ISSN: 2633-4828 Vol. 5 No.4, December, 2023

International Journal of Applied Engineering & Technology

COMBINING THE LSTM ALGORITHM AND STACKING TECHNIQUE IN MACHINE LEARNING TO IMPROVE DISEASE PREDICTION ABILITY

Vo Hoang Khang[1] and Nguyen Dinh Thuan[2]

[1] University of Information Technology, Ho Chi Minh City 700000, Vietnam, Vietnam National University Ho Chi Minh City 700000, Vietnam, Hutech University 700000, Vietnam

[2] University of Information Technology, Ho Chi Minh City 700000, Vietnam, Vietnam National University Ho Chi Minh City 700000, Vietnam

[1]khangvh.ncs@ms.uit.edu.vn and [2]thuannd@uit.edu.vn

ABSTRACT

The problem of disease diagnosis in healthcare is always an intriguing field at the intersection of medicine and information technology. Disease diagnosis typically relies on patients' symptoms and clinical manifestations, and through their experience, doctors can diagnose the diseases afflicting patients. Many studies have used machine learning models and algorithms for diagnosing specific diseases such as influenza, dengue fever, dengue hemorrhagic fever, and cardiovascular diseases, based on symptoms and clinical manifestations. In this study, we used two datasets, the first containing 41 diseases and 17 symptoms respectively, including 4.921 million instances describing specific disease symptoms. The second set includes 148 disease types and 13 associated descriptive attributes for each disease type, this dataset contains 2.129 cases describing specific disease symptoms. By combining the LSTM deep learning model with Stacking (Ensemble Learning technique), based on symptoms and clinical manifestations, the study has tested a model capable of identifying 41 diseases with an accurate rate of 100% (1st dataset) and 96% accuracy with 148 diseases (2nd dataset). This model can assist doctors in quickly diagnosing patients at healthcare facilities, allowing for timely decisions on necessary biochemical tests to accurately determine the patient's condition and prescribe medication or provide appropriate treatment protocols. The study's initial results bring positive signals, especially in the ability to diagnose many diseases.

Index Terms – Deep Learning, Diseases, LSTM, Machine Learning, Stacking.

INTRODUCTION

Disease diagnosis is an important and fascinating challenge in healthcare and artificial intelligence. Through the patient's initial symptoms, the doctor can predict the likelihood of the patient suffering from one or more diseases. Disease symptoms are observable or felt signs of a person's health, mood, or behavior, such as rash, swelling, coughing, or sneezing. Signs of illness are unusual sensations or discomfort that patients often report to their doctor or health care professional. These symptoms may include pain, fever, itching, nausea, dizziness, etc.

Both symptoms and signs of disease can aid in the diagnosis or assessment of a patient's condition, and they can vary from person to person and depending on the type of disease. For example, a person with the common cold may exhibit symptoms such as fever, cough, runny nose, sore throat, and signs such as heat, pain, itching, and nausea. On the other hand, people suffering from a stroke may present with partial paralysis, difficulty speaking, and loss of balance with symptoms such as headaches, dizziness, and paralysis.

In this study, the proposed solution is the LSTM algorithm combined with the Stacking (Ensemble Learning) model in machine learning to achieve high accuracy in disease prediction. This model can support medical staff during the process of receiving patients. Based on clinical manifestations, with the support of machine learning models, staff can provide instructions for patients to go to the appropriate clinic. Doctors at these clinics can make a preliminary diagnosis, order additional testing and imaging (if needed), and then make an official diagnosis of the patient's condition. Finally, the doctor can prescribe medication or provide a suitable treatment regimen for each patient.

RELATED WORKS

Currently, there has been a lot of research and applications in the field of disease diagnosis. In the field of cancer diagnostics, Kyle Swanson and his team have applied machine learning to clinical cancer treatment, exploring the application of these techniques to medical imaging and molecular data obtained from liquid and solid biopsy specimens for cancer diagnosis, prognosis, and treatment design [1].

S. Shanthi and colleagues efficiently applied deep learning for the automatic classification of brain tumors using an optimized deep neural network. Medical images undergo preprocessing steps such as image enhancement and noise removal. Subsequently, the preprocessed images were classified using a CNN-LSTM model, achieving a maximum accuracy of 97.5% [2].

Fayroza Alaa Khaleel and her team used the Indian Pima Diabetes dataset to predict the onset of diabetes based on diagnostic criteria. The results obtained using Logistic Regression, Naïve Bayes, and K-nearest Neighbor algorithms are 94%, 79%, and 69% respectively [3].

Parkinson's disease (PD) is a neurodegenerative disorder that affects 60% of people over the age of 50. Aditi Govindu and colleagues conducted experiments on voice data (MDVP) from 30 Parkinson's patients and healthy individuals, training four machine learning models. Among these models, the Random Forest classification model achieved the highest accuracy, detecting PD at 91.83%, a sensitivity of 0.95 [4].

María Teresa García-Ordás and colleagues applied various deep-learning techniques to determine the severity of Parkinson's disease and predict disease progression in specific patients. They achieved a 99.15% success rate in distinguishing between severe and non-severe cases, using UPDRS (Unified Parkinson's Disease Rating Scale) and perceptron neural networks for both classification and retrieval regulation [5].

Predicting heart disease has become one of the most challenging medical tasks in recent years. Khandaker Mohammad Mohi Uddin and his team used various machine-learning techniques to identify heart abnormalities. The Decision Tree algorithm achieved the highest accuracy of 99.16% when compared to other machine learning algorithms [6]. Mohamed Djerioui and colleagues developed an intelligent system based on the LSTM technique to predict heart disease to make appropriate decisions to prevent and monitor heart disease and stroke [7]. Chintan M. Bhatt and colleagues used models such as random forest (RF), decision tree classifier (DT), multilayer perceptron (MP), and XGBoost (XGB) along with cross-validation techniques to diagnose the highest accuracy in predicting and predicting cardiovascular disease is 87.28% [8].

Dementia syndrome is a general term for loss of memory and thinking abilities so severe that it interferes with daily life. Alzheimer's disease accounts for about 60% to 80% of diseases that cause memory loss. Monika Sethi and colleagues classified Alzheimer's disease using Gaussian-based Bayesian parameter optimization for a deep convolutional LSTM network [9].

Machine learning algorithms have shown the ability to combine risk factors such as the timing of diabetes diagnosis and insulin use to stratify the risk of diabetic retinopathy, facilitating the development of better clinical decision support systems. The IDx system uses machine learning to analyze retinal images in diabetic retinopathy, achieving 87% sensitivity and 99% specificity, and was approved by the FDA in 2018, representing a significant advance in intelligence—artificial intelligence/machine learning in healthcare [10].

Several hospitals in Vietnam have conducted research and deployed diagnostic models, of which deep learning application is an outstanding example of identifying disease areas. At Hanoi Medical University, the Department of Dentistry uses deep learning to diagnose diseases, specifically detecting periapical lesions using X-ray images to diagnose the disease [11]. In addition, the DrAidTM application sponsored by VinBrain can detect and warn of COVID-19, including cases without symptoms or mild lung damage, achieving a sensitivity of up to 95%, a specificity of 99%, and an F1-score of 94%. Damien K. Ming and his research team used the XGBoost model to diagnose dengue fever based on the patient's age, gender, hematocrit, platelets, white blood cells, and lymphocyte counts. Model performance varies significantly over time due to seasonality and other factors. Incorporating a dynamic threshold that continuously learns from recent cases resulted in more consistent performance throughout the year (NPV > 90%) [12]. Regarding symptom-based disease diagnosis, Hai Thanh Nguyen and colleagues proposed using text-processing techniques combined with classifiers (such as Random Forests), Multilayer Perceptron (MLP), Embedding, and bidirectional long short-term memory (LSTM) of symptoms. As observed from the results, deep two-dimensional LSTM can reach 0.982 AUC in classifying 10 diseases on 230,457 pre-diagnosis samples collected from Vietnamese hospitals used in the training and testing phase [13]. Tran Dinh Toan and his colleagues conducted two experiments applying many supervised learning methods to classify diseases on three paraclinical data sets: heart disease, chronic kidney disease, and breast cancer [14].

AI technology in healthcare has the potential to improve healthcare access and quality of healthcare. Currently, in countries with many medical facilities, the company is still researching and finding new machine-learning models and algorithms as well as how to combine machine-learning models to improve diagnostic capabilities. Diagnose and detect diseases through symptoms, sounds, images... These studies hope to bring many benefits to people's health care.

METHODS

A. LSTM Neural Network [15], [16]

Long Short-Term Memory (LSTM) neural networks are a type of recurrent neural network used in deep learning. LSTMs can handle long and complex sequential data, such as speech, text, video, and music. LSTM networks can store information and pass it from one layer to another. The output of the hidden layer depends on information from all previous time steps.

LSTM architecture:

The LSTM network can consist of multiple LSTM cells connected. The idea of LSTM is to introduce an additional cell internal state (s_t) and three gates to filter the input and output information for the cell, including the input gate (i_t), the forget gate (f_t), and the output gate (o_t). At each time step *t*, the gates sequentially receive input values (x_t) representing elements in the input sequence and the previous hidden state (h_{t-1}) obtained from the output of memory cells from the previous time step *t-1*. Each gate serves to filter information for different purposes. The gates are defined as follows:

+ Input Gate: This gate determines which values from the input will be used to modify the memory. The sigmoid function decides which values should pass through (0 or 1), and the *tanh* function provides weights to the transmitted values, determining their significance in the range from -1 to 1.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i)$$

$$C_t = \tanh(W_c.[h_{t-1}, x_t] + b_c)$$

+ Forget Gate: This gate identifies the details that need to be discarded from the cell. A sigmoid function determines it. It considers the previous state (h_{t-1}) and the input content (x_t) and outputs a number between 0 (forget this) and 1 (retain this) for each element in the cell state c_{t-1}.

$$f_t = \sigma(W_t.[h_{t-1}, x_t] + b_f)$$

+ Output Gate: The input and memory of the cell are used to determine the output. A sigmoid function decides which values should pass through (0 or 1), and a *tanh* function decides which values should pass through (between 0 and 1). The *tanh* function provides weights to the transmitted values, determining their significance in the range from -1 to 1, and multiplies them with the sigmoid output.

$$O_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

The hidden state h_t can be used as the output of the model or passed into the model at subsequent steps. The context state C_t can also be utilized within the model, depending on the specific requirements of the application.

The parameters W_f, b_f, W_g, b_g, W_i, b_g, W_o, b_o are the weight matrices and bias vectors of the LSTM model, learned through the backpropagation process during training.

LSTM model operation:

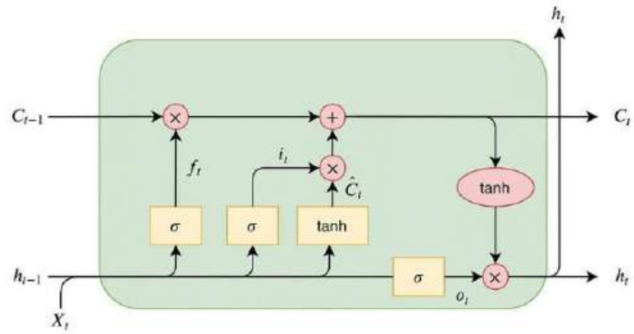


Fig. 1. The Operational Model Of LSTM

B. Stacking [17], [18], [19]

Stacking (also known as stacked generalization) is a machine learning algorithm that belongs to the Ensemble Learning model. It uses a set of base models to create a stronger overall model. In Stacking, base models are trained on the initial dataset and then used to predict values for new data. The prediction results from the base models are used as inputs for the overall model. The overall model is then trained based on the prediction results from the base models to make the final predictions. The goal of the Stacking model is to combine the strengths of different base models and leverage their diversity to improve predictive performance. The processing model of Stacking:

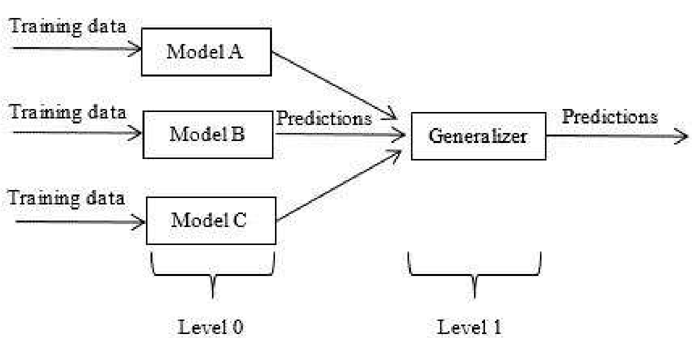


FIG. 2. The Processing Model Of Stacking

C. Two proposed processing models:

Model 1: LSTM with 2 layers.

In the model, the LSTM layer is applied to capture important features in the dataset. In contrast, the Dropout layer is used to address the issue of overfitting, and the Dense layer is responsible for generating the final output.

This model is constructed using the tf.keras.Sequential class from TensorFlow. It consists of 2 LSTM layers with 32 and 16 units respectively, connected through Dropout layers with dropout rates of 0.3 and 0.2 respectively. Finally, the model is terminated with a Dense layer with 41 units and a softmax non-linear activation function to provide the final predictions.

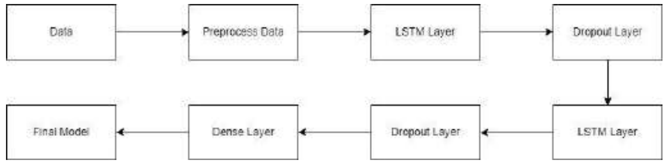


FIG. 3. The LSTM with 2 Layers

The functions of each layer are as follows:

✓ Long Short-Term Memory (LSTM) class:

o Function: Process data series, especially useful when working with time series or sequential data. This class has 32 memory units (or cells) in this case.

o input_shape: Requires the input sequences to have the shape of (x_train.shape[1], 1), where x_train.shape[1] is the length of each input sequence and 1 is the number of features in each step time.

o return_sequences: Set to True, indicates that the class should return the entire result sequence for each input sequence. This is often used when layering multiple LSTMs.

✓ Dropout class:

o Function: Introduces dropout into the model, randomly setting a portion (30%) of the input units to 0 during each update during training. This helps prevent over-refinement by introducing a degree of noise or uncertainty.

✓ Long Short-Term Memory (LSTM) class:

o Function: The LSTM layer is different with 16 memory units. This layer processes the strings returned by the first LSTM layer.

o return_sequences: Set to False, indicating that the layer should only return the output of the last time step for each input sequence.

✓ Dropout class:

o Function: Another Dropout layer, similar to the second layer, but with a lower dropout rate (20%).

✓ Dense class:

o Function: Creates the final output of the model. Has num_classes unit (output classes) and uses softmax activation function, commonly used in multi-class classification problems. The output represents the probability of the input belonging to each class.

The processing process of the LSTM model includes the following steps:

+ **Step 1**: Divide the dataset into two parts: the training dataset (2/3 of the dataset) and the test dataset (1/3 of the dataset).

+ **Step 2**: Build a sequential neural network model with fully connected layers and convolutional layers.

+ **Step 3**: Train the model based on the training dataset.

+ **Step 4**: Evaluate the model's accuracy based on the test dataset.

Model 2: Combination of LSTM and Stacking.

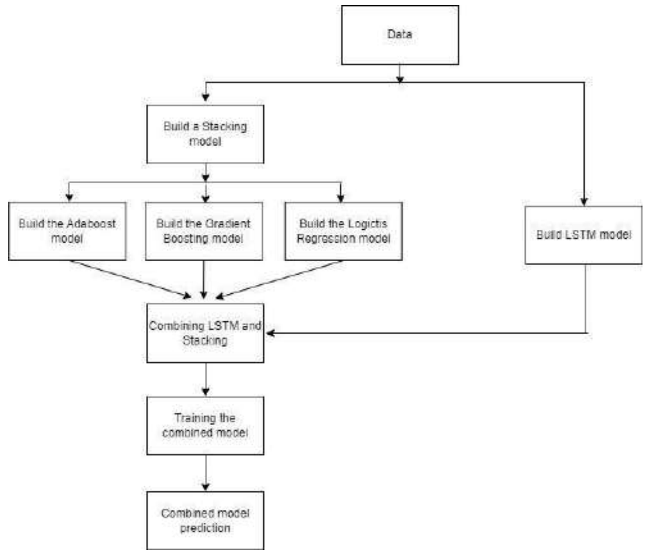


FIG. 4. The Processing Model Combining LSTM And Stacking

The processing process of the model combining LSTM and Stacking includes the following steps:

+ **Step 1**: Read input data.

+ **Step 2**: Initialize the LSTM model (setting up LSTM parameters as in Stage 1).

+ **Step 3**: Initialize the Adaboost, GradientBoosting, and Logistic Regression models.

+ **Step 4**: Combine the Adaboost, GradientBoosting, Logistic Regression, and LSTM models using the Stacking technique [20].

+ **Step 5**: Perform model training.

+ **Step 6**: Test and evaluate the model.

D. Evaluation formulas [21]

Precision is calculated using the following formula:

$$\frac{True\ Positive}{True\ Positive + False\ Positive} = \frac{Correct\ True\ Positive\ Predictions}{True\ Positive\ Predictions}$$

Recall is calculated using the following formula:

$$\frac{True\ Positive}{True\ Positive + False\ Negative} = \frac{Number\ of\ correct\ TP\ predictions}{Total\ number\ of\ TP}$$

Calculation formula for F1-score: $F1\ score = 2 * \frac{Precision*Recall}{Precision+Recall}$

EXPERIMENT RESULTS

A. Research dataset

This study used 2 datasets for experiments.

Information about the first dataset:

Access the dataset through the following link:

https://www.kaggle.com/code/youssefatef/disease-predictions-100-accuracy/input?select=dataset.csv.

The dataset.csv contains information on 41 different diseases along with 17 corresponding symptoms. It comprises 4,921 records, each describing the symptoms of a specific disease.

Detail Compact Column 10 of 18 columns

Disease	Symptom_1	Symptom_2	Symptom_3
Diseases that may be present	the symptoms experienced during the disease	the symptoms experienced during the disease	the symptoms experienced during disease
41 unique values	vomiting 17% fatigue 14% Other (3408) 69%	vomiting 18% fatigue 8% Other (3648) 74%	fatigue high_fever Other (3870)

FIG. 5. Information About The Disease Dataset

Disease	Symptom_1	Symptom_2	Symptom_3	Symptom_4	Symptom_5
Fungal inf	itching	skin_rash	nodal_skin_eruptions	dischromic_patches	
Fungal inf	skin_rash	nodal_skin_e	dischromic_patches		
Fungal inf	itching	nodal_skin_e	dischromic_patches		
Fungal inf	itching	skin_rash	dischromic_patches		
Fungal inf	itching	skin_rash	nodal_skin_eruptions		
Fungal inf	skin_rash	nodal_skin_e	dischromic_patches		
Fungal inf	itching	nodal_skin_e	dischromic_patches		
Fungal inf	itching	skin_rash	dischromic_patches		
Fungal inf	itching	skin_rash	nodal_skin_eruptions		
Fungal inf	itching	skin_rash	nodal_skin_eruptions	dischromic_patches	
Allergy	continuous_s	shivering	chills	watering_from_eyes	
Allergy	shivering	chills	watering_from_eyes		
Allergy	continuous_s	chills	watering_from_eyes		
Allergy	continuous_s	shivering	watering_from_eyes		
Allergy	continuous_s	shivering	chills		

FIG. 6 The Dataset Contains Information About Diseases And Their Symptoms

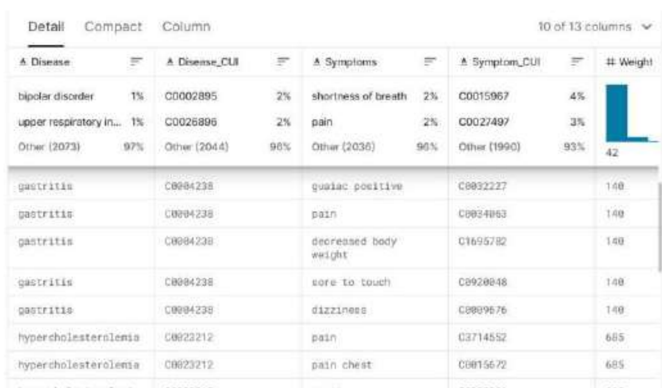
This dataset has a low-class imbalance, with approximately an equal number of samples in each disease group. It is widely used in data science and machine learning to train disease prediction models.

Information about the second dataset:

Access the dataset through the following link:

https://www.kaggle.com/datasets/usamag123/disease-prediction-through-symptoms

This is a symptom-based disease prediction dataset. This data set includes 13 data columns containing patient symptoms and information. The data set includes 148 specific diseases.



Detail Compact Column 10 of 13 columns

Disease	Disease_CUI	Symptoms	Symptom_CUI	Weight
bipolar disorder 1% upper respiratory in... 1% Other (2073) 97%	C0002895 2% C0026896 2% Other (2044) 96%	shortness of breath 2% pain 2% Other (2036) 96%	C0015967 4% C0027497 3% Other (1990) 93%	42
gastritis	C0004238	guaiac positive	C0032227	140
gastritis	C0004238	pain	C0034063	140
gastritis	C0004238	decreased body weight	C1695782	140
gastritis	C0004238	sore to touch	C0920048	140
gastritis	C0004238	dizziness	C0009676	140
hypercholesterolemia	C0023212	pain	C3714552	685
hypercholesterolemia	C0023212	pain chest	C0015672	685
hypercholesterolemia	C0023212	sweat	C0235886	685

Fig. 7 The Data Set Includes 148 Diseases and Symptoms

This study utilized open-source software libraries such as Numpy, Pandas, and Keras, along with Python 3.11 programming language to set up an LSTM neural network model. The execution results were obtained on a laptop with the following configuration: CPU: Intel core i7-12700H, GPU: Intel Iris XE Graphics, RAM: 16Gb.

The main parameters of the LSTM function are presented in

Table I. Parameters of LSTM

Parameters	Data types	Description	Default value
activation	Str	The activation function used. If None is passed, no activation function is used (i.e., "linear": a(x) = x).	'tanh'
recurrent_activation	Str	The activation function is used within the recurrent steps. If None is passed, no activation function is used (i.e., "linear": a(x) = x).	'sigmoid'
recurrent_initializer	Str	Initializer for the 'recurrent_kernel' weight matrix, used for the linear transformation of the recurrent state.	'orthogonal'
bias_initializer	Str	Bias vector initializer.	'zeros'
unit_forget_bias	Boolean	If True, add 1 to the bias of the forget gate during initialization.	True
dropout	Float	It has a value from 0 to 1. The rate at which the linear transformation of the inputs is reduced.	0
recurrent_dropout	Float	It has a value from 0 to 1. The rate at which the linear transformation of the recurrent states is reduced.	0
return_sequences	Boolean	Return the last output in the output sequence or the full sequence.	False
return_state	Boolean	Return the last state apart from the output or not.	False
go_backwards	Boolean	If True, process the input sequence in reverse and return the reversed sequence.	False
stateful	Boolean	If True, the final state of each sample at index i in the batch will be used as the initial state for the sample at index i in the next batch.	False

Experimental Procedure:

This study conducted experiments in 2 stages:

+ Stage 1: Execute Model 1 on the first and second datasets respectively.

+ Stage 2: Execute Model 2 on the first and second datasets respectively.

B. Stage 1

Execute Model 1 on the first dataset consisting of 41 diseases and 17 symptoms. Below is the Classification report and Confusion matrix:

Results: Accuracy: ~ 96%. Confusion Matrix:

Table II. Results After Testing LSTM For The First Data Set

Class	Precision	Recall	F1-score	Support
0	0.67	0.88	0.76	32
1	0.96	0.67	0.79	39
2	0.97	0.95	0.96	41
3	1.00	1.00	1.00	36
4	1.00	1.00	1.00	35
5	0.70	0.92	0.80	36
...	...	...	...	...
36	1.00	1.00	1.00	35
37	1.00	1.00	1.00	34
38	1.00	1.00	1.00	36
39	0.95	1.00	0.97	37
40	1.00	1.00	1.00	39
Accuracy			0.96	1476
Macro avg	0.96	0.96	0.96	1476
Weighted avg	0.97	0.96	0.96	1476
Accuracy	**0.9613**			

Do the same for the second data set including 148 types of diseases.

Below is the Classification report and Confusion matrix:

Table III. Results After Testing LSTM For The Second Data Set

Class	Precision	Recall	F1-score	Support
0	0.00	0.00	0.00	6
1	0.00	0.00	0.00	2
2	0.00	0.00	0.00	1
3	0.33	0.40	0.36	5
4	0.00	0.00	0.00	4
5	0.00	0.00	0.00	6
...	…	…	…	…
143	0.00	0.00	0.00	3
144	0.00	0.00	0.00	3
145	0.50	0.71	0.59	7
146	0.08	1.00	0.15	4
147	0.00	0.00	0.00	2
Accuracy			0.17	533
Macro avg	0.05	0.14	0.06	533
Weighted avg	0.07	0.17	0.08	533

Accuracy	**0.1726**			

C. Stage 2

Execute Model 2 on the first dataset. Below is the Classification report and Confusion matrix:

Results: Accuracy: 100%. Confusion Matrix:

Table IV. The Results of the LSTM Model Combined with the Stacking Technique

Class	Precision	Recall	F1-score	Support
0	1	1	1	32
1	1	1	1	39
2	1	1	1	41
3	1	1	1	36
4	1	1	1	35
5	1	1	1	36
...	…	…	…	…
36	1	1	1	35
37	1	1	1	34
38	1	1	1	36
39	1	1	1	37
40	1	1	1	39
Accuracy			1.00	1476
Macro avg	1.00	1.00	1.00	1476
Weighted avg	1.00	1.00	1.00	1476
Accuracy	**1.00**			

It can be observed that when executed individually, the LSTM model has provided reliable results with an accuracy of up to 96% and a training time of about 5 minutes. However, when combining LSTM with the Stacking technique, the study achieved an absolute accuracy of 100% for ACC with an execution time of approximately 60 minutes. In practice, when using a combination of machine learning models, we may have to accept the trade-off between execution time and model training time.

Do the same for the second data set including 148 types of diseases. Below is the Classification report and Confusion matrix:

Table V. Results After Applying The LSTM Model Combined With The Stacking Technique To The Second

Class	Precision	Recall	F1-score	Support
0	1.00	0.83	0.91	6
1	1.00	1.00	1.00	2
2	1.00	1.00	1.00	1
3	1.00	1.00	1.00	5
4	1.00	0.75	0.86	4
5	1.00	1.00	1.00	3
...	…	…	…	…
143	1.00	0.83	0.91	6
144	1.00	1.00	1.00	2
145	1.00	1.00	1.00	1

146	1.00	1.00	1.00	5
147	1.00	0.75	0.86	4

Accuracy			0.96	0.96
Macro avg	0.96	0.97	0.96	533
Weighted avg	0.96	0.96	0.96	533
Accuracy	**0.9606**			

It can be seen that, after combining LSTM and stacking techniques, the model's performance improves significantly. This improvement is clearly shown in the second dataset (ACC increases from 17% to 96%).

Below are some results of studies related to disease diagnosis.

Table VI. Results of Studies Related to Disease Diagnosis

Research Title	**Used Algorithm**	**Accuracy**
[7] Heart Disease prediction using MLP and LSTM models	Multi-Layer Perceptron and Long Short-Term Memory	99.8%
[8] Effective Heart Disease Prediction Using Machine Learning Techniques	Multilayer perceptron with cross-validation	87.28%
[9] Classification of Alzheimer's Disease Using Gaussian-Based Bayesian Parameter Optimization for Deep Convolutional LSTM Network	Gaussian-based Bayesian optimized for Long Short-Term Memory	92.56%
[22] A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images	Concurrent Neural Network and Long Short-Term Memory	99.4%
[13] Deep bidirectional	A text-based disease diagnosis	98.2%

LSTM for disease classification supporting hospital admission based on pre-diagnosis: a case study in Vietnam	using text-processing techniques combined with classifiers	
[14] Data mining in the healthcare system on patients' clinical symptoms dataset	SVM, Decision Tree and Naïve Bayes	94%
(*) Combining the LSTM Algorithm and Stacking Technique in Machine Learning to Improve Disease Prediction Ability	LSTM combined Stacking	100%, 96%

(*) This research

CONCLUSION

LSTM is used to address issues in natural language processing, such as text classification, machine translation, automatic text generation, speech synthesis, and other natural language data processing tasks. Additionally, LSTM can be applied to address problems in audio and image processing, such as speech recognition, image classification, face recognition, object detection, and other computer vision applications. One of the limitations of LSTM is that training the model can be time-consuming and computationally expensive. Furthermore, overfitting can occur if regularization techniques are not used.

Combining LSTM with the Stacking technique helps improve the accuracy of disease diagnosis, enabling doctors to save time and examine more patients during their working hours. Patients can also avoid unnecessary diagnostic costs, reducing additional expenses and waiting times. In practice, this combined model can be implemented at the patient intake stage in healthcare facilities. Medical teams can rely on clinical signs and symptoms to guide patients to the appropriate specialized clinics for their conditions, avoiding the need for general check-ups to identify the type of disease.

In the future, the research team will develop additional features to extract information from various medical tests and make disease predictions based on these tests as requested by the doctors (such as CT scans, ultrasound, X-rays, endoscopy, etc.). This will enable doctors to accurately diagnose diseases and provide more appropriate treatment plans.

REFERENCES

[1] Kyle Swanson, Eric Wu, Angela Zhang, Ash A. Alizadeh, James Zou, "From patterns to patients: Advances in clinical machine learning for cancer diagnosis, prognosis, and treatment," pp. 1772-1791, 2023.

[2] S. Shanthi, S. Saradha, J.A. Smitha, N. Prasath, H. Anandakumar, "An efficient automatic brain tumor classification using optimized hybrid deep neural network," KeAi, vol. 3, pp. 188-196, 2022.

[3] Fayroza Alaa Khaleel, Abbas M. Al-Bakry, "Diagnosis of diabetes using machine learning algorithms," vol. 80, no. 3, 2023.

[4] Aditi Govindu, Sushila Palwe, "Early detection of Parkinson's disease using machine learning," vol. 218, pp. 249-261, 2023.

[5] María Teresa García-Ordás, José Alberto Benítez-Andrades, Jose Aveleira-Mata, José-Manuel Alija-Pérez & Carmen Benavides, "Determining the severity of Parkinson’s disease in patients using a multi task neural network," 1 6 2023.

[6] Khandaker Mohammad Mohi Uddin, Rokaiya Ripa, Nilufar Yeasmin, Nitish Biswas, Samrat Kumar Dey, "Machine learning-based approach to the diagnosis of cardiovascular vascular disease using a combined dataset," vol. 7, 2023.

[7] M. Djerioui, M. Ladijal, Y. Brik, B. Attallah, "Heart Disease prediction using MLP and LSTM models," International Conference on Electrical Engineering (ICEE), 2020.

[8] Chintan M. Bhatt, Parth Patel, Tarang Ghetia, Pier Luigi Mazzeo, "Effective Heart Disease Prediction Using Machine Learning Techniques," MDPI, 6 2 2023.

[9] M. Sethi, S. Ahuja, S. Rani, P. Bawa, A. Zaguia, "Classification of Alzheimer’s Disease Using Gaussian-Based Bayesian Parameter Optimization for Deep Convolutional LSTM Network," Computational and Mathematical Methods in Medicine, 2021.

[10] Sriram Gubbi, Pavel Hamet, Johanne Tremblay, Christian A Koch, Fady Hannah-Shmouni, "Artificial Intelligence and Machine Learning in Endocrinology and Metabolism: The Dawn of a New Era," 28 3 2019.

[11] Vo Truong Nhu Ngoc, Agwu Chinedu Agwu, Le Hoang Son, Tran Manh Tuan, Cu Nguyen Giap, Mai Thi Giang Thanh, Hoang Bao Duy, Tran Thi Ngan, "The Combination of Adaptive Convolutional Neural Network and Bag of Visual Words in Automatic Diagnosis of Third Molar Complications on Dental X-Ray Images," MDPI, vol. 10, no. 4, 9 4 2020.

[12] Damien K. Ming, Bernard Hernandez et al., "The Diagnosis of Dengue in Patients Presenting With Acute Febrile Illness Using Supervised Machine Learning and Impact of Seasonality," Frontiers, 14 3 2022.

[13] Hai Thanh Nguyen, Khoa Dang Dang Le, Ngoc Huynh Pham, Chi Le Hoang Tran, "Deep bidirectional LSTM for disease classification supporting hospital admission based on pre-diagnosis: a case study in Vietnam," International Journal of Information Technology, vol. 15, p. 2677–2685, 15 5 2023.

[14] Toan Tran Dinh, Lan Huynh Thi Chau, Tho Tran Van, Hung Le Minh, Lang Tran Van, "DATA MINING IN HEALTHCARE SYSTEM ON PATIENTS CLINICAL SYMPTOMS DATASET," Proceedings of the 12th National Science and Technology Conference on Basic and Applied Information Technology (FAIR), 2019.

[15] Sepp Hochreitor, Jurgen Schmidhuber, "Long Sort-Term Memory," Neural Computation, pp. 1735-1780, 12 1997.

[16] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network," Elsevier, vol. 404, 3 2020.

[17] L. Bbeiman, Bagging Predictors, vol. 24, 1996, pp. 123-140.

[18] J. H. Friedman, Stochastic gradient boosting, 2002, pp. 367-378.

[19] Vo Hoang Khang, Cao Tung Anh, Nguyen Dinh Thuan, "Detecting Fraud Transaction using Ripper Algorithm Combines with Ensemble Learning Model," International Journal of Advanced Computer Science and Applications, vol. 14, 2023.

[20] B. Pavlyshenko, "Using Stacking Approaches for Machine Learning Models," IEEE Second International Conference on Data Stream Mining & Processing, 2018.

[21] D. M. W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," International Journal of Machine Learning Technology 2:1, pp. 37-63, 2011.

[22] Z. Islam, M. Islam, A. Asraf, "A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images," Informatics in Medicine Unlocked, 2020.