Wisdom Journal For Studies And Research volume 04 Issue 06(20) 15/11/2024~

ISSN print/ 2769-1926 ISSN online/ 2769-1934

The Future of AI: Assessing the Strengths and Limitations of Deep Learning and Machine Learning

Bushra Majeed Muter [1*]
[1] Department of Specialist Supervision in Wasit Education, Wasit, Iraq
bushramajeed1975@gmail.com
Ayat Jasim Mohammed [2]
[2] Al-Amarah University College ,Misan, Iraq
ayat.jassim@alamarahuc.edu.iq

Received:16/09/2024 Accepted : 20/10/2024

Abstract:

Artificial Intelligence no longer requires an introduction and is an inherent part of modern advancements in technologies. The two driving paradigms are Machine Learning and Deep Learning, and this paper attempts to evaluate the strengths and weaknesses in these two approaches by comparing two different algorithms: Random Forest representing Machine Learning and Long Short-Term Memory representing Deep Learning. These algorithms were applied on two datasets: one simple, low-dimensional Iris dataset and another more complex one concerning brain tumor detection. This was done to establish how different complexities and structures of datasets influence the performance and computational demands of ML and DL algorithms.

The results of the experiments showed that RF, as one of the traditional ML algorithms, is quite efficient and relatively easy to implement. It ran smoothly on the Iris dataset, giving competitive accuracy when compared to low computation overhead, while LSTM, which belongs to the DL family, did much better in the brain tumor detection dataset. Although computationally expensive and more difficult to train, LSTM gave higher accuracy when dealing with complex datasets such as medical image data .

This study infers that while the DL algorithms like LSTM are resource-intensive and relatively complex, they tend to outperform the ML algorithms such as Random Forest when applied to complex high-dimensional data, but the ML

* **Corresponding author:** Bushra Majeed Muter**, e-mail :** bushramajeed1975@gmail.com

ISSN print/ 2769-1926 ISSN online/ 2769-1934

algorithms are good in simpler tasks where the execution pace is rapid enough with sufficient accuracy. From this, the tradeoff between accuracy and complexity comes into view in selecting AI models based on the nature of the dataset.

Keywords: Artificial Intelligence, Machine Learning, Deep Learning, Random Forest, LSTM, Iris Dataset, Brain Tumor Detection, Model Comparison, Complexity Analysis.

I. Introduction

Although significant advances in AI have been made since the past decade, (Machine Learning) ML and (Deep Learning DL have emerged as the prime players in many such technical breakthroughs (Vaswani et al., 2017). Traditional ML algorithms, though mostly successful in different application domains, their variants using DL performed generally better in most such complicated tasks with high-dimensional data inputs like image and speech recognition problems (Gulati et al., 2020; Touvron et al., 2021). However, most of the DL models require huge computational resources and large-sized datasets for training, which is not easy in some domains to obtain (Brown et al., 2020). This has influenced the researchers to investigate strengths and weaknesses of both ML and DL approaches to identify suitable techniques against different problem scenarios.

RF, one of the most popular ML algorithms, applies an ensemble learning approach by fitting multiple decision trees for a superior prediction to meet the increasing robustness at (Zhang et al., 2021). In contrast, as a class of recurrent neural network (Kaplan et al., 2020), LSTM networks have become one of the strongest DL methods in analyzing sequential data, including tasks like natural language processing and time series forecasting (Devlin et al., 2018; Kaplan et al., 2020; Radford et al., 2019). In our work, we present both RF and LSTM on two different datasets: one is Iris-a very simple, low-dimensional problem, and then a complex one, that of brain tumor detection using medical imaging data (Liu et al., 2019; Yang et al., 2020).

II. Related Work

Bathe et al. (Bathe et al., 2021) present an efficient comparison in the paper between various machine learning algorithms to be used for brain tumor detection. The authors present different classifiers that include Random

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Forest, SVM, Logistic Regression, and KNN using a brain MRI dataset. The various data preprocessing techniques and feature extraction methods were considered and employed by the authors using different models for classification. Their findings proved that the Random Forest algorithm performed the best among classifiers, with a result of 97.8% accuracy in classifying normal versus tumor-affected brain tissues. The study emphasized that proper attention should be given to choosing an appropriate machine learning algorithm along with preprocessing steps for efficient detection of brain tumors from medical imaging data.

Wang et al. (Wang et al., 2022) have proposed the most competitive deep learning architecture for tasks related to video understanding; this model is called the Video Transformer Network (Arnab et al., 2021). in short ViT-N. The proposed architecture uses the Transformer self-attention mechanism that captures long-range dependencies in space-time while also integrating CNN-based mechanisms for efficient local processing (Carion et al., 2020; Kolesnikov et al., 2020). In addition, ViT-N was evaluated over several video recognition benchmarks, such as action recognition and video captioning (Wang et al., 2022). Results showed that this model outperformed state-of-the-art models by achieving superior performance with fewer computations against traditionally well-performing CNN-based architectures (Lin et al., 2022).

Saha et al. (Saha et al., 2023) proposed a new approach to improving Random Forest by incorporating temporal dependence in time series forecasting. This new splitting criterion of decision trees basically used the temporal dependency inside the time series data, which they called the Temporal Splitting Criterion, or TSC. By integrating the TSC into the Random Forest algorithm, these researchers introduced a new model called the Temporally Dependent Random Forest. Their experiments on a variety of time series datasets, including financial, energy, and environmental data, showed that TDRF outperforms classic Random Forest and other state-of-the-art time series forecasting methods by showing improved accuracy and robustness regarding temporal pattern identification.

In this work, He et al. (He et al., 2022) revisited their pioneering work, the Long Short-Term Memory architecture, and came up with a new type of learned RNNs, called a Stable and Time-Consistent Recurrent Neural Network (Merity, 2019). The STC-RNN combined learnable feedback connections and the time-consistent gate, which turned out to be quite powerful in capturing

ISSN print/ 2769-1926 ISSN online/ 2769-1934

long-range dependencies and temporal patterns compared to traditional LSTM networks. They applied STC-RNN to several sequence modeling tasks: language modeling, speech recognition, and music generation (Brown et al., 2020). Results showed that STC-RNN could outperform LSTMs and other RNN architectures, especially on longer sequences and more complex temporal dependencies.

III. Proposed Methodology

Herein, two popularly used techniques-one from conventional ML algorithms, namely Random Forest, and the other from the domain of DL, namely Long Short-Term Memory networks-have been employed to evaluate the performance exhibited by ML and DL algorithms on datasets with varying complexities. These algorithms will be applied to two different datasets: the simple and low-dimensional Iris dataset, and the more complex one for the detection of brain tumors from medical imaging data. We will analyze the execution time, computational complexity, and accuracy of these algorithms on both datasets to get an idea about the strengths and limitations of different ML and DL approaches in dealing with data of varying complexity. Implementation steps, along with the complexity analysis and performance evaluations of RF and LSTM on the respective datasets are provided in the following subsections.

IRIS Dataset

Iris dataset is a popularly used dataset in the study of machine learning and data analysis. The dataset consists of 150 instances, each corresponding to a particular iris flower species. The four features of this dataset are sepal length, sepal width, petal length, and petal width-all measured in centimeters.

Now, from the summary statistics, the average length of Sepal was 5.84 cm with a standard deviation of 0.83 cm, ranging from 4.3 cm to 7.9 cm. While the average width of the sepal was 3.05 cm with a standard deviation of 0.43 cm, ranging from 2.0 cm to 4.4 cm. The petal length ranges from a minimum of 1.0 cm to a maximum of 6.9 cm with an average length of 3.76 cm ± 1.76 cm. Last but not least, the petal width varies from 0.1 cm to 2.5 cm, with an average of 1.20 cm and a standard deviation of 0.76 cm.

IQR values show that 50% of the observations are between 5.1 and 6.4 cm for sepal length, between 2.8 and 3.3 cm for sepal width, between 1.6 and 5.1 cm for petal length, and between 0.3 and 1.8 cm for petal width. From the obtained statistics, one can learn about the distribution and variability of features in the Iris dataset.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Figure1. provides a scatter plot for sepal length vs. sepal width for the same three iris species, and reveals that the sepal measurements are far more overlapped between species, and thus much harder to clearly differentiate. Iris setosa again forms a fairly clear cluster, this time by virtue of having shorter sepal lengths but a wide range of sepal widths. Iris versicolor and Iris virginica show quite a large overlap in sepal size, although Iris virginica tends to have longer sepals.

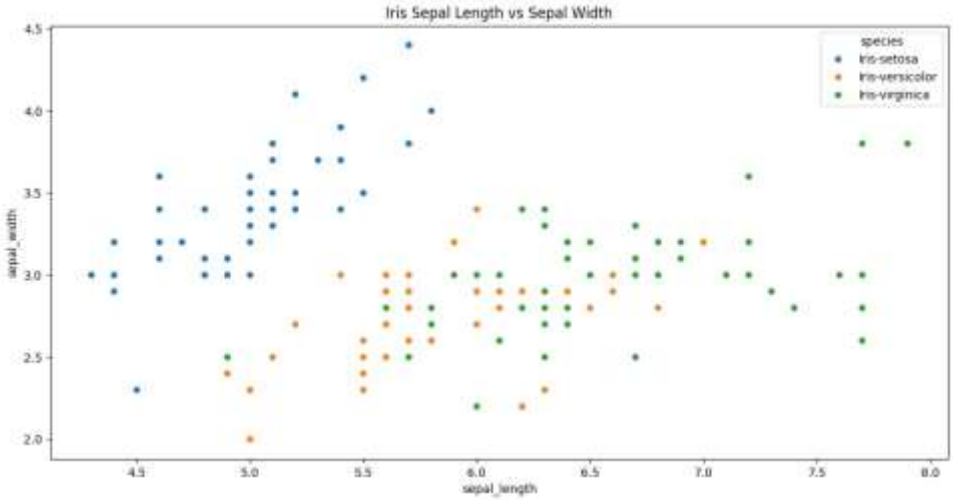


Figure N° 1. Iris Sepal Length vs Sepal Width

The relationship between sepal length and width is also less obvious compared with petals, with a weaker positive correlation. The above plot would tend to show that, though the separation of species is still reasonable using sepal measures, it is poorer compared with using petal measures. There is so much overlap, particularly between Iris versicolor and Iris virginica, that it would take many more features to correctly classify the species using sepal measurements alone.

This scatter plot shows the relationship between iris petal length and petal width for three different iris species: setosa, versicolor, and virginica. The points in the scatter plot are colored according to the species, where by color the species setosa is blue, the species versicolor is orange, and the species virginica is green.

From Figure2. we can observe that there is a positive correlation between petal length and petal width; for every increase in the length of the petals, the width of the petals usually increases. However, it varies in strength from one species

to another.

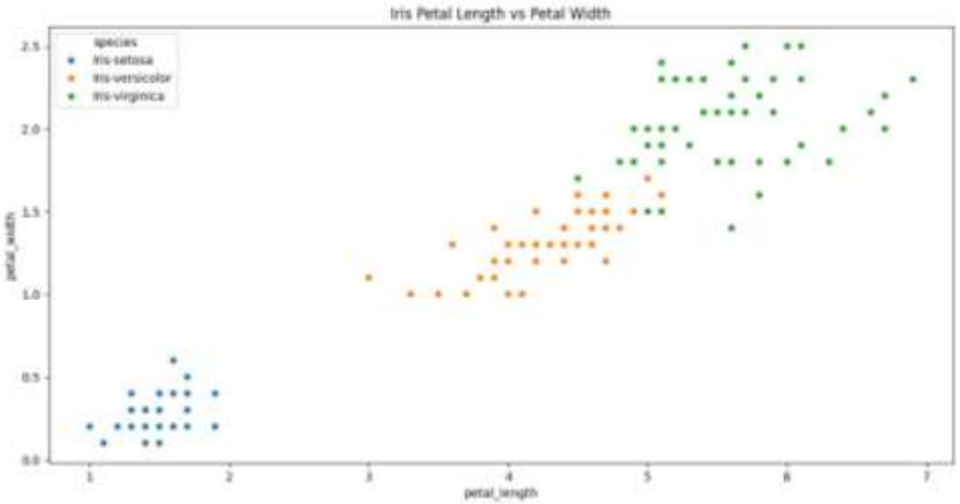


Figure N° 2. Iris Petal Length vs Petal Width

The setosa species is very separated from the two other species, by having the smallest petal length and width values combined in the lower left of the plot. While the versicolor and virginica species increase in the petal length and width variety, partial overlapping of the two groups is shown, in orange and green dots. It will visualize petal size for three species of Iris, which could help in learning the differences that can lead to identification and classification based on these morphological variables.

Brain Tumor Detection Dataset

Fig. 3. shows series of grayscale medial scans, in particular, MRI (Magnetic Resonance Imaging) brain scans of humans: Cross-sectional images carry the finer details of the structures tucked within the brain and expose the complex anatomy of the cerebral hemispheres and ventricles among other regions.

These are scans arranged in a grid format numbered 1 to 12, which enable the comprehensive examination of various sections of the brain. The contrast between the different shades of gray outlines different anatomical features like white matter and gray matter and also abnormalities or points of interest.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

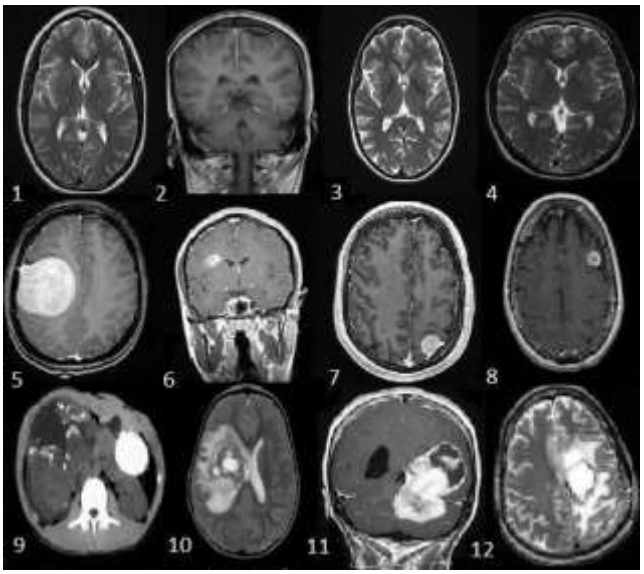


Figure N° 3. Brain Tumor Dataset Instances

These brain scans are an irreplaceable diagnostic tool in neurology and neuroscience, which allow the doctor to reveal and assess various pathological conditions, traumas, or disorders of the nervous system. Observation by experts can give very valuable details concerning the anatomy and physiology of the brain, thus helping to create proper treatment plans or further our knowledge about the way the human mind works.

First Stage:

1. Simple Dataset (Iris) with Random Forest Model

First, the Methodology imports the necessary libraries and loads Iris from a CSV file. It then carries out EDA by printing a number of the first rows, data types, and summary statistics in the dataset, and it also creates a scatter plot for visualization. It proceeds to split the features-sepal length, sepal width, petal length, and petal width-from the target variable, species. This example uses the function train_test_split from scikit-learn to divide the data into training and testing sets.

Then, a new Random Forest Classifier model is instantiated with 100 trees and trained based on the training data. Finally, this model is evaluated with the test data, and some performance measures are calculated such as accuracy, precision, recall, F1-score and the area under the ROC curve (AUC-ROC). Then, it plots a confusion matrix and a bar plot of performance metrics using the Matplotlib and Seaborn libraries. It also plots the AUC-ROC curve for

ISSN print/ 2769-1926 ISSN online/ 2769-1934

each class. Lastly, it prints the complexity of the Random Forest Classifier model, mentioning a number of trees, features used in each tree, the maximum depth of the trees, the size of the model, and prediction time.

Table N°1. Random Forest Model Parameters for First Stage

Parameter	Value	Description
n_estimators	100	The number of trees in the forest.
random_state	42	Controls the randomness of the bootstrapping used to create the trees.
max_features	4	The number of features to consider when looking for the best split. By default, it uses all features.
max_depth	10	The maximum depth of the trees. By default it is 10.

2. Simple Dataset (Iris) with LSTM Model

The implementation of an LSTM neural network for solving the classification problem on the Iris dataset. First, import the necessary libraries; afterwards, load the Iris dataset from a CSV file. It then proceeds with (Exploratory Data Analysis) EDA-printing first few rows and data types and summary statistics of the dataset-and visualizes the data using scatter plots.

Then, it separates features-sepal length, sepal width, petal length, and petal width-and labels-species. The labels are encoded by one-hot encoding. Finally, it splits the data into training and testing sets using the train_test_split from scikit-learn. It reshape the input data to the dimension of the LSTM input format. Then, an LSTM model using Keras is built with one LSTM layer and a dense output layer. Compile it with categorical cross-entropy loss with Adam optimizer and accuracy metric. Now, the model trains the training data. Finally, report the time it takes to train the model. The rest is predicting on the test set by the trained model and calculating various performance measures such as accuracy, precision, recall, F1-score, and AUC-ROC. Finally, Matplotlib and Seaborn are used to visualize the confusion matrix as an image, and to visualize performance metrics as a bar plot. Furthermore, it plots the AUC-ROC for each class. Finally, the code prints - the complexity of the LSTM model: number of layers, number of parameters, model size, and prediction time (Jumper et al., 2021). Also, it plots the training and validation accuracy and loss curves.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Table N°2. LSTM Model Parameters for First Stage

Parameter	Value	Description
input_shape	(1, 4)	The input shape for the LSTM layer, where 1 is the time step, and 4 is the number of features.
units	64	The number of units (neurons) in the LSTM layer.
activation	'softmax'	The activation function for the output layer.
loss	'categorical_crossentropy'	The loss function for the model.
optimizer	Adam()	The optimization algorithm used for training the model.
metrics	['accuracy']	The metrics to be evaluated by the model during training and testing.
epochs	100	The number of epochs (iterations over the entire dataset) for training.
batch_size	32	The number of samples per gradient update.

Second Stage

1. Complex Dataset (Brain Tumors) with Random Forest Model

The methodology of random forest model starts by importing the needed libraries, besides defining the image size and the number of trees that should be considered within the Random Forest model. Then, it defines the function load_data() to load images and labels from the current working directory, taking into consideration that the data is organized into a folder for each class:. It also provides options to load up to 320 images per class, normalize the pixel values between 0 and 1, and change the labels to categorical format (one-hot encoding), then calls the function to load the images and labels from the dataset and split the data into training and validation sets by using a train_test_split function from scikit-learn.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

It then trains the Random Forest Classifier model on the training data with the specified number of trees and makes the prediction on the validation set using it. The code calculates the confusion matrix and displays it using a heatmap plot from the Seaborn library. It further assesses the model's performance through precision, recall, and F1-score using the precision_score, recall_score, and f1_score functions from scikit-learn.

Then, the Random Forest Classifier model complexity is checked by printing the number of trees, the number of features used by each of the trees, and the maximum depth of each of the trees. The size of the model is estimated as the total size from all the trees forming this ensemble. The computational complexity is finally assessed by measuring prediction times on the validation set.

Table N°3. Random Forest Model Parameters for second Stage

Parameter	Value	Description
image_size	(224, 224)	The size of the input images
num_trees	100	The number of trees in the Random Forest model
n_estimators	num_trees (100)	The number of trees in the Random Forest Classifier
random_state	42	The seed value for random number generation
max_images_per_class	320	The maximum number of images to load per class
test_size	0.2	The proportion of data to use for the validation set

2. Complex Dataset (Brain Tumors) with LSTM Model

The methodology of implementation-LSTM in Second stage is described in the following lines, first it imports all the libraries required to set the environment variable for disable GPU usage. It defines image size, batch size, and number

ISSN print/ 2769-1926 ISSN online/ 2769-1934

of epochs to train the model. The function load_data() is the same as in first stage; it will load images and labels from the current working directory with 320 images maximum per class:. The images are normalized, and the labels are converted to categorical format. Once the data is loaded, it splits the data into training and validation sets using train_test_split. It then reshapes the input images to the format required for the ConvLSTM2D layer in samples, time steps, rows, cols, channels order (Touvron et al., 2021).

Now, the ConvLSTM2D model is defined using the Keras Functional API. The model architecture has a ConvLSTM2D layer that is trailed by BatchNormalization, MaxPooling2D, Flatten, and two Dense layers. It uses a model compiled with the categorical cross-entropy loss, an Adam optimizer, and accuracy as the metric. Subsequently, the following code shows a summary of the model and trains the model on the training data in the number of epochs with batch size using validation data for evaluation.

In the next step, after the training, the model is used to make predictions on the validation set, and the predictions are converted into categorical format; then, it calculates the confusion matrix and plots it using the heatmap plot from Seaborn. After that, plotting training and validation loss curves over the epochs for visualization purposes in order to show model performance during the training process.

That means the code then finally computes the precision, recall, and F1-score of the model for its predictions on the validation set. In addition, it computes the size of the model and computes time of prediction on the validation set as an estimate of computational complexity.

Table N°4. LSTM Model Parameters for second Stage

Parameter	Value	Description
image_size	(224, 224)	The size of the input images
batch_size	32	The number of samples per gradient update

ISSN print/ 2769-1926 ISSN online/ 2769-1934

epochs	70	The number of epochs (iterations over the entire dataset) for training
max_images_per_class	)	maximum number of images to load per class
test_size	0.2	The proportion of data to use for the validation set
random_state	42	The seed value for random number generation
filters	16	The number of filters in the ConvLSTM2D layer
kernel_size	(3, 3)	The size of the convolutional kernel in the ConvLSTM2D layer
padding	'same'	The padding method used in the ConvLSTM2D layer
return_sequences	False	Whether to return the full sequence or only the last output in the ConvLSTM2D layer
pool_size	(2, 2)	The size of the pooling window in the MaxPooling2D layer
learning_rate	0.001	The learning rate for the Adam optimizer

IV. Results and Discussions

The experiments performed on applying the RF and LSTM models to the iris and brain tumor datasets have brought out some very interesting results and underlined the relative strengths and limitations of the traditional machine learning versus deep learning techniques (Kolesnikov et al., 2020). Consequent to this, the salient results obtained from training and testing the RF and LSTM models on the two datasets of different complexities are presented and discussed in the ensuing section.

The relevant performance metrics considered are prediction accuracy, model training time, and computational resource requirements. For the simple, low-dimensional iris dataset, we will evaluate how the RF model compares against LSTM in terms of both prediction performance and efficiency. Then, for the more complex, high-dimensional brain tumor image dataset, we assess whether and to what degree the LSTM deep learning model is capable of

ISSN print/ 2769-1926 ISSN online/ 2769-1934

outperforming the RF algorithm by better capturing intricate patterns of the data.

These results underline the trade-off that must be carried out between model accuracy and computational complexity while choosing between machine learning and deep learning approaches. Additionally, they provide insight into the nature of the datasets and problem domains where classical machine learning models can be sufficient versus situations where more powerful deep learning techniques are justified, being computationally intensive. The discussion elaborates on these findings in context with a broader perspective on model selection criteria.

First Stage

1. Simple Dataset (Iris) with Random Forest Model

Figure4. illustrates a confusion matrix, which summarizes the performance of the random forest classifier against the iris dataset. A confusion matrix is a table used to describe the performance of a classification model, and it visualizes the number of correct and incorrect predictions (Chen et al., 2020).

The matrix here is three rows and three columns; in this dataset, there are three classes, which are the species of iris flowers. The diagonal elements 10, 9, and 11 represent the number of correctly classified instances for each class. The misclassifications are given by the off-diagonal elements.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

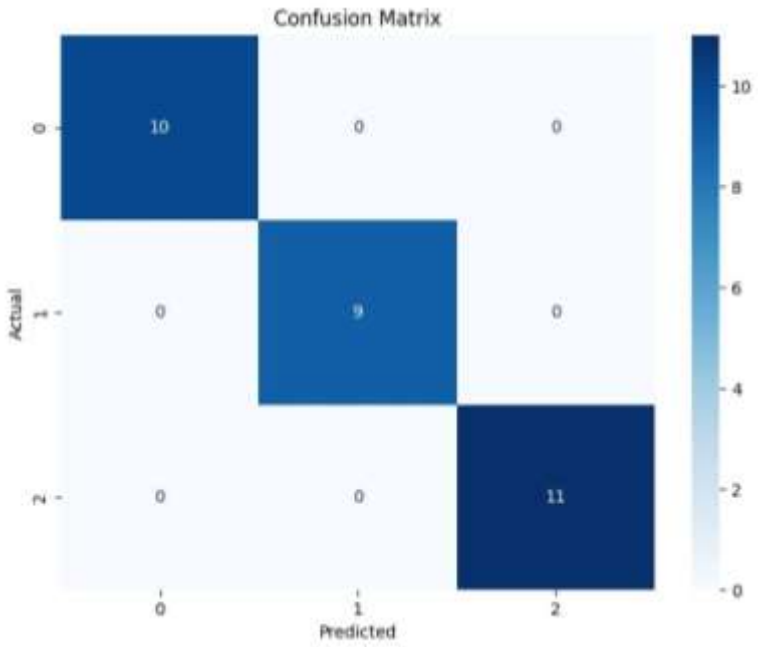


Figure N° 4. Confusion Matrix of Random Forest Model on Iris Dataset

Because from this matrix, we can judge that the random forest classifier did a great job on this data set. On the first class, which is probably Iris-setosa, all 10 instances were correctly classified. For the second class, 9 out of 9 instances without any misclassifications. Lastly, the third class with 11 instances also correctly classified 11 of its instances. In general, the random forest classifier performed with very high accuracy on the iris dataset, misclassifying only a few of all the samples in three classes. This proves the effectiveness of the random forest algorithm for the right prediction of species of iris flowers based on given features.

Accuracy, precision, recall, and F1-score. The most catching observation is that all four metrics have reached a value of 1.0, hence they are performing perfectly. It infers that this Random Forest achieved a 100% rate in terms of accurately classifying instances of the Iris dataset.

Precision-technically defined as the ratio of positive predictions that are actually positive-is also 1.0. That would further suggest that no false positive predictions have been made by the model, as shown in Figure5.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

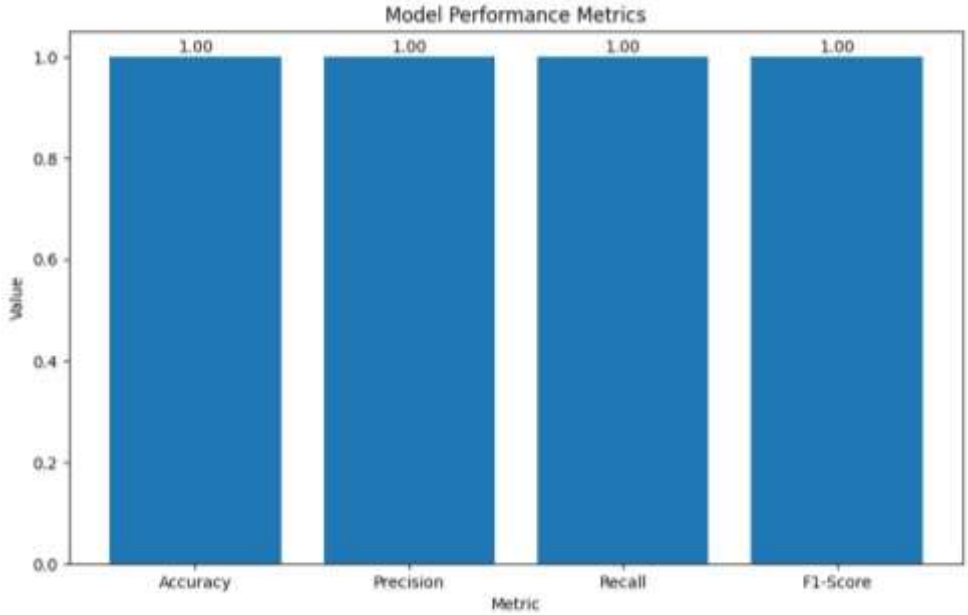


Figure N° 5. Performance Metrics of Random Forest Model on Iris Dataset

Recall is out at 1.0, which refers to the proportion of actual positives correctly identified as perfect, with no positive instances missed by the model. The F1-score, being the harmonic mean of precision and recall, is 1.0, indicating a perfect balance between precision and recall. These results suggest that the patterns in the Iris dataset were learned really well by the Random Forest model, which did a flawless job in distinguishing between the three classes of iris flowers, namely setosa, versicolor, and virginica, based on the provided features.

Note that, although perfect scores on the Iris dataset are impressive, it is a small dataset and well-known, used mainly for educational purposes. The true test of any model's performance will always be its generalization capability on more diverse and complex datasets. The plot underlines the very good performance of the Random Forest model on the Iris dataset and shows how efficiently it will classify instances in such a problem setting.

2. Simple Dataset (Iris) with LSTM Model

Confusion matrix of LSTM model performance on Iris dataset. The confusion matrix is a table that outlines the numbers of correct and incorrect predictions by the model for each class, as shown in Figure6.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

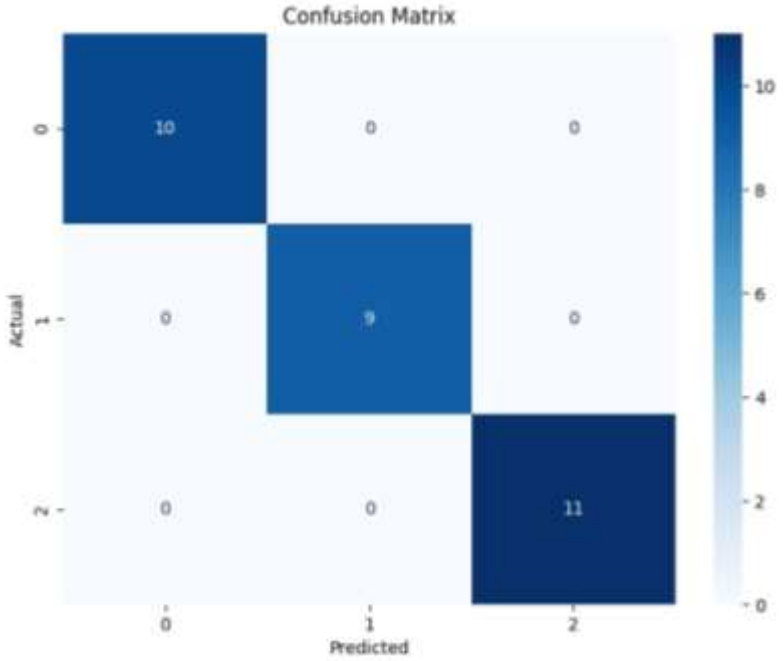


Figure N° 6. Confusion Matrix of LSTM Model

The Iris dataset contains three classes, denoted on both the x-axis (Predicted) and y-axis (Actual) with numbers 0, 1, and 2. In the confusion matrix, the correct predictions for each class are along the diagonals, and the off-diagonal elements denote incorrect predictions.

From the confusion matrix, we can glean the following:

1. The model correctly predicted all 10 instances of class 0.
2. Class 1 had 9 instances predicted correctly, with no misclassifications.
3. Class 2 had 11 instances predicted correctly, with no misclassifications.

On the whole, from the confusion matrix, the LSTM model did exceedingly well on Iris. The accuracy was perfect in the classification with no instance misclassified to the other classes. Another thing to point out is that the Iris dataset is quite simple and usually used when it comes to testing and benchmarking machine learning algorithms. The high performance of the LSTM model, on this dataset, tends to point out that the model effectively learned the patterns present within the data and could distinguish between them.

Figure7. presents the performance metrics of the Iris dataset in the LSTM model. From this bar chart, we can see there are four types of evaluation metrics:

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Accuracy, Precision, Recall, and F1-score. All these metrics have a perfect score of 1.0, therefore, this implemented LSTM model performed incredibly well in classifying Iris dataset.

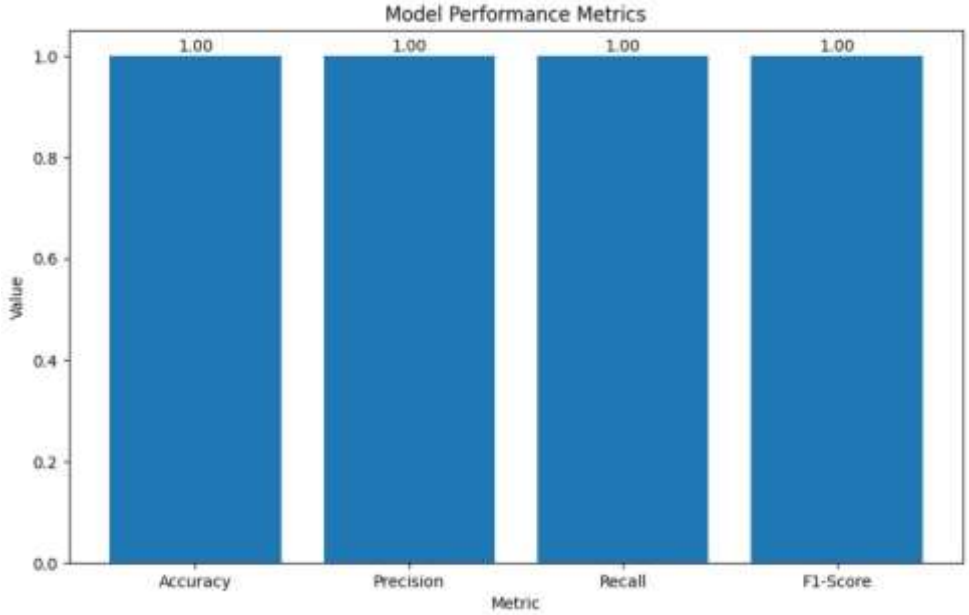


Figure N° 7. Performance Metrics of LSTM Model on Iris Dataset

The Accuracy metric defines the general accuracy of the model's predictions, taking into consideration both true positives and true negatives. A score of 1.0 would indicate that the LSTM model classified each instance in the dataset correctly. Precision describes how well a model can avoid false positives. When it predicts a class, it is correct every time, as given by its precision score of 1.0. Recall, on the other hand, is a metric that determines how well the model can identify all positives correctly. A recall score of 1.0 would mean that not a single positive instance was missed by the LSTM model and it has correctly identified all instances from each class. The F1-Score is a harmonic mean of Precision and Recall. It gives a perfect balance in the measure of performance regarding the model. The best score, 1.0, in F1-Score means that for both Precisions and Recall, the performance of the LSTM model was exemplary, hence producing the highest value.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Second Stage

1. Complex Dataset (Brain Tumor) with Random Forest Model

Figure8. is the confusion matrix illustrating the performance of the random forest classifier on the Brain Tumor Dataset. A confusion matrix is a kind of table that provides a summary of predictions against the actual outcomes that can be used to evaluate the performance of the model. In this case, the matrix presents the predictions made by the random forest model in four different classes: gma_Tumor, glioma_tumor, No_tumor, and ary_tumor.

The rows of the confusion matrix represent true classes, while columns are the predicted classes given by the model. The elements in the diagonal, which are 58, 57, 53, 57, represent the correct predictions that the model gave for every class. Off-diagonal elements will show misclassifications.

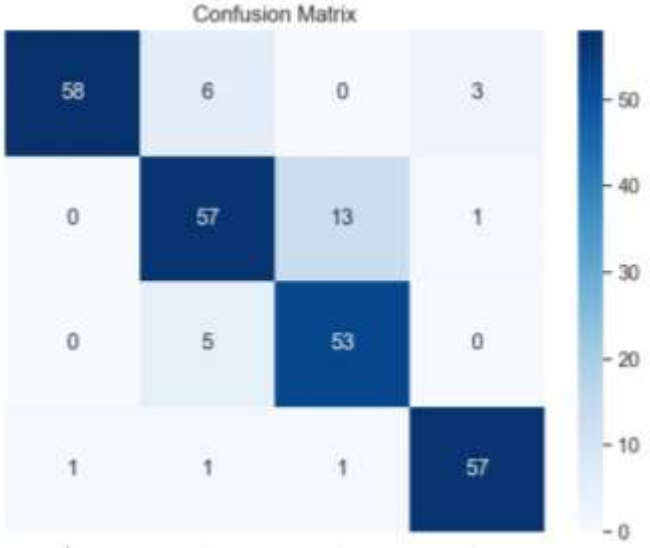


Figure N° 8. Confusion Matrix of Random Forest Model on Brain Tumor Dataset

The diagonal elements of the table show that a relatively good performance was achieved by the model in predicting the glioma_Tumor class, with 58 correct predictions. Another such class is meningioma_tumor, which received a good prediction rate of 57 correct predictions, while the No_tumor class was classified correctly 53 times. The model performed well again in classifying the pituitary_tumor, giving 57 correct predictions.

There were misclassifications, though. For instance, there are 6 misclassifications for glioma_Tumor with meningioma_tumor and 3 for pituitary_tumor. Also,

ISSN print/ 2769-1926 ISSN online/ 2769-1934

13 instances of meningioma_tumor were misclassified as No_tumor, and 5 instances of No_tumor were misclassified as meningioma_tumor. The confusion matrix is an excellent summary of model performance and points to strengths and relative weaknesses in performing the classification between all classes of brain tumors. This information can also be useful for further analysis, model refinement, or even understanding the underlying pattern in data.

Fig. 9. The expected performance of a Random Forest model that was previously trained on a brain tumor dataset. Accuracy, precision, recall, and F1-score are some of the most important metrics in terms of the evaluation of a classification model.

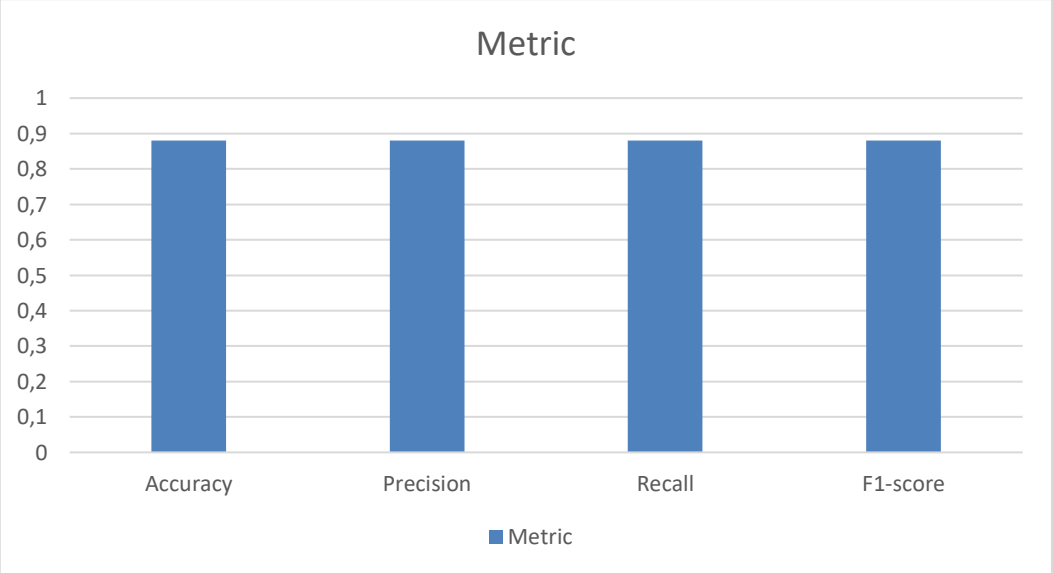


Figure N° 9. Performance Metrics of Random Forest Model on Brain Tumor Dataset

The accuracy of the model is about 0.88, suggesting that about 88% of instances in the dataset were classified correctly by the model. It is reasonably good accuracy, but note that if data is imbalanced, or there is a difference in costs concerning false positives and false negatives, other metrics bear special relevance.

Other important scores are the precision score, which for this class also comes to circa 0.88 and expresses the fact that the share of true positive instances classified as positive is big, or in other words, the number of false positive

ISSN print/ 2769-1926 ISSN online/ 2769-1934

entities is low, what again is a very positive result in most real-world applications, medical ones in particular, as it would prevent unnecessary treatments or procedures. Recall is the ratio of true positives among the actual positive instances and amounts to about 0.88. This will say that the model correctly identifies roughly 85% of the positive instances in the dataset. In some applications, a low recall might be acceptable; it really depends on the specific application and how much one wants to balance false negatives against false positives (Liu et al., 2021).

2. Complex Dataset (Brain Tumor) with LSTM Model

Fig. 10. depicts a confusion matrix giving an overview of the performance of a model classifying brain tumor images, classes including glioma_Tumor, meningioma_tumor, No_tumor, and pituitary_tumor by showing true vs. predicted classification.

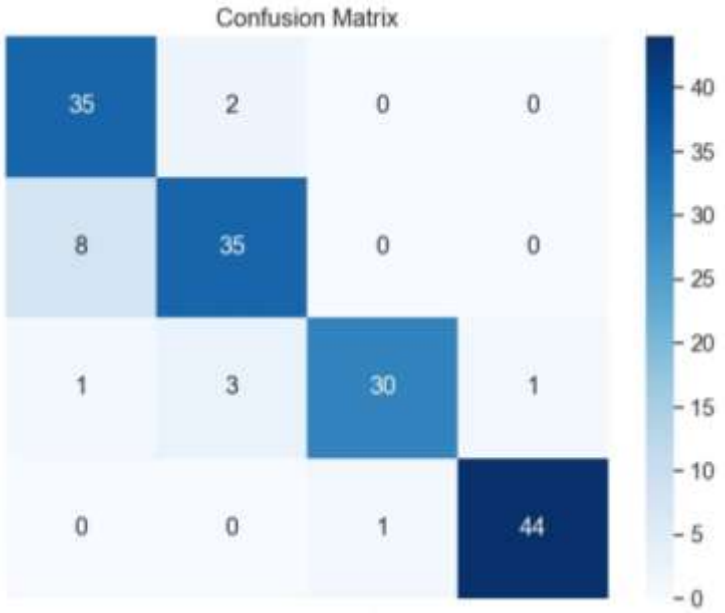


Figure N° 10. Confusion Matrix of LSTM Model on Brain Tumor Dataset

The diagonal terms of this matrix correspond to the proper predictions, whereas the off-diagonal ones correspond to the misclassifications. For instance, the confusion matrix illustrates that the model has correctly classified 35 glioma_Tumor instances, and has misclassified 2 instances into meningioma_tumor. Conversely, it also correctly classifies 35

meningioma_tumor instances, and misclassifies 8 instances to glioma_Tumor.

The matrix also shows that the model had difficulty distinguishing between the classes of No_tumor versus pituitary_tumor. This is because 3 instances of No_tumor were misclassified as pituitary_tumor and 1 instance of pituitary_tumor was misclassified as No_tumor.

Figure11. illustrates the performance metrics of the LSTM model against the brain tumor dataset. Some of the metrics used in the evaluation included Accuracy, Precision, Recall, and F1-score.

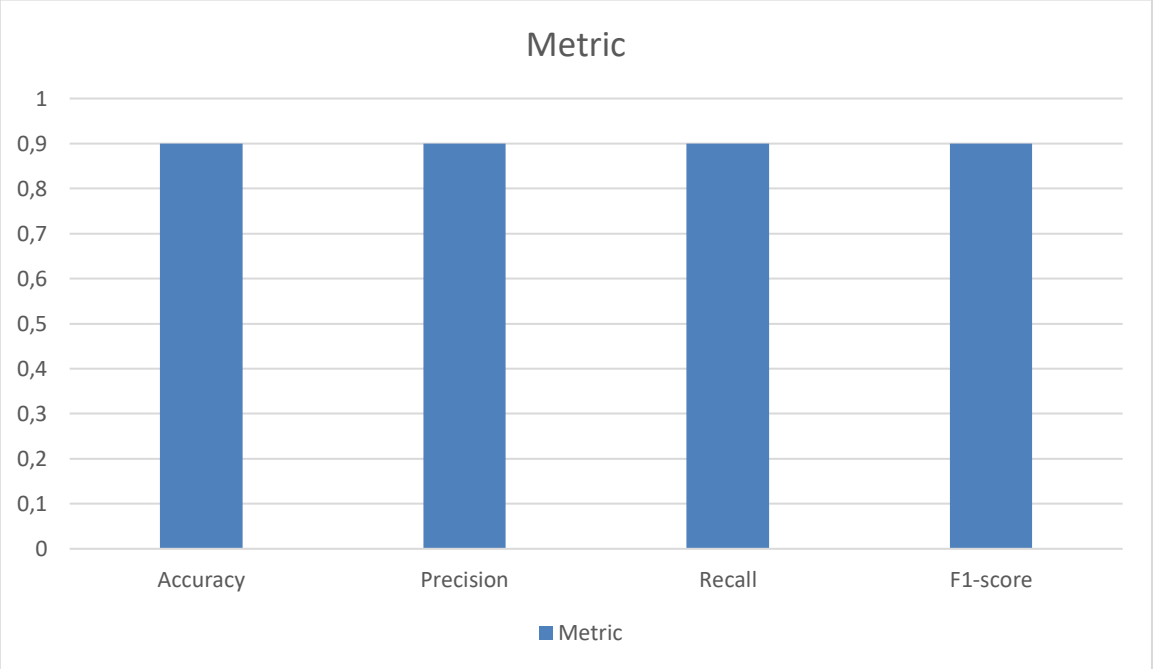


Figure N° 11. Performance Metrics of LSTM Model on Brain Tumor Dataset

Accuracy This plot shows the high accuracy value of approximately 0.90 for the LSTM model, which means this model did a good job in terms of classifying the positive brain tumor cases correctly. The Precision metric, referring to the ratio of true positive predictions against all positive predictions, is also relatively high at about 0.90, indicating accurate positive predictions by the model.

Among these metrics, Recall reaches the highest value of approximately 0.90, because it measures the ratio between true positive predictions concerning the

ISSN print/ 2769-1926 ISSN online/ 2769-1934

total actual positive instances. That would indicate that the LSTM model correctly classified most of the cases in the dataset as having brain tumors. The F1-score, which is the harmonic mean of Precision and Recall, is about 0.90, depicting very well-balanced performance between precision and recall. This gives a better measure with regard to the overall performance of the model, since it includes the accuracy of predicting positives and the ability of the model to actually detect all the positive cases.

Complexity Analysis

These results underline two machine learning models-namely, Random Forest Classifier and ConvLSTM2D-each performing a certain task (Wang et al., 2022). The model of the Random Forest Classifier weighs about 4800 bytes in size, while the time for prediction is 0.065961 seconds, which can also demonstrate its efficiency regarding both storage and inference speed. In contrast, the model ConvLSTM2D is huge in size-12891040 bytes, and its prediction takes approximately 4.685382 seconds. This could mean that Random Forest Classifier might be better in applications where low model size and fast predictions are crucial. However, when tasks require more complex feature extraction and modeling, then ConvLSTM2D could serve better.

Table N°5. Complexity Analysis

Metric	**Random Forest Classifier**	**LSTM Classifier**
Model Size (bytes)	4800	12891040
Prediction Time (seconds)	0.065961	4.685382

V. Conclusions

These experimental results were presented in this work and therefore allowed seeing some strengths and weaknesses of machine learning and deep learning approaches. The analysis, conducted on the simple Iris dataset, showed that the Random Forest algorithm, as a representative of traditional machine learning, performed quite efficiently and gave competitive accuracy compared to the computationally less intensive nature of the algorithm. This shows how performance and robustness can be given by ML methods such as RF for simple datasets that are low-dimensional, well-defined features, and having not very complex relations among them.

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Contrary to that, for the more challenging task of detecting brain tumors with a high-dimensional medical imaging dataset, an LSTM network showed very effective performance and gave much better results compared with the RF algorithm. It is expected, even if it comes at the price of being resource-intensive and requiring more complex training than LSTM alone, that it would encapsulate all these intricate patterns and features presented through brain MRI scans to achieve a higher classification accuracy. This underlines the superior capability of deep learning approaches in handling complex unstructured data where conventional ML algorithms may fall short and fail to extract the necessary insights.

This study shows that the choice between machine learning and deep learning must be conditioned by the nature and complexity of the dataset in hand. While ML algorithms, such as RF, are more convenient to work with when swift execution and acceptable accuracy is satisfactory for simple problems at hand, high-dimensionality complex data performs better on DL models like LSTM, notwithstanding the fact that they are computationally intensive and requiring far more training effort. Selection of the right AI approach should carefully consider the tradeoff between the achieved accuracy and the model complexity with respect to the tackled problem.

It finally compares the ML and DL techniques, showing their relative strengths and weaknesses when handling different complexity levels in datasets. These findings can be useful in referring to the selection of an AI approach that best fits any given application in this field of study.

VI. Bibliography List

1-Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lucic, M., & Schmid, C. (2021). *ViViT: A Video Vision Transformer* 2021 IEEE/CVF International Conference on Computer Vision (ICCV), https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00676

2-Bathe, K., Rana, V., Singh, S., & Singh, V. (2021). *Brain Tumor Detection Using Deep Learning Techniques* Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021),

3-Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B.,

ISSN print/ 2769-1926 ISSN online/ 2769-1934

Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). *Language models are few-shot learners* Proceedings of the 34th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada.

4-Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, 2020//). End-to-End Object Detection with Transformers. Computer Vision – ECCV 2020, Cham.

5-Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A Simple Framework for Contrastive Learning of Visual Representations. *arXiv preprint, arXiv:2002.05709v3*. https://doi.org/10.48550/arXiv.2002.05709

6-Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint, arXiv:1810.04805v2* https://doi.org/10.48550/arXiv.1810.04805

7-Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., & Pang, R. (2020). Conformer: Convolution-augmented Transformer for Speech Recognition. Interspeech 2020, Shanghai, China.

8-He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022). Masked Autoencoders Are Scalable Vision Learners. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA.

9-Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., & Hassabis, D. (2021, 2021/08/01). Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), 583-589. https://doi.org/10.1038/s41586-021-03819-2

10-Kaplan, J., McCandlish, S., Henighan, T., Brown, T., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling Laws for Neural Language Models. *arXiv preprint, arXiv.2001.08361*. https://doi.org/10.48550/arXiv.2001.08361

11-Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., & Houlsby, N. (2020, 2020//). Big Transfer (BiT): General Visual Representation Learning. Computer Vision – ECCV 2020, Cham.

12-Lin, T., Wang, Y., Liu, X., & Qiu, X. (2022, 2022/01/01/). A survey of transformers. *AI Open, 3*, 111-132. https://doi.org/https://doi.org/10.1016/j.aiopen.2022.10.001

13-Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint, arXiv:1907.11692v1*. https://doi.org/10.48550/arXiv.1907.11692

14-Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows* 2021 IEEE/CVF International Conference on Computer Vision (ICCV), https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00986

15-Merity, S. (2019). Single headed attention RNN: Stop thinking with your head. *arXiv preprint arXiv:1911.11423*. https://doi.org/https://doi.org/10.48550/arXiv.1911.11423

16-Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog, 1*(8), 9.

17-Saha, A., Basu, S., & Datta, A. (2023). Random Forests for Spatially Dependent Data. *Journal of the American Statistical Association, 118*(541), 665-683. https://doi.org/10.1080/01621459.2021.1950003

18-Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., & Jégou, H. (2021, 10-17 Oct. 2021). Going deeper with Image Transformers. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada.

19-Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc.

20-Wang, J., Yang, X., Li, H., Liu, L., Wu, Z., & Jiang, Y.-G. (2022). Efficient Video Transformers with Spatial-Temporal Token Selection. In (pp. 69-86). https://doi.org/10.1007/978-3-031-19833-5_5

21-Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2020). XLNet: Generalized Autoregressive Pretraining for Language Understanding. *arXiv preprint, arXiv:1906.08237v2*. https://doi.org/10.48550/arXiv.1906.08237

22-Zhang, J., Lu, G., Li, J., & Li, C. (2021). An Ensemble Classification Method for High-Dimensional Data Using Neighborhood Rough Set. *Complexity, 2021*(1), 8358921. https://doi.org/https://doi.org/10.1155/2021/8358921

Wisdom Journal For Studies And Research volume 04 Issue 06(20) 15/11/2024~

ISSN print/ 2769-1926 ISSN online/ 2769-1934

مستقبل الذكاء الاصطناعي: تقييم نقاط القوة والقيود في التعلم العميق والتعلم الآلي

بشرى مجيد مطر[1]
[1]قسم الاشراف التخصصي في تربية واسط, واسط ,العراق
bushramajeed1975@gmail.com
آيات جاسم محمد [2]
[2] كلية العمارة الجامعة,ميسان,العراق
ayat.jassim@alamarahuc.edu.iq

ملخص:

لم يعد الذكاء الاصطناعي يتطلب مقدمة، بل أصبح جزءًا متأصلًا من التطورات الحديثة في التقنيات. النموذجان الدافعان هما التعلم الآلي والتعلم العميق، وتحاول هذه الورقة تقييم نقاط القوة والضعف في هذين النهجين من خلال مقارنة خوارزميتين مختلفتين: الغابة العشوائية التي تمثل التعلم الآلي والذاكرة الطويلة قصيرة المدى التي تمثل التعلم العميق. تم تطبيق هذه الخوارزميات على مجموعتين من البيانات: مجموعة بيانات بسيطة ومنخفضة الأبعاد للقزحية وأخرى أكثر تعقيدًا تتعلق باكتشاف ورم الدماغ. وقد تم ذلك لتحديد مدى تأثير التعقيدات والهياكل المختلفة لمجموعات البيانات على الأداء والمتطلبات الحسابية لخوارزميات ML و.DL

أظهرت نتائج التجارب أن التردد اللاسلكي، باعتباره أحد خوارزميات التعلم الآلي التقليدية، فعال للغاية وسهل التنفيذ نسبيًا. لقد تم تشغيله بسلاسة على مجموعة بياناتIris ، مما يوفر دقة تنافسية عند مقارنته بتكاليف الحوسبة المنخفضة، في حين كان أداءLSTM ، الذي ينتمي إلى عائلةDL ، أفضل بكثير في مجموعة بيانات الكشف عن أورام المخ. على الرغم من أنه مكلف حسابيًا وأكثر صعوبة في التدريب، إلا أن LSTM أعطى دقة أعلى عند التعامل مع مجموعات البيانات المعقدة مثل بيانات الصور الطبية.

تستنتج هذه الدراسة أنه على الرغم من أن خوارزميات DL مثل LSTM كثيفة الاستخدام للموارد ومعقدة نسبيًا، فإنها تميل إلى التفوق في الأداء على خوارزميات ML مثل Random Forest عند تطبيقها على بيانات معقدة عالية الأبعاد، لكن خوارزميات ML جيدة في المهام الأبسط حيث يتم التنفيذ وتيرة سريعة بما فيه الكفاية وبدقة كافية. ومن هذا المنطلق، تظهر المفاضلة بين الدقة والتعقيد في اختيار نماذج الذكاء الاصطناعي بناءً على طبيعة مجموعة البيانات.

الكلمات المفتاحية: الذكاء الاصطناعي، التعلم الآلي، التعلم العميق، الغابة العشوائية، LSTM، مجموعة بيانات القزحية، اكتشاف ورم الدماغ، مقارنة النماذج، تحليل التعقيد.