Journal of
Neurology & Experimental Neuroscience

https://doi.org/10.17756/jnen.2022-092

Research Article

Open Access

SeizureSeeker: A Novel Approach to Epileptic Seizure Detection Using Machine Learning

Hamza Arman Lateef [1*], Gabriel Ralston[1], Tony Bright[1], Aravindh Soundarajan[2] and Jessica Carpenter[3]

[1]*George Mason University, Virginia, USA*
[2]*Amrita University, Tamil Nadu, India*
[3]*University of Maryland School of Medicine, Baltimore, USA*

[*]Correspondence to:
Hamza Arman Lateef
George Mason University, Virginia, USA
Email: hamzaalateef@gmail.com

Received: September 24, 2021
Accepted: January 04, 2022
Published: January 06, 2022

Citation: Lateef HA, Ralston G, Bright T, Soundarajan A, Carpenter J. 2022. SeizureSeeker: A Novel Approach to Epileptic Seizure Detection Using Machine Learning. *J Neurol Exp Neurosci* 8(1): 1-8.

Published by United Scientific Group

Abstract

Background: Epilepsy is a neurologic disease characterized by seizures which occur due to sudden and synchronized bursts of excessive electrical energy in the brain. An electroencephalogram, or EEG, can detect seizures in real time but requires trained medical expertise for extended periods of time. The main objective of this research was to devise a more efficient method (SeizureSeeker) for analyzing EEG data using machine learning algorithms that allows for complex data processing and can automatically distinguish between normal EEG signal and epileptic seizures.

Methods and Study Design: An open access EEG dataset, containing pre-identified records of 500 patients, was used. Seizure activity was designated as a simple binary 1 or 0, where 1 indicated a seizure and 0 indicated no seizure. The database was then partitioned into two randomly assigned groups, a training set of 80% of the data and a testing set containing the remaining 20%. The study compared 3 different classification algorithms: Logistic Regression, Support Vector Machines (SVM), and Long Short-Term Memory (LSTM). All models were fitted using existing software from Python libraries and the Orange data mining application.

Results: Logistic Regression had poor accuracy, but SVM achieved impressive results with an overall accuracy of 94%. LSTM is a more complex algorithm based on recurrent neural networks and generated near perfect classification results with an accuracy of 99%.

Conclusions: The memory property of the LSTM model makes it an ideal choice for the time series EEG data. The LSTM results proved the efficacy of the machine learning model to automatically detect seizure activity in EEG data. Models such as SeizureSeeker can be developed to reach more timely diagnoses of seizures and can be used where access to specialized medical expertise is especially limited.

Keywords

Seizure, Seeker, Neurology

Introduction

Seizures are caused by sudden, uncontrolled electrical disturbances in the brain and they can present in humans as changes in behavior, movements or feelings, and altered consciousness [1]. Recurrent seizures characterize epilepsy, a condition which affects around 50 million people worldwide, making it one of the most common neurological diseases globally [2]. Nearly 80% of people with epilepsy live in low- and middle-income countries and 75% of epilepsy patients living in low-income countries do not get optimal treatment. Seizures can also occur in hospitalized patients without a diagnosis of epilepsy and in very young infants in the first month of life. Such seizures that occur in critically ill individuals are not necessarily accompanied by physical convulsions and up to 92% of seizures in the ICU are clinically silent [3]. The success of seizure treatment is also dependent on seizure duration, with longer seizures becoming harder to contain. Thus, early detection and treatment of seizures is essential to improve the outcome.

An electroencephalogram (EEG) can be used to detect the abnormal surges of electrical activity that occur during a seizure, when brain waves are disrupted by sudden and synchronized bursts of excessive electrical energy. The EEG records electrical activity on the surface of the brain to identify the location of the abnormally firing neurons that cause seizures [4]. Small electrodes are attached to various parts of the patient's scalp (Figure 1), and the EEG records the electrical power passing between two nodes, known as a channel. The result of this process is a record of numerous channels of electrical activity in different parts of the brain over time (Figure 2). Currently, seizure detection, even in ICUs, relies on trained medical professionals who must meticulously read EEG data to identify seizures.

The manual process of seizure detection is time-consuming and heavily depends on trained human expertise. Also, the process is subject to human error and can lead to misdiagnosis. In modern ICUs, however, while vital functions are monitored continuously, brain functions are not. In up to 82% of monitored neurological patients, continuous EEG will have an impact on medical decision making (Jordan 2005) [5], but only four of every five centers in the United States have in-house continuous EEG monitoring available, and even in these cases there is no continuous reading provided. The ability to have automated seizure detection, especially in ICU settings where expert interpretation of the EEG is unavailable, is of utmost importance.

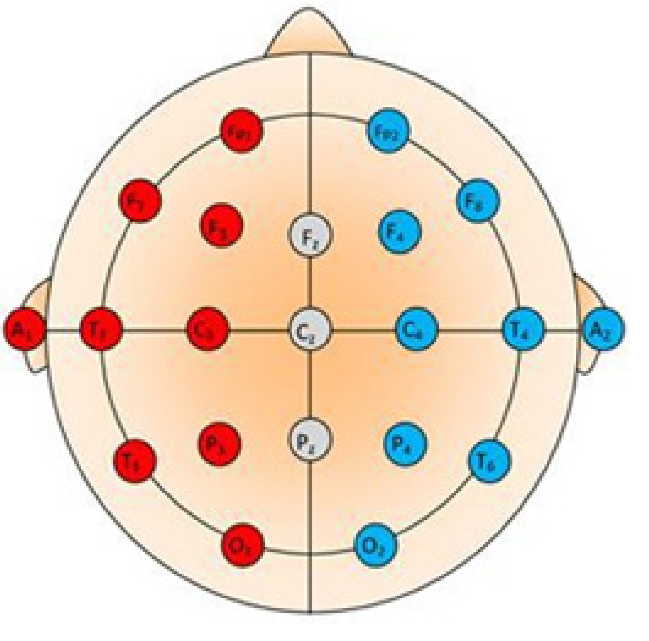

Figure 1: [21]. 10/20 System of electrode placement, F - Frontal lobe, T - Temporal lobe,C - Central lobe, P - Parietal lobe, O - Occipital lobe, Even numbers denote the right side of the head, Odd numbers denote the left side of the head.

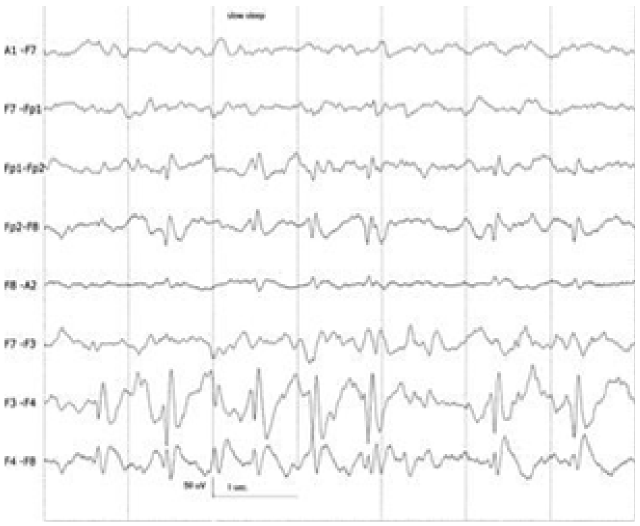

Figure 2: [22]. EEG recording from patient having a seizure.

Creating a means for automated analysis of EEGs can improve the quality of patient care by shortening the time to diagnosis, reducing manual error, and automatically detecting injurious events. Thordoff and colleagues conducted one of the first successful automated classifications of seizures with EEG data in 2016. They created a Recurrent Convolutional Network that converted 969 hours of signal data from 23 epileptic child patients into images. More recently, research generated the largest known database of EEG seizure data, called the TUH EEG Seizure Corpus, and built a model to classify whether the data represented a seizure or not [6, 7].

The aim of this project was to devise a more efficient method for analyzing EEG data using machine learning algorithms that allows for complex data processing and provides high accuracy.

Methods and Materials

The process of developing machine learning models for classification of data typically consists of 4 main steps (Figure 3). The first step is to acquire the data that will be used to train the models. Acquiring enough data with the appropriate explanatory variables is important to produce models with good classification performance. Data Preparation is one of the most critical steps in the machine learning process and involves cleaning the data and partitioning the data into a training and test set [8]. With appropriate data, multiple machine learning methods can be used to train models that can then be evaluated and improved in the final step.

Data Collection
Data Preparation
Model Training
Model Evaluation

Figure 3: Primary steps in conducting a machine learning classification model.

Data collection

Machine learning classifiers attempt to predict the class of observations of data points using a machine learning model that maps the values of the explanatory variables to a predicted target class [9]. This process typically requires large amounts of data that has been correctly classified that can be used to train the models. Such EEG data could be from patients that exhibited both seizure and non-seizure activity. Unfortunately, EEG data is not readily available and conducting experiments with live patients was not feasible for this study. Fortunately, the popularity of machine learning studies has produced online repositories of data. Kaggle.com is one of the most popular machine learning sites and contains data from the American Epilepsy Society that is available to researchers to develop and improve epilepsy prediction models [8].

This project used a data set from Kaggle that contained EEG data for 500 patients. The EEG for each patient was broken up into 23 single channel segments of 23.6 seconds, resulting in 11,500 total observations. For each observation, the EEG signal was recorded at 178 time intervals (every 0.13 sec). These values were used as the explanatory variables, while the response variable was the classification of seizure activity on a scale of 1-5. The description for the values in the response variable is provided in table 1. An example of the EEG output for each of the seizure activity levels is shown in figure 4. Initial observations indicate that differentiating between level 1 and 2 could be challenging, while levels 3-5 show more obvious differences in the EEG pattern.

Table 1: Description of EEG response variable values.

Response Value	Description
1	Patient exhibiting seizure activity
2	Section of brain has a tumor, but no seizure activity
3	Section of brain previously had a tumor, but no seizure activity
4	Patient had eyes closed, but no seizure activity
5	Patient had eyes open, but no seizure activity

Data preparation

One of the most important steps in the machine learning process is preparing the data by partitioning it into 2 sets: a training set and a test or validation set [10]. Partitioning the data is important because it helps to ensure that the model is robust and will maintain its classification performance when faced with new data. The training set is used to build or fit the model. If the model were using all of the data, the model would likely perform well, but would likely not perform well with a new set of data [10]. The training data is the subset that the model uses to learn the relationships between the explanatory variables and the response variable. In our study, the models learn the relationship between the EEG signal at various time intervals and the resulting seizure activity value. To further guard against over fitting the data to the training set, cross validation can be used.

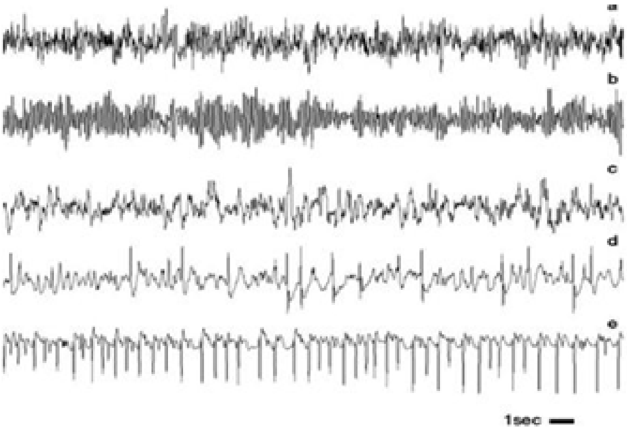

Figure 4: EEG output data for the seizure activity level that is the response variable for this research.

Cross-validation further divides the training set into smaller "folds" which are smaller training and test sets. Models are trained across all folds which helps to improve overall model performance [8]. Figure 5 illustrates how the original data is partitioned into training and test data and how cross validation is used. Once the models are fit to the training data, the test, or validation, set is used to evaluate the model performance.

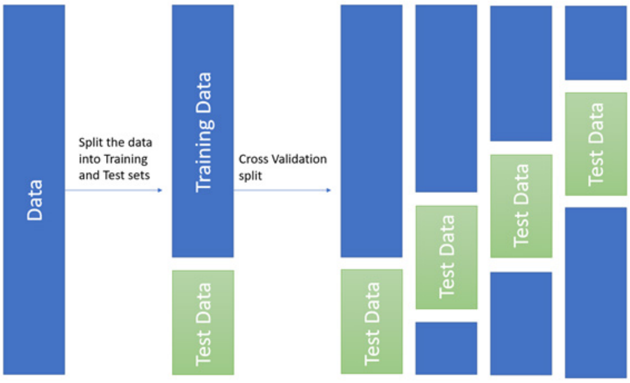

Figure 5: Partition of data into training and test sets with cross validation.

For this study, the 80/20 Pareto Rule was used to partition the data. The 80/20 rule suggests that 80% of the response comes from 20% of the causes, and this principle has proven statistically effective in explaining phenomena involving humans, machines, and the environment [11]. For this reason, most machine learning studies will partition data into a training set consisting of 80% of the observations, and a test set consisting of the remaining 20%. To avoid any bias, the observations were randomly assigned to the training and test sets.

The primary purpose of this research was to identify seizures in EEG data. Because of this, the response variable in this data set needed to be transformed into a binary value where 1 indicated that a seizure was present in the data and 0 indicated no seizure. The response variable in this set was easily mapped to a binary response variable by converting all response values >1 to 0. In the original data, only a seizure activity value of 1 indicated an actual seizure.

Model training

There are many machine learning algorithms that can be used for classification models. The methods range from simple linear models or nearest neighbor algorithms to more complex neural network models. This study used a range of machine learning algorithms, including the traditional methods of Logistic Regression and Support Vector Machines (SVM) as well as the more complex Long Short-Term Memory (LSTM) method. Each model was fit using Python machine learning libraries and the Orange (v 3.27.1) data mining application.

Logistic regression

Logistic Regression has long been the standard machine learning algorithm for classification models with a binary response variable [12]. Logistic Regression is considered a very simplistic method that like most machine learning algorithms, defines a relationship between the independent and dependent variables and seeks to define a boundary value to divide the classification labels. Logistic Regression assumes a linear relationship between the independent and dependent variables, and it may not be appropriate for data with complex non-linear relationships (Waseem 2020) [13]. Although it is simplistic, Logistic Regression models calculate probability values in addition to a predicted class.

Logistic Regression utilizes a sigmoid function, which is an s-shaped function that can map any real number to a value between 0 and 1 [10]. The graph in figure 6 shows an example of a sigmoid function over one explanatory variable in a Logistic Regression. The logistic function results in a probability value which can be used to classify the observation as either a 0 or 1. Logistic Regression creates a linear combination of the independent variables and like linear regression, the algorithm finds the optimal coefficients for the sigmoid function for each variable.

Support vector machines

Support Vector Machines (SVM) is another simple, yet foundational, machine learning algorithm. The goal of SVM is to find an optimal hyperplane, across the n-dimensional space created by the independent variables that distinctly classifies the data points [14]. SVM can be used for both regression and classification purposes, but it generally performs best as a classifier. In Logistic Regression, the output of a function is mapped on the probability range of [0, 1] and values above a 0.5 threshold are classified as 1 and below 0.5 is classified as a 0. In SVM, if the output of the linear function is greater than 1, then the observation is classified as 1, and if the output is less than -1, then the observation is classified as 0, which results in a margin space on the range of [-1,1] [14].

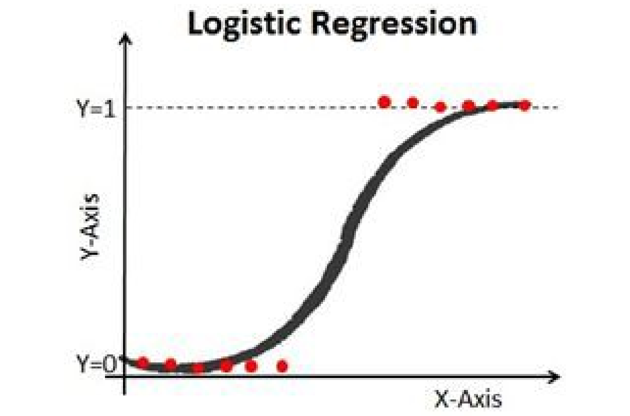

Figure 6: [12]. Example sigmoid function used in Logistic Regression.

The plot in figure 7 illustrates how SVM works. In two dimensions, the data is divided by a line which is defined as the line that maximizes the distance between the nearest observations from different classes. This distance is called the margin, and the data observations that are closest to the line define the margin and are called support vectors. The fact that the algorithm is only dependent on the support vector observations, makes SVM a very computationally efficient algorithm. This makes SVM a preferred method for large data sets with many independent variables. Maximizing the margin distance helps SVM algorithms be robust to new data (Gandhi 2018) [14]. The support vectors are the data points that are most difficult to classify and have the biggest influence on how the hyper plane that divides the data is determined.

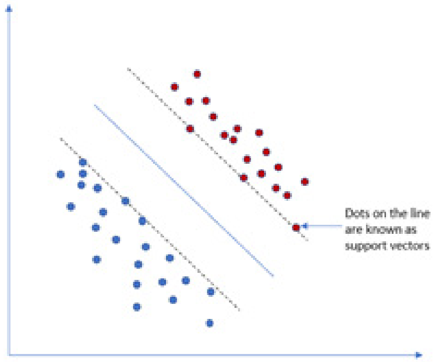

Figure 7: Depiction of how SVM defines hyperplanes to maximize the margin space.

Long short-term memory

Long Short-Term Memory (LSTM) is a machine learning algorithm based on Recurrent Neural Networks (RNNs). Neural networks mimic the human brain. At their simplest form, they take linear combinations of the observations in the input layer data observations and create hidden layers, which then feed forward to the output layer [15]. The nodes in the hidden layers are neurons that are "activated" when they have positive weights and therefore influence the output. The picture in figure 8 shows the basic structure for a neural network. More complex models involve more hidden layers which results in a deeper, deep learning model. RNNs use the state information from the previous layer to inform the activation of the next layer which helps provide context information [16].

LSTM is an RNN that is able to extend the length of the memory at each node. RNNs have a short-term memory in that the current node only has the memory of the information from the previous node. LSTM enables longer term memory by utilizing a series of gates in a memory block that is connected through layers [17]. The diagram in figure 9 shows the basic gate structure for an LSTM model. The gates act like switches to turn the memory block on or off. The memory feature of LSTMs makes them ideally suited for time series data, and like other neural network models, they are able to perform well with complex data relationships [17]. However, the complexity of neural network algorithms makes it difficult to derive the underlying relationships between the input variables and the response variable.

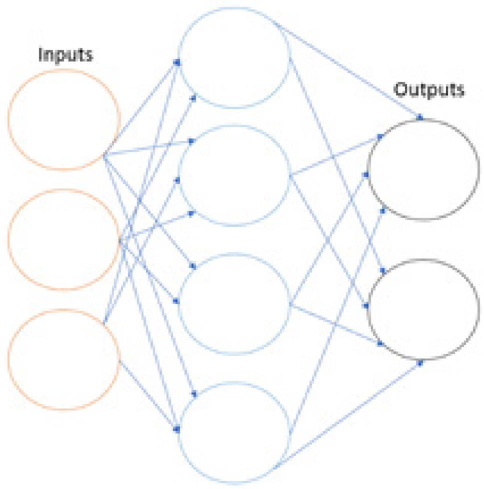

Figure 8: Illustration of a basic neural network structure.

Model evaluation

The final phase in the machine learning process is model evaluation. Once the models have been fitted to the training data, the resulting models can be used to classify the test data. The predicted classification for each model can then be evaluated against the true classification value to determine the overall accuracy of the classification model. In addition to overall accuracy, there are many metrics that can be used to evaluate classifier performance.

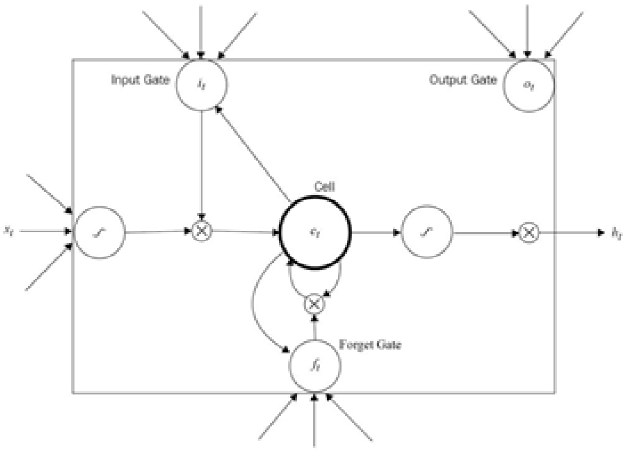

Figure 9: [23], Diagram of LSTM memory gate structure.

A confusion matrix is the most basic model assessment tool. The confusion matrix consists of 4 specific metrics: true positives, false positive, true negatives, and false negatives [8]. True positives are the observations where the model correctly predicted a classification of 1, and false positives are the observations where the model incorrectly predicted a classification of 1. True negatives are the observations where the model correctly predicted a classification of 0, and false negatives are the observations where the model incorrectly predicted a classification of 0. Overall accuracy is calculated using the equation below:

$$Accuracy = (True\ Positive + True\ Negative)/(Total\ Number\ of\ Observations)$$

In most cases, accuracy alone is not a sufficient metric to assess classification performance. Precision and Recall are often used to help assess the real predictive power of the model [18]. Precision measures how well the model performs when it predicts a positive, whereas Recall measures how well the model performs at finding the real positives. Precision and Recall are related to 2 other metrics: Sensitivity and Specificity. Recall is the same as Sensitivity, whereas Specificity is the ratio of True Negatives to the total number of negatives. Specificity is used in studies where classification of negative is more important. In some applications, Precision or Recall may be of higher priority, but in cases where a balance is needed, then the F1 metric is typically used. All of these metrics were used in the evaluation of the three model types. These metrics were calculated using the equations below:

$$Precision = True\ Positives/(True\ Positives + False\ Positives)$$

$$Recall = True\ Positives/(True\ Positives + False\ Negatives)$$

$$F1 = 2 * (Precision * Recall)/(Precision + Recall)$$

Results

Each classification model was fit using cross validation of the training data. Once the best model fit was achieved with the training data, the test data was used to compare and evaluate the methods. The results for training and testing for each method are provided below. The training results in tables 2-4 are shown as a confusion matrix which shows the overall classification results for each method against the training data.

The training methodology for the LSTM model was slightly more complex. The LSTM algorithm with a Dropout value of 0.3 was used for all of our layers, and the activation 'softmax' was used to classify our data. The LSTM model was trained using the 5 category response variable, which was then transformed back to a binary response variable for the test data. LSTM also utilizes epochs to achieve better results and avoid over fitting and obtain an average sampling rate of 37 ms/step. The graph in figure 10 shows the relationship between the validation accuracy and validation loss during our training session. In the graph, as the number of epochs went up the accuracy got better and the loss decreased. This relationship is due to the algorithm trying to reduce the loss amount as much as possible to try to maximize the accuracy. Against the 5 categories in the training data, LSTM achieved an accuracy of approximately 75%.

Table 2: Confusion matrix for Logistic Regression training data.

		Predicted	
		0	1
Actual	0	59%	22%
	1	10%	9%

Table 3: Confusion matrix for SVM training data

		Predicted	
		0	1
Actual	0	76%	5%
	1	1%	18%

Table 4: Confusion matrix for LSTM training data.

		Predicted	
		0	1
Actual	0	79%	1%
	1	1%	19%

The test results for the classification methods are provided in the following tables. Table 5-7 show the confusion matrices for the overall classification results against the test data set. Table 8 provides the resulting evaluation metrics, including the overall Accuracy, Precision, Recall, F1, Sensitivity, and Specificity for each method using both the training and test data sets.

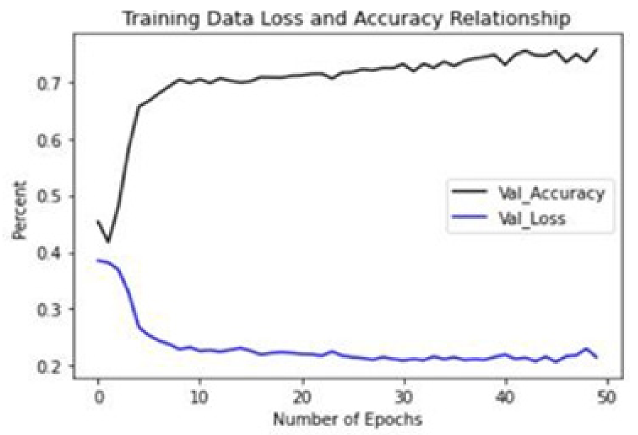

Figure 10: Accuracy vs Loss for training epochs using LSTM.

Table 5: Confusion matrix for Logistic Regression test data.

		Predicted	
		0	1
Actual	0	53%	25%
	1	13%	9%

Table 6: Confusion matrix for SVM test data.

		Predicted	
		0	1
Actual	0	74%	4%
	1	2%	19%

Table 7: Confusion matrix for LSTM test data.

		Predicted	
		0	1
Actual	0	80%	0%
	1	1%	19%

Discussion

For the training data, SVM performed well with an accuracy of 94% and F1 metric of 86%. The Logistic Regression training results were lower with an accuracy of 68% and an F1 of 37%. The results against the test data were similar which indicated that the models were not over-fit to the training data. In our testing data, once again SVM outperformed Logistic Regression with an accuracy of 94%, which is 26% better than Logistic Regression. Across all evaluation metrics, LSTM outperforms SVM and SVM outperforms Logistic Regression in both training and test data results.

A Receiver Operating Characteristic (ROC) curve shows the trade-off between sensitivity and specificity. Classifiers that give curves closer to the top-left corner indicate a better performance. The ROC curve in figure 11 compares the overall performance of the Logistic Regression

Table 8: Evaluation metrics for classification methods.

	Logistic Regression		SVM		LSTM	
	Training	**Test**	**Training**	**Test**	**Training**	**Test**
Accuracy	68%	62%	94%	94%	98%	99%
Precision	30%	27%	80%	82%	95%	98%
Recall	48%	42%	94%	90%	95%	97%
F1	37%	32%	86%	86%	95%	98%
Sensitivity	48%	42%	94%	90%	95%	97%
Specificity	73%	68%	94%	95%	99%	100%

and SVM classifiers. The Logistic Regression model does not perform well, and struggles to provide any meaningful predictive result. SVM clearly outperforms the Logistic Regression model and generates accurate predictions that could be acceptable for automated identification of seizure activity. By comparison, the ROC curve for LSTM is shown in figure 12. Although the results are close, the LSTM curve dominates the curve shown for SVM.

Overall, LSTM outperforms the other classifier models across all metrics. The overall accuracy of 99% and the outstanding F1 metric, proves that LSTM is the best classification method for identifying seizure activity.

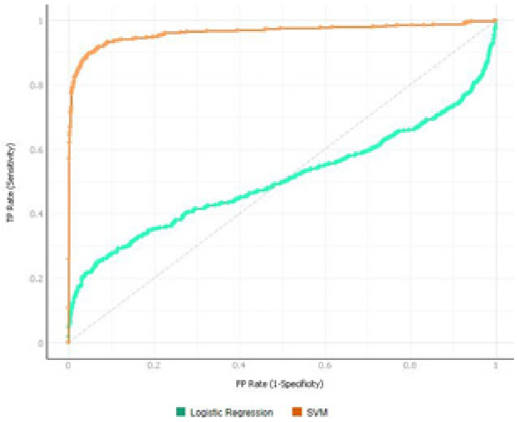

Figure 11: ROC curve comparing Logistic Regression and SVM model performance against the test data.

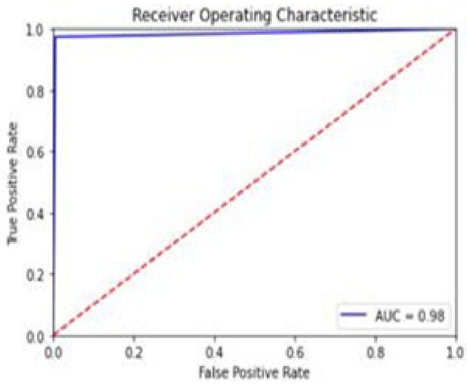

Figure 12: ROC curve showing LSTM performance against the test data.

Our findings are similar to the accuracy obtained by others using LSTM and CNN methods to classify electrographic recordings as seizure or not seizure (Abdelhameed 2018 [19, 20].

Even though the accuracy results for our final testing phase using the LSTM model was 99%, the accuracy when the response included all 5 categories of seizure activity was not as high. The primary purpose of this experiment was simply to identify when a patient had a seizure or not, therefore the use of a binary response and the resulting accuracy are appropriate. In the future, more research should be done to improve the performance of the LSTM classifier across all 5 seizure activity categories. This would further improve the applicability, and credibility, of the model to provide doctors with automated seizure identification that they trust.

Conclusions

Although the SVM model performed well, the long-term memory characteristic of LSTM makes it the best performer across all evaluation metrics for this type of data. Datasets collected in EEG research can be challenging to generate or gain access to. The data for this project was pre-processed, and further work should be done to understand the data collection and transformation process so that this model could be applied to real-time EEG data. Overall, this research has shown the potential to use complex machine learning classification models such as LSTM to accurately identify seizure activity in EEG data. This has the potential to assist healthcare professionals and improve neurological diagnosis and treatment. Besides accelerating the analysis of EEG data such models may also be trained to distinguish between different seizure types and also provide more accurate localization of the seizure onset to potentially enhance surgical outcomes.

References

1. Seizures. [https://www.mayoclinic.org/diseases-conditions/seizure/symptoms-causes/syc-20365711]
2. Epilepsy. [https://www.who.int/news-room/fact-sheets/detail/epilepsy]

3. Claassen J, Mayer SA, Kowalski RG, Emerson RG, Hirsch LJ. 2004. Detection of electrographic seizures with continuous EEG monitoring in critically ill patients. *Neurology* 62(10): 1743-1748. https://doi.org/10.1212/01.wnl.0000125184.88797.62

4. Electroencephalogram (EEG). [https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/electroencephalogram-eeg]

5. Jordan KG. 1993. Continuous EEG and evoked potential monitoring in the neuroscience intensive care unit. *J Clin Neurophysiol* 10(4): 445-475. https://doi.org/10.1097/00004691-199310000-00006

6. Obeid I, Picone J. 2016. The temple university hospital eeg data corpus. *Front Neurosci* 10: 196. https://doi.org/10.3389/fnins.2016.00196

7. Golmohammadi M, Ziyabari S, Shah V, Diego SL, Obeid I, et al. 2017. Deep architectures for automated seizure detection in scalp EEGs. arXiv:1712.09776.

8. Pant A. 2019. Workflow of a machine learning project [https://towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94]

9. Asiri S. 2018. Machine learning classifiers. [https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623].

10. Brownlee J. 2020. What is the difference between test and validation datasets? [https://machinelearningmastery.com/difference-test-validation-datasets/]

11. The Data Detective. 2020. Finally: why we use an 80/20 Split for Training and Test Data Plus an Alternative Method (Oh Yes...). [https://towardsdatascience.com/finally-why-we-use-an-80-20-split-for-training-and-test-data-plus-an-alternative-method-oh-yes-edc77e96295d]

12. Open Data Science. Logistic regression with python. [https://opendatascience.com/logistic-regression-with-python/]

13. Mohammad W. 2020. Classification in machine learning: classification algorithms. [https://www.edureka.co/blog/classification-in-machine-learning/]

14. Gandhi R. 2018. Support vector machine - introduction to machine learning algorithms. [https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47]

15. McCullum N. 2020. Deep learning neural networks explained in plain english. [https://www.freecodecamp.org/news/deep-learning-neural-networks-explained-in-plain-english/]

16. Brownlee J. 2017. A gentle introduction to long short-term memory networks by the experts. [https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/]

17. Gall R. 2020. What Is LSTM? [https://hub.packtpub.com/what-is-lstm/].

18. Brownlee J. 2020. What is the difference between test and validation datasets?" [https://machinelearningmastery.com/difference-test-validation-datasets/].

19. Abdelhameed M, Daoud HG, Bayoumi M. 2018. Deep convolutional bidirectional LSTM recurrent neural network for epileptic seizure detection. 2018 16th IEEE International New Circuits and Systems Conference (NEWCAS). pp 139-143. https://doi.org/10.1109/NEWCAS.2018.8585542

20. Shekokar KS Dour S. 2021. Automatic epileptic seizure detection using LSTM networks. *World Journal of Engineering* inpress. https://doi.org/10.1108/WJE-06-2021-0348

21. Smith EJ. 2022. Introduction to EEG. [https://www.ebme.co.uk/articles/clinical-engineering/introduction-to-eeg]

22. Caraballo RH, Fortini S, Flesler S, Pasteris MC, Portuondo LCE. 2015. Encephalopathy with status epilepticus during sleep: unusual EEG Patterns. *Seizure* 25: 117-125. https://doi.org/10.1016/j.seizure.2014.10.003

23. Stack Overflow. 2017. How are sigmoid/tanh functions forgetting and including information in LSTMs. [https://stackoverflow.com/questions/44751800/how-are-sigmoid-tanh-functions-forgetting-and-including-information-in-lstms]