuracy, future improvements may include larger medical datasets, multilingual healthcare support, real-time cloud deployment, and integration with IoT healthcare devices for advanced smart healthcare applications.

VII. CONCLUSION AND FUTURE WORK

A. *Conclusion*

This research presented an NLP-Based Medicine Recommendation System using a Bidirectional Gated Recurrent Unit (Bi-GRU) model for intelligent disease prediction and medicine recommendation. The proposed system integrates Natural Language Processing techniques with deep learning architecture to analyse textual symptom descriptions and predict diseases accurately. Several preprocessing techniques including text cleaning, tokenization, sequence padding, and label encoding were applied to improve data quality and model learning capability. The Bidirectional GRU architecture effectively captured contextual relationships among symptoms by processing sequences in both forward and backward directions.

Experimental results demonstrated that the proposed model achieved approximately 99% prediction accuracy along with strong Precision, Recall, F1-Score, and ROC-AUC performance. Confusion matrix analysis also indicated very low misclassification rates across most disease categories. Compared with traditional machine learning approaches, the proposed Bidirectional GRU framework provided better contextual understanding, improved sequential learning capability, and higher classification performance. The developed system can support intelligent healthcare applications, telemedicine systems, and AI-based clinical decision support platforms.

Overall, the integration of NLP and Bidirectional GRU significantly improved disease prediction efficiency and medicine recommendation reliability for multi-class healthcare classification tasks [8], [10], [15].

B. *Future Work*

Although the proposed system achieved high prediction performance, several improvements can be implemented in future research.

Future enhancements may include:

i. Integration of larger real-world healthcare datasets
ii. Multilingual symptom analysis support
iii. Real-time cloud-based deployment
iv. Integration with IoT healthcare devices
v. Development of mobile healthcare applications
vi. Transformer-based deep learning architectures
vii. Personalized medicine recommendation systems

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 6, June 2026
DOI: 10.17148/IJARCCE.2026.15630

viii. Healthcare chatbot integration
ix. Explainable AI techniques for medical decision transparency

The proposed framework provides a strong foundation for developing intelligent AI-driven healthcare systems capable of supporting automated disease diagnosis and personalized medical assistance [1], [12], [14].

REFERENCES

[1]. Y. Bao and X. Jiang, "An Intelligent Medicine Recommender System Framework," 2016 IEEE ICIEA, 2016. DOI: 10.1109/ICIEA.2016.7603801
[2]. R. C. Chen, J. Y. Chiu, and C. T. Bau, "The Recommendation of Medicines Based on Multiple Criteria Decision Making and Domain Ontology," ICMLC, 2011. DOI: 10.1109/ICMLC.2011.6016685
[3]. R. C. Chen, Y. H. Huang, C. T. Bau, and S. M. Chen, "A Recommendation System Based on Domain Ontology and SWRL for Anti-Diabetic Drugs Selection," Expert Systems with Applications, 2012. DOI: 10.1016/j.eswa.2011.09.061
[4]. C. Doulaverakis, G. Nikolaidis, A. Kleontas, and I. Kompatsiaris, "GalenOWL: Ontology-Based Drug Recommendations Discovery," Journal of Biomedical Semantics, 2012. DOI: 10.1186/2041-1480-3-14
[5]. C. Doulaverakis, G. Nikolaidis, A. Kleontas, and I. Kompatsiaris, "Panacea: A Semantic-Enabled Drug Recommendations Discovery Framework," Journal of Biomedical Semantics, 2014. DOI: 10.1186/2041-1480-5-13
[6]. A. A. Hamed, R. Roose, M. Branicki, and A. Rubin, "T-Recs: Time-Aware Twitter-Based Drug Recommender System," IEEE/ACM ASONAM, 2012. DOI: 10.1109/ASONAM.2012.178
[7]. B. Stark, C. Knahl, M. Aydin, and K. Elish, "A Literature Review on Medicine Recommender Systems," IJACSA, 2019. DOI: 10.14569/IJACSA.2019.0100802
[8]. S. Garg, "Drug Recommendation System Based on Sentiment Analysis of Drug Reviews Using Machine Learning," IEEE Confluence, 2021. DOI: 10.1109/Confluence51648.2021.9377188
[9]. P. Keikhosrokiani, K. Balasubramaniam, and M. Isomursu, "Drug Recommendation System for Healthcare Professionals' Decision-Making Using Opinion Mining and Machine Learning," Springer, 2024. DOI: 10.1007/978-3-031-59091-7_15
[10]. M. Zomorodi et al., "RECOMED: A Comprehensive Pharmaceutical Recommendation System," Artificial Intelligence in Medicine, 2024. DOI: 10.1016/j.artmed.2024.102981
[11]. M. Sajde, H. Malek, and M. Mohsenzadeh, "RecoMed: A Knowledge-Aware Recommender System for Hypertension Medications," Informatics in Medicine Unlocked, 2022. DOI: 10.1016/j.imu.2022.100950
[12]. J. P. Gupta, A. Singh, and R. K. Kumar, "A Computer-Based Disease Prediction and Medicine Recommendation System Using Machine Learning Approach," IJARET, 2021. DOI: 10.34218/IJARET.12.3.2021.062
[13]. N. V. Shaik et al., "Medicine Recommendation System: Health Harbour," WJARR, 2025. DOI: 10.30574/wjarr.2025.25.2.0382
[14]. K. P. Swain et al., "Leveraging Machine Learning and Patient Reviews for Developing a Drug Recommendation System to Reduce Medical Errors," Open Bioinformatics Journal, 2024. DOI: 10.2174/0118750362291402240621044046
[15]. K. Tiwari and D. K. Singh, "Machine Learning-Based Recommendation System for Disease-Drug Material and Adverse Drug Reaction: Comparative Review," Materials Today: Proceedings, 2022. DOI: 10.1016/j.matpr.2021.05.404
[16]. L. Schmidt-Thieme, "Compound Classification Models for Recommender Systems," IEEE ICDM, 2005. DOI: 10.1109/ICDM.2005.46
[17]. F. Ricci, L. Rokach, and B. Shapira, Recommender Systems Handbook, Springer, 2015. DOI: 10.1007/978-1-4899-7637-6
[18]. C. C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016. DOI: 10.1007/978-3-319-29659-3
[19]. P. Lops, M. de Gemmis, and G. Semeraro, "Content-Based Recommender Systems: State of the Art and Trends," Springer, 2011. DOI: 10.1007/978-0-387-85820-3_3
[20]. J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender Systems Survey," Knowledge-Based Systems, 2013. DOI: 10.1016/j.knosys.2013.03.012
[21]. R. Burke, "Hybrid Recommender Systems: Survey and Experiments," User Modeling and User-Adapted Interaction, 2002. DOI: 10.1023/A:1021240730564
[22]. S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep Learning Based Recommender System: A Survey and New Perspectives," ACM Computing Surveys, 2019. DOI: 10.1145/3285029
[23]. G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems," IEEE TKDE, 2005. DOI: 10.1109/TKDE.2005.99

© IJARCCE This work is licensed under a Creative Commons Attribution 4.0 International License

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 6, June 2026
DOI: 10.17148/IJARCCE.2026.15630

[24]. X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Advances in Artificial Intelligence, 2009. DOI: 10.1155/2009/421425
[25]. Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," Computer, 2009. DOI: 10.1109/MC.2009.263
[26]. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," ACM KDD, 2016. DOI: 10.1145/2939672.2939785
[27]. J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," Annals of Statistics, 2001. DOI: 10.1214/aos/1013203451
[28]. L. Breiman, "Random Forests," Machine Learning, 2001. DOI: 10.1023/A:1010933404324
[29]. C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, 1995. DOI: 10.1007/BF00994018
[30]. T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," IEEE Transactions on Information Theory, 1967. DOI: 10.1109/TIT.1967.1053964
[31]. D. W. Aha, D. Kibler, and M. K. Albert, "Instance-Based Learning Algorithms," Machine Learning, 1991. DOI: 10.1007/BF00153759
[32]. H. Habehh and S. Gohel, "Machine Learning in Healthcare," Current Genomics, 2021. DOI: 10.2174/1389202922666210705124359
[33]. A. Rajkomar, J. Dean, and I. S. Kohane, "Machine Learning in Medicine," New England Journal of Medicine, 2019. DOI: 10.1056/NEJMra1814259
[34]. F. Jiang et al., "Artificial Intelligence in Healthcare: Past, Present and Future," Stroke and Vascular Neurology, 2017. DOI: 10.1136/svn-2017-000101
[35]. A. Esteva et al., "Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks," Nature, 2017. DOI: 10.1038/nature21056
[36]. T. Ching et al., "Opportunities and Obstacles for Deep Learning in Biology and Medicine," Journal of the Royal Society Interface, 2018. DOI: 10.1098/rsif.2017.0387
[37]. R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep Patient: An Unsupervised Representation to Predict the Future of Patients from Electronic Health Records," Scientific Reports, 2016. DOI: 10.1038/srep26094

© IJARCCE This work is licensed under a Creative Commons Attribution 4.0 International License