Comparison of Predictive Models for Rainfall Prediction using Big Data Technologies

Abdullah McDonald
Department of Computer Science
University of the Western Cape
3708949@myuwc.ac.za

Keelan Govender
Department of Computer Science
University of the Western Cape
3770037@myuwc.ac.za

Unathi Ntame
Department of Information Systems
University of the Western Cape
3821919@myuwc.ac.za

***Abstract*—Rainfall is a form of precipitation and is responsible for providing most of the freshwater for animals and plants. Machine learning can be used to analyze data trends to develop a model. Deep learning on the other hand focuses more on using images specifically to analyze data. Trying to understand the patterns of rainfall to predict it has proven to be a difficult undertaking, as seen by the various research using machine learning and deep learning for this problem. When implementing a solution to this rainfall prediction problem, a vast amount of computational resources are usually required to execute it. Thus arises a need to properly store and analyze the data to effectively approach the prediction aspect. This paper investigated the comparison of predictive models for rainfall prediction using big data technologies and radar rainfall images. The literature on state-of-the-art prediction models was investigated and compared to survey which models could achieve satisfactory prediction results in combination with big data technologies. The models chosen were Random Forest Regressor and Deep LSTM and were used to predict 1,2, and 3 days ahead using monthly rainfall data. Results from this study showed that the Deep LSTM model performed better than the Random Forest Model for sequence lengths of 4, 8, and 12 when predicting 1, 2, and 3 months ahead.**

***Index Terms*—Big Data Applications, Rainfall Prediction, Random Forest, Machine Learning, Long-Term-Short-Term-Memory**

I. Introduction

Rainfall is a form of precipitation and is responsible for providing most of the freshwater for animals and plants [1]. Trying to understand the patterns of rainfall to predict has proven to be a difficult undertaking, as seen by the various research on it. This is in part due to the unique behavior that rainfall exhibits that is not prevalent in other time-series data. Nevertheless, it is important to be able to forecast rainfall as it can be a major tool in water management to lessen the effect of natural disasters [2], as well helping understand and plan the agricultural development of countries. Predicting rainfall is also important since it can provide weather guidance for airports, manage floods, transportation, agriculture and manage the daily lives of people [3] [4].

As mentioned before, rainfall is very dynamic in nature. Now since climate change can transform the patterns of rainfall data, it becomes increasingly difficult to predict rainfall [5]. Rainfall predictions require high accuracy and a high spatiotemporal resolution to achieve meaningful and trustworthy results and also prevent the types of events mentioned above [3]. Machine learning and deep learning techniques can provide these requirements.

Machine learning can be used to analyze data trends to develop a model. From analyzing these trends, the machine learning model can address the many issues facing rainfall forecasting, such as improving rainfall prediction accuracy, reduce the computational complexity of these predictions and tackle the issue of overfitting [6].

Deep learning on the other hand has become a talking point due to Convolutional Neural Networks (CNNs) ability to successfully classify images and self-engineer its features. Numerous research has thus far been done to evaluate the use of convolutional neural networks on rainfall images.

Often when implementing a machine learning or deep learning solution to a problem, a vast amount of computational resources are usually required to execute it. Big data technologies such as Apache Hadoop, Apache Spark, and Hive allow us to set up a cluster of computers which effectively makes use of the computational resources of many computers. The advantage of using such technologies to run various portions of solutions on these clusters and achieve better performance than running the solution on a single computer.

In this paper, we mention how big data technologies were used to compare a machine learning and deep learning solution to rainfall prediction.

II. Literature Review

This literature review is separated into 3 sections; Machine Learning for Rainfall Prediction, Deep Learning for Rainfall Prediction, and Big Data Technologies for Rainfall Prediction.

A. Machine Learning for Rainfall Prediction

Machine learning offers a method for reviewing historical data of rainfall in order to produce accurate results in predictions by determining accuracy and error. There exist a lot of different machine learning models and approaches to rainfall prediction, both in classification and regression based on the requirements of the system, therefore choosing the correct approach relies heavily on understanding the requirements and current state-of-the-art approaches. The research on machine learning for rainfall predictions looks at the types of models that previous researchers in this field used, their approach to this problem, also the results that they obtained.

Cramer *et al.* [7], showed the benefits of machine learning for rainfall prediction against current state-of-the-art techniques. Compares the state-of-the-art techniques Markov Chain extended with rainfall prediction with six other popular machine learning rainfall prediction algorithms: Genetic Programming, Support Vector Regression, Radial Basis Neural Networks, M5 Rules, M5 Model trees, and k-Nearest Neighbours. The data accumulated comprised of rainfall data from 20 and 22 cities from around Europe and the USA respectively. The authors used RMSE (Root Mean Square Error) to determine the overall performance of each proposed technique. It is seen from the results that SVR, RBF, and GP performed the best, contrasting M5P, GP, and MCRP which were the top performers prior to rainfall accumulation. It was also observed that the algorithms faced a lot of issues when addressing daily rainfall. Both prediction and fitting posed a challenge when trying to address daily rainfall. The models seemed to be underfitting the data to compensate. Each algorithm seemed to limit itself to focusing on the regression aspect while not taking into account the irregular nature of the data.

Other common approaches to rainfall prediction make use of neural networks and random forest regressors. In [6], the author describes an approach to rainfall prediction through the implementation of a hybrid intelligent system data mining technique for solving novel practical problems. The 'hybrid' referring to a mix of ANN's (Artificial Neural Networks) and Genetic Algorithms achieved satisfactory results. Another paper, [8] proposes a supervised learning model that is based on machine learning algorithms of data mining. The aim of this model is the classification of the different levels (low, mid, and high) of rainfall. The accuracy of the model on the selected regions is then surveyed and compared against classifiers such as Random Forest, SMO, Naive Bayes, and Multilayer Perceptron. The experiment used monthly rainfall data where the average monthly rainfall from January to December was calculated and two different thresholds for low, medium and high volume range of data were identified. The paper analyses the results to identify the correctly classified instances (CCI) and incorrectly classified instances (ICCI), ROC area, as well as accuracy and confusion matrix. It was observed that the Random Forest classifier provides better accuracy for all the regions. This was due to the generated number of decision trees which join together to enable a more accurate prediction.

B. Deep Learning for Rainfall Prediction

As mentioned earlier, deep learning has mainly become a talking point due to the ability of CNNs to successfully classify images and also their ability to self-engineer optimal features. On this note, various research in this field has taken advantage of CNNs to predict or classify rainfall. The research on rainfall prediction using deep learning can be divided using 3 data formats: time series, radar image, and satellite images. Having divided the research into this we look at what type of deep learning technique researchers used, their approach to the rainfall problem (regression or classification), and also the results they obtained.

Time series refers to a series of values of a quantity that is collected at successive times or at equal intervals. Time series values for rainfall can be collected in a number of ways, for example, sensors or rain gauges. With that being said, in Aswin *et al.* [1], aimed to model rainfall using deep learning architectures and time-series data. They worked with time-series data of global monthly average rainfall. They make use of the ConvNet and LSTM deep learning techniques to model their problem. A monthly rainfall amount (mm) was provided as input to the deep learning models and a monthly rainfall amount at 70 months later was provided as output. To evaluate their model, they used the Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) as metrics. The results of their work showed that for the ConvNet model, they achieved an RMSE of 2.44 and a MAPE of 1.72 and for the LSTM model they got an RMSE of 2.55 and a MAPE of 1.69.

Another type of data format that is common in the field of rainfall predictions is radar images. Zhang et al. propose a Tiny-RainNet model to directly predict rainfall within the next 1-2 hours using sequential radar echo maps. Their Tiny-RainNet model consists of 3 parts, a convolutional layer, bi-directional LSTM layers, and a few dense layers. The convolutional layer is used to extract context information from different receptive fields. The bi-directional LSTM layers are used to capture long-distance dependencies. The dataset consisted of radar echo maps. As input to the Tiny-RainNet model, they used 60 radar echo maps and produced a rainfall amount (mm/h) as output. Comparing their model to other CNNs, their results showed that they outperformed the rest [9]. The last data format that we look at is satellite images. In [10] proposed to predict rainfall at 1 day ahead using a ConvLSTM model and a satellite image dataset. As input to this model, they used 15 satellite images while predicting the satellite image at 1 day ahead as output. After comparing their model to the state-of-the-models, their results showed that they performed better.

C. Feasibility

In this section, we look at the feasibility/practicality of rainfall prediction.

For the obligation of distinction and unlimited determination precipitation devices, the National Severe Storms Laboratory, the National Weather Service and National Oceanic and Atmospheric Administration, jointly founded the Multisensor QPE and National Mosaic Project established real-time measurable rainfall approximations and announced all types of high tenacity (QPE) devices [12]. The permitted MPing software (V 2.0) can be utilized in gathering the climatological information utilizing social participation, collects the climatological information about the society's situation, and spreads it to the host [11]. GICS is an extensively held layer-based diagnostic instrument in a variety of sceneries due to its gentleness, smoothness, and speediness. The weather-predicting technique has accomplished long-term growth for

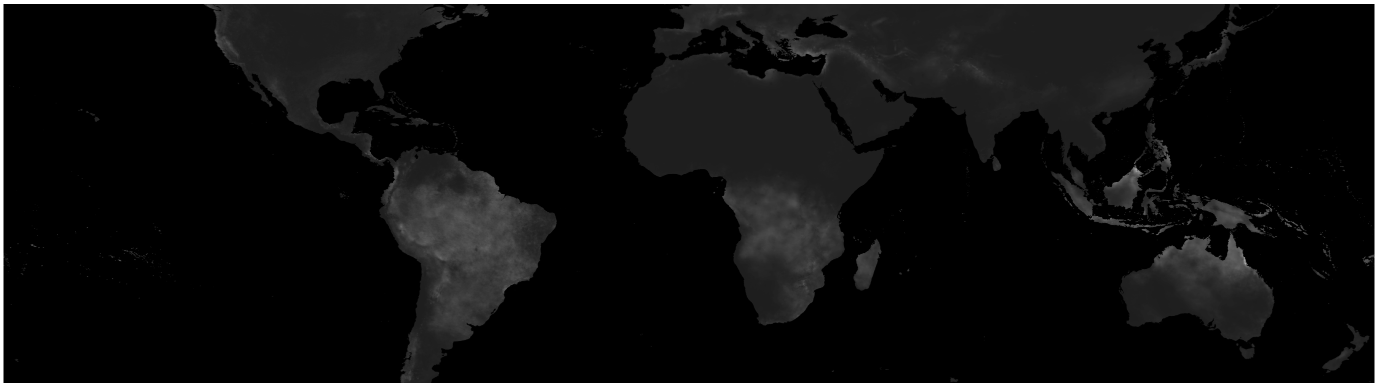


Fig. 1: Sample image from the dataset

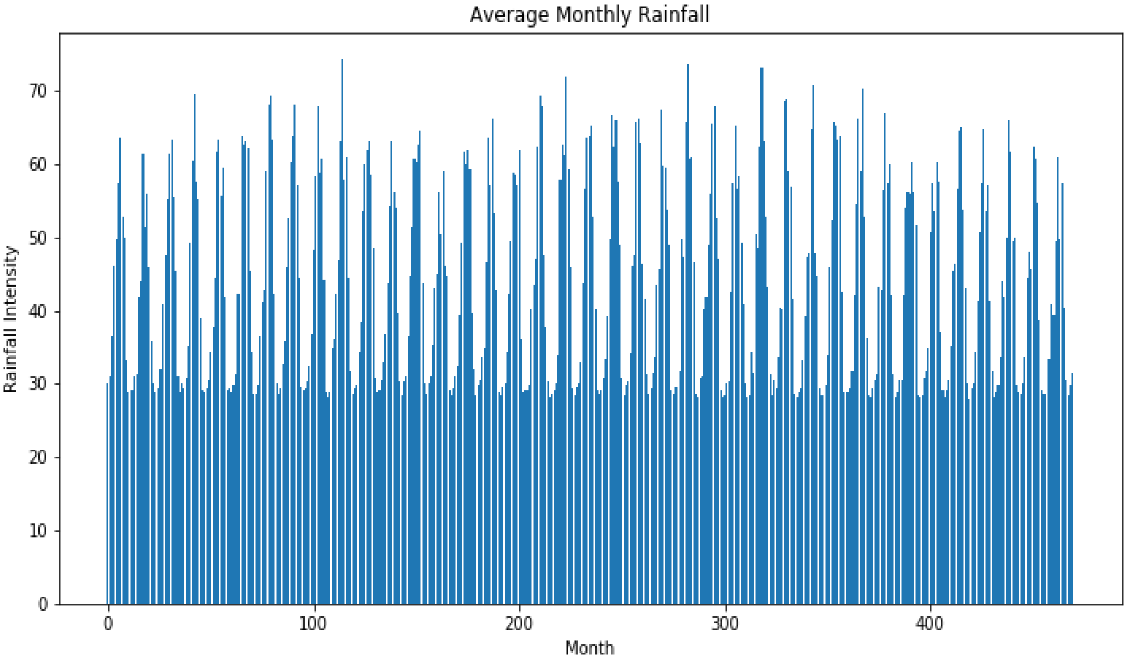


Fig. 2: Graph showing average monthly rainfall of Bangladesh

the past few years, with a lot of researchers instituting some forecast models around rainfall predicting, such as the ARIMA model, Markov model, and so on [13] [14]. These readings assisted with the improvement of the rainfall approximations. Equally, some of the confines must still be learned. Deep learning is the ground-breaking method recommended within the field of reproduction brainpower some years ago. The Markov technique is more suitable for the development of rainfall. Furthermore, the deep learning method education is faster, and it shows a better presentation than the general development of the method with the development of coaching trials. Deep learning can be able to be a well-organized big data management method by educating big data, gathering and mining the profound relationship between the big data to grow the organization and prediction precision. The weather forecast technique created on deep learning is ready to exceed the limitations of the current prediction approaches. Owing to the achievements of deep learning processes in diverse areas, many people have tried to use deep learning procedures in the field of climate forecasting, development has been completed [15]. Hsu [15] presented the possibility of discovering the alignment and restrictions of a three-layer frontward. Liu [15] recommended the deep neural network technique can refine the structures from the uncooked climate information. Adamowski and Belayneh measured the efficiency of three methods, the support vector regression, artificial neural networks, and wavelet neural networks (WN), by utilizing the usual rainfall guide for predicting dry circumstances in Ethiopia [18]. Afshin recommended a prolonged period of precipitation forecasting method utilizing combined neuro-fuzzy and wavelet prolonged precipitation predicting technique [16]. Ha used DBN to develop rainfall accuracy with the historical rain data, weather conditions in Seoul. Experiments did confirm that DBN functioned far better than other forecasting precipitation [19]. The other study has established the significance of a collective of ANNs and studying shapes for climate forecasting in Canada. Under the context of climatological big data, the deep learning machinery can use enormous multi-source climatological data and take adequate reflection data as preparation samples to guarantee the correctness of the meteorological conditions predicting technique. The deep learning technique can discover the essential data association between climatological fundamentals in-depth, and discover a more correct alternative model of multifaceted mechanism techniques between climate conditions and climatological elements [17].

III. Methodology

A. Dataset

The dataset used for this study consists of radar rainfall images. It contains 470 images. The resolution of all the images is 7200x2000 and each image is consists of 1 color channel. The time interval between the images is 1 month. Figure 1 shows a sample of one of the images from the dataset. Seeing that these images were quite large in size, we decided to only use a cropped region of Bangladesh. Once cropped, we resized the images to 40x40. Figure 4 shows a sample of one of the images from the Bangladesh region.

B. Dataset Exploration

To gather some insight from the dataset, we decided to plot the average monthly rainfall of each image using the grayscale rainfall intensity representative. This process was done using Spark's map and reduce functions. An overview of the process is shown in Figure 5.

Using the output from this Spark, we can now visualize the result of our average monthly rainfall. Figure 2 shows this. From this figure, we can see that the average monthly rainfall for the Bangladesh region appears to be periodic. This makes sense because Bangladesh experiences little rainfall in January-May, experiences the majority of its rainfall between its monsoon season, June-October, and then experiences little rainfall between November-December as well.

C. Preprocessing

Predictions for this project are done on a pixel by pixel basis. What this means is that we tried to predict each pixel in an image when given a sequence of pixels. To create this format, the preprocessing for this project involves 3 key steps; cropping images, flatten pixels of images, and sequence creation. As explained earlier, we cropped the original images of the dataset to just include the Bangladesh region and then resized this cropped image to an image resolution of 40x40. The next thing we did was to create an image array for each image and then produce a flattened image array as shown on the left of Figure 3. Following this, we created a sequence of pixels where the sequence length is the number of images that will be used to make a prediction. An example of a sequence length of 3 is shown in Figure 3, we took 3 monthly rainfall images to make a prediction. For the study, we used 100 of the cropped images to create the training set sequences, and 50 of the cropped images to create the testing set sequences.

D. Evaluation Process

The aim of this project is to compare the rainfall predictions of a Random Forest model versus the rainfall prediction of a Deep LSTM model for sequence lengths of 4, 8, and 12 when predicting 1, 2, and 3 months ahead. To do this, we had to train the models individually on each sequence length and for each month ahead. In order to make sure that we can compare the 2 models, we had to make sure that each model is trained on the same training set sequences. Following this, we were then able to predict the values for the testing set sequences. The metrics used to calculate the error between the testing sets values and prediction sets values is the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE). Once we calculated the metrics for the testing and prediction sets values for every sequence length and month ahead, we then passed the results to spark for the thorough analysis. Spark then used these results to determine which model performed best for sequence lengths of 4, 8, and 12 and for 1, 2, and 3 months ahead.

IV. Architecture

A. Big Data System Design and Architecture

In order to use big data technologies for our project, we needed to implement these technologies in various parts of our architecture. The big data technologies that we uses for the project were Apache Hadoop and Apache Spark. Apache Hadoop provided us with a robust storage solution while Apache Spark provided us with an efficient way to query data for analysis.

Using Figure 6 to demonstrate our architecture, the first thing we did is the preprocessing of the radar rainfall images. As mentioned earlier, this step involves cropping the images and sequence creation. We also created a flattened image dataset. These sequences and flattened images are then saved as a structured data format and ingested into Hadoop HDFS for robust storage. We then used Hadoop's and Spark's map-reduce functionality to get the average monthly rainfall using the stored flattened images and provide some analysis about the Bangladesh region's periodicity. Next, we fetched the sequences from Hadoop using Spark and created training and testing sets so that we can predicting monthly rainfall. We then created a Random Forest model using the Scikit-Learn Python library and a Deep LSTM model using the Keras Python API. These machine learning and deep learning models are next trained on the sequence lengths of 4, 8, and 12 and used to predict 1, 2, 3 months ahead for each sequence length. The RMSE and MAE were then calculated between the testing and prediction values are then sent to Spark. In Spark, we create

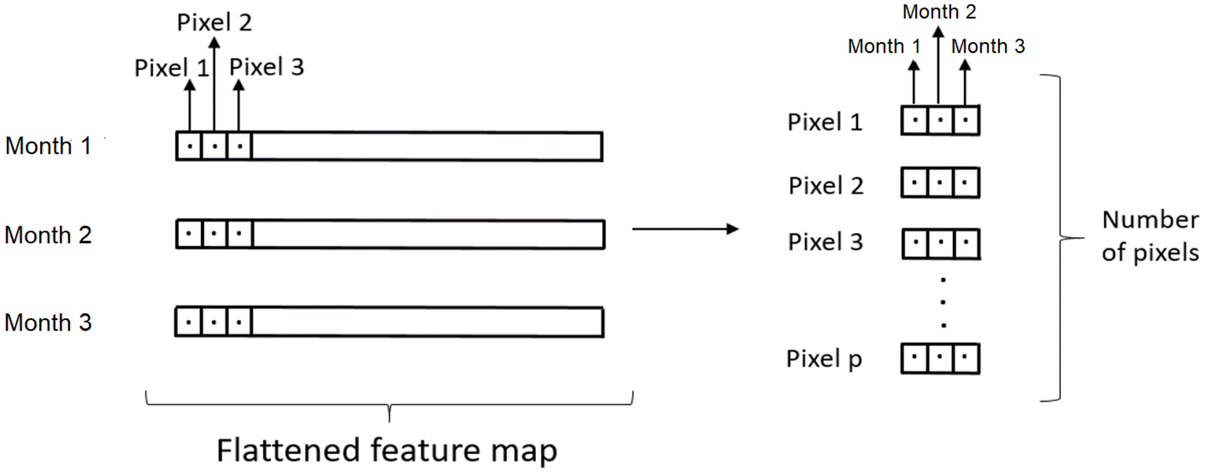


Fig. 3: Preprocessing steps

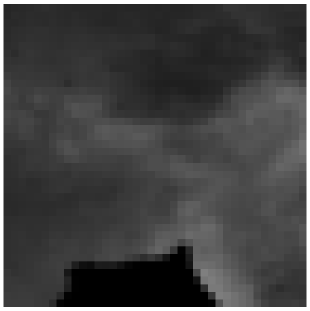


Fig. 4: Sample cropped image

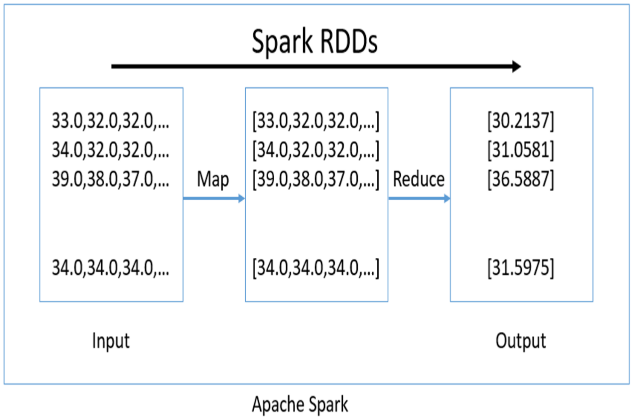


Fig. 5: Spark process

a Spark data frame using these results and a schema for it. This data frame was then converted to a Spark SQL table to be able to query specific results.

B. Random Forest Model Design

A Random Forest is a supervised learning algorithm based on decision trees. Decision trees offer predictions based on branches of ”if...then...” decision splits and are a commonly used technique in creating predictive models. An example of a decision tree can be seen in figure 7, where tree #1 starts at the base of the tree (the first blue dot), then at this point, a decision is made and follows the branch according to the decision made until it reaches the endpoint of the tree, also called a leaf. In machine learning, if a sample is chosen as the base, then the decision or split in branches can be seen as a feature and the endpoint as a prediction or value. The hyper-parameters of the model specify the percentage of the total features that can be split on at each node, which results in the model not relying on individual features and enabling usage of all potential features. At each branch, the feature thresholds that best split the (remaining) samples locally is found. The problem with decision trees is that they are not very robust and do not generalize well to unseen data (often results in overfitting), herein lies the purpose of Random Forest. Random Forest uses an ensemble learning method by combining the result from multiple decision (as seen in figure 7) to be used in regression and classification. At training time, a large number of decision trees are created and aims at providing the mode of the classes (classification) or mean prediction (regression) from the collection of individual trees. Random Forest uses bootstrapping which is the procedure of randomly sampling subset data with replacement, i.e. randomly sampling data from the original dataset when the splits at each node are generated which adds the element of randomness and prevents

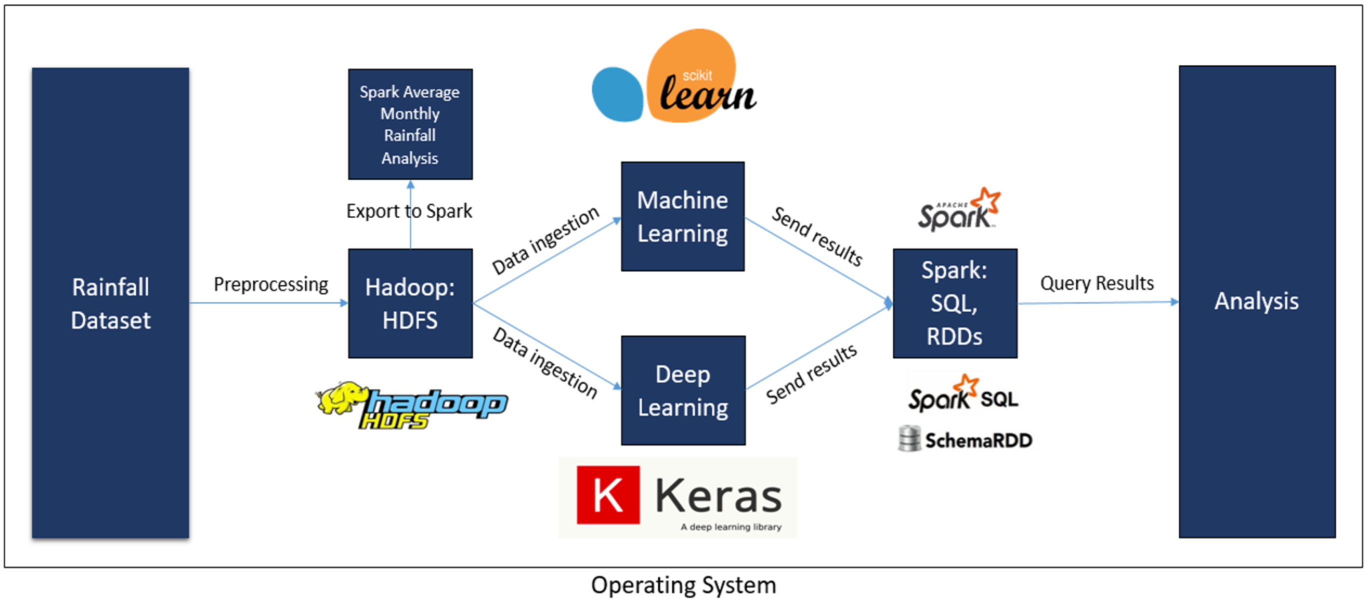


Fig. 6: Big data system architecture

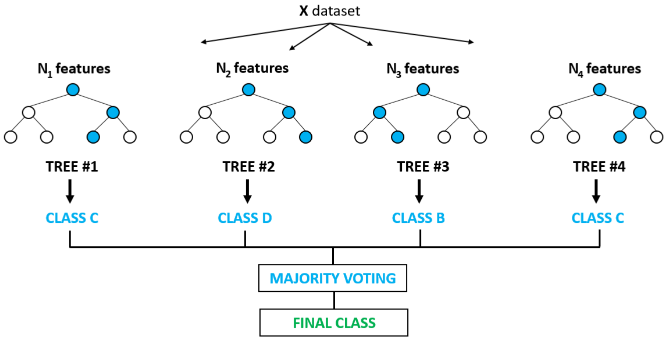


Fig. 7: Random Forest

overfitting. This also allows for a better understanding of the variance and bias within the dataset. The random forest also utilizes a bagging technique which means trees are run in parallel with no interaction until the outputs are aggregated where no preference is given to any model. For our model, we used a random forest regressor to predict rainfall. The hyper-parameters of the model are the parameters which are set before the model is trained and defines how the training of the model is done. Each hyper-parameter offers a different way of tuning the tree in order to achieve optimal results. Our model consisted of the hyper-parameters seen in figure 9.

C. Deep LSTM Model Design

Long Short Term Memory (LSTM) is a type of recurrent neural network (RNN) used in sequential problems. LSTMs makes use of gates to control the flow of information in the recurrent structure. Some advantages of LSTMs are that they are good at holding memory over long time steps and they

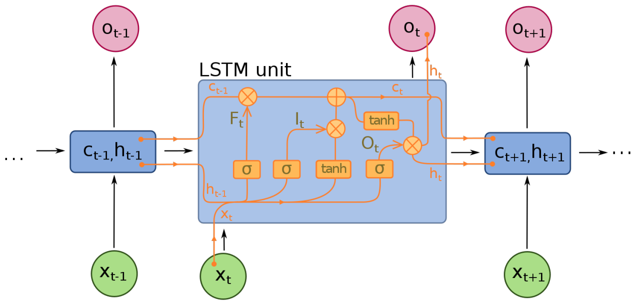
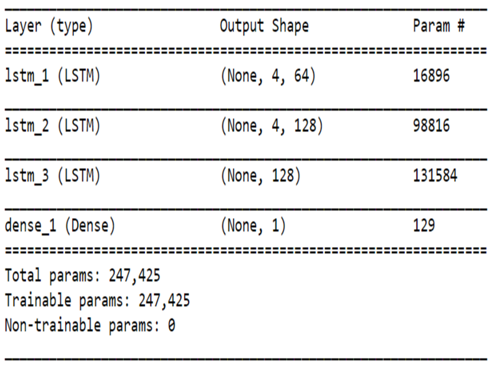


Fig. 8: LSTM cell

```
{'bootstrap': True,
 'ccp_alpha': 0.0,
 'class_weight': None,
 'criterion': 'gini',
 'max_depth': None,
 'max_features': 'auto',
 'max_leaf_nodes': None,
 'max_samples': None,
 'min_impurity_decrease': 0.0,
 'min_impurity_split': None,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 100,
 'n_jobs': None,
 'oob_score': False,
 'random_state': None,
 'verbose': 0,
 'warm_start': False}
```

Fig. 9: Random Forest hyper-parameters



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_1 (LSTM)                (None, 4, 64)             16896
_________________________________________________________________
lstm_2 (LSTM)                (None, 4, 128)            98816
_________________________________________________________________
lstm_3 (LSTM)                (None, 128)               131584
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 129
=================================================================
Total params: 247,425
Trainable params: 247,425
Non-trainable params: 0
_________________________________________________________________
```

Fig. 10: Deep LSTM model architecture

overcome the problems of exploding and vanishing gradients experienced in many other RNNs [20]. Figure 8 shows an example of an LSTM cell. The cell makes use of a memory unit that keeps information over time and gating units which regulate the flow of information in and out of the memory. The types of gating units found within an LSTM cell are the input, output, and forget gates. The input gate controls how much new information is added to a cell state from the current input, the forget gate manages what information to discard from memory, and the output gate conditionally decides what to output from memory [20]. In this project, a Deep LSTM model is used for the prediction of rainfall. A Deep LSTM is essentially just a combination of many LSTM cells where the output of the one cell is fed as input to the next cell. Deep LSTM models have a tendency to perform better than regular LSTM models when given enough data. The structure of our Deep LSTM model is shown in Figure 10. Our model consists of 3 LSTM cells and 1 dense layer that provides the predicted value.

V. Experimental Setting

A. Environment

The environment in which we worked was Windows 10 which had 16 GB RAM and an i7 8th Generation. Apache Hadoop, Apache Spark, Jupyter Notebook were all installed in this environment.

B. Tools and Hardware

The tools we made use of for this project were:

- Python programming language
- Jupyter Notebook
- Apache Hadoop
- Apache Spark (Pyspark)
- Keras Python API
- Tensorflow Python Library
- Scikit-Learn Python Library

VI. Results and Analysis

According to the methodology chapter of this paper, the Evaluation Process section would compare the rainfall predictions of a Random Forest (RF) model versus the rainfall prediction of a Deep LSTM. The model for sequence lengths of 4, 8, and 12 when predicting 1, 2, and 3 months ahead will also be looked at. The RMSE and the MAE can be utilized collectively to establish the dissimilarity on the inaccuracies in a group of predictions. The MAE will constantly be smaller or equal to the RMSE. In essence, this means that the larger the variance amongst them, then the greater the difference in the specific errors in the illustration. If the MAE equals RMSE, then that means all errors have equal size. It is possible for both the RMSE and the MAE to range starting at 0 to infinity. These are referred to as negatively-oriented results. The lower the values the better the performance. Therefore, a descriptive analysis has been conducted based on each month over a three months period.

A. 1 Month Ahead Results and Analysis

1 Month *MAE* Analysis: According to the 1 Month ahead graphs are shown in Figure 11 and 12, the results indicate that the Deep LSTM performed better than RF at sequence length 4 with values of 7.01 and 8.74 respectively. RF performed relatively better than the Deep LSTM at sequence length 8 with RF having an MSE value of 6.73 and Deep LSTM having an MSE value of 7.51. Deep LSTM performed slightly better than RF at sequence length 12 with a 5.41 MAE value as compared to the RF which had a MAE value of 5.54.

1 Month *RMSE* Analysis According to the 1 Month ahead graphs shown in Figure 13 and 14, the Deep LSTM performed better than the RF at sequence length 4 with RF having a value of 14.76 compared to the 11.68 value of the Deep LSTM. At a sequence length 8, the same outcome was seen with the RF having a value of 13.26 compared to the 11.35 value for the Deep LSTM. The RF and Deep LSTM performed relatively similar at a sequence length of 12 with RF having a value of 9.89 and the Deep LSTM having a value of 9.27.

B. 2 Months Ahead Results and Analysis

2 Months *MAE* Analysis: According to the 2 Months ahead graphs shown in Figure 15 and 16, the results indicate that the Deep LSTM performed better than RF at sequence length 4 with values of 8.81 and 10.76 respectively. RF performed slightly better than the Deep LSTM at sequence length 8 with RF having a MSE value of 8.68 and Deep LSTM having a MSE value of 8.81. Deep LSTM performed slightly better than RF at sequence length 12 with a 5.59 MAE value compared to 5.63.

2 Months *RMSE* Analysis: According to the 2 Months ahead graphs shown in Figure 17 and 18, the Deep LSTM performed much better than the RF at sequence length 4 with RF having a value of 18.04 compared to the 14.88 value of the Deep LSTM. At a sequence length 8, the same outcome was seen with the RF having a value of 15.14 compared to the 13.04 value for the Deep LSTM. The Deep LSTM performed slightly better than the RF at a sequence length of 12 with RF having a value of 10.02 and the Deep LSTM having a value of 9.67.

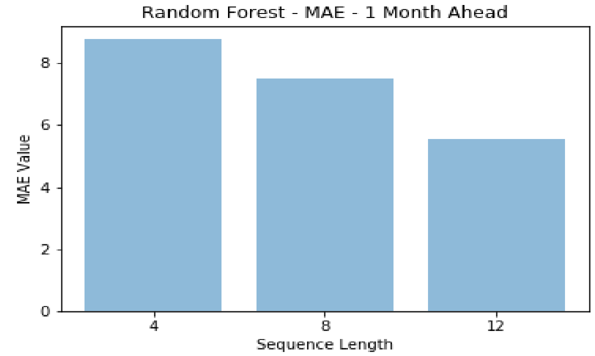


Fig. 11: Random Forest MAE - 1 Month Ahead

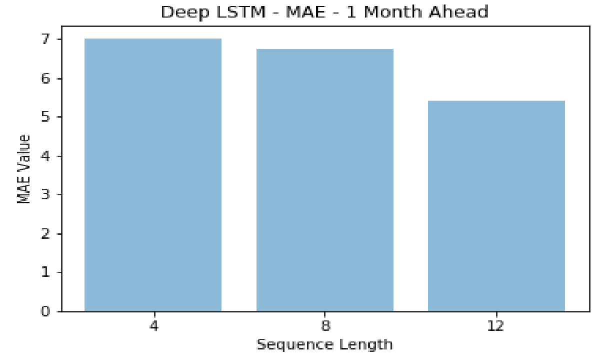


Fig. 12: Deep LSTM MAE - 1 Month Ahead

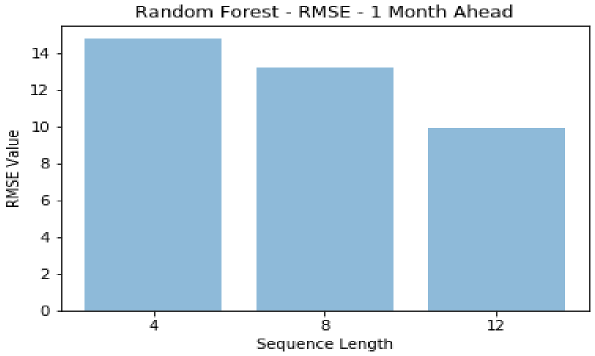


Fig. 13: Random Forest RMSE - 1 Month Ahead

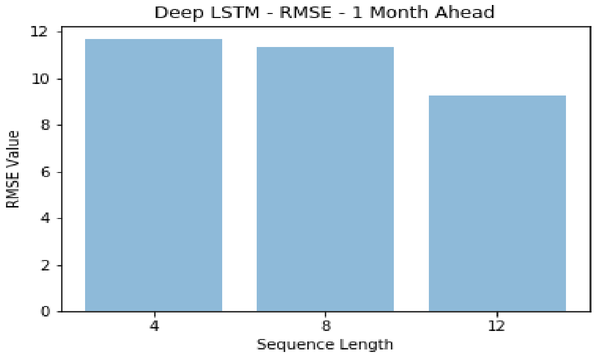


Fig. 14: Deep LSTM RMSE - 1 Month Ahead

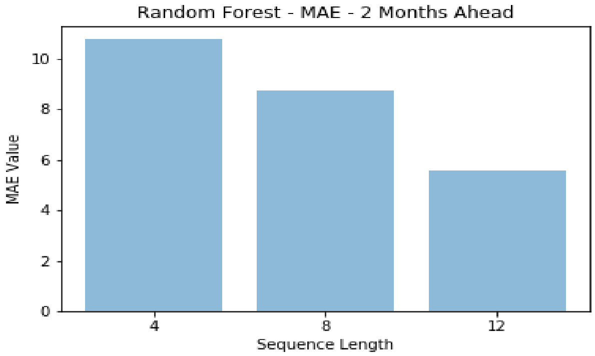


Fig. 15: Random Forest MAE - 2 Month Ahead

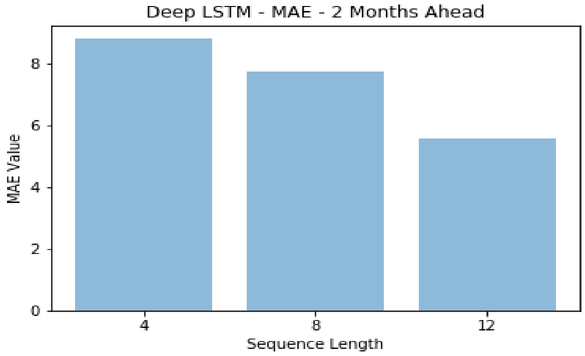


Fig. 16: Deep LSTM MAE - 2 Month Ahead

C. 3 Months Ahead Results and Analysis

3 Months *MAE* Analysis: According to the 3 Months ahead graphs shown in Figure 19 and 20, the results indicate that the Deep LSTM performed better than RF at sequence length 4 with values of 9.78 and 11.91 respectively. The same result for a sequence length of 8 was seen with the Deep LSTM performing slightly better than the Rf with RF having a MSE value of 8.89 and Deep LSTM having a MSE value of 8.22. The RF performed slightly better than the Deep LSTM at sequence length 12 with a 5.59 MAE value compared to 5.90.

3 Months *RMSE* Analysis: According to the 3 Months ahead graphs shown in Figure 17 and 18, the Deep LSTM performed much better than the RF at sequence length 4 with RF having a value of 19.99 compared to the 16.25 value of the Deep LSTM. At a sequence length 8, the same outcome was seen with the RF having a value of 15.60 compared to the 13.66 value for the Deep LSTM. The RF performed slightly better than the Deep LSTM at a sequence length of 12 with RF having a value of 9.95 and the Deep LSTM having a value of 10.04.

Although the Deep LSTM and the RF methods attained

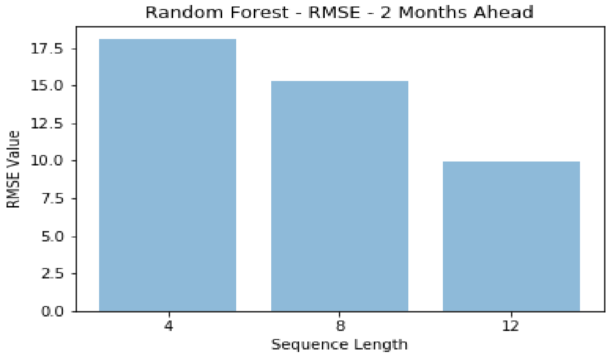


Fig. 17: Random Forest RMSE - 2 Month Ahead

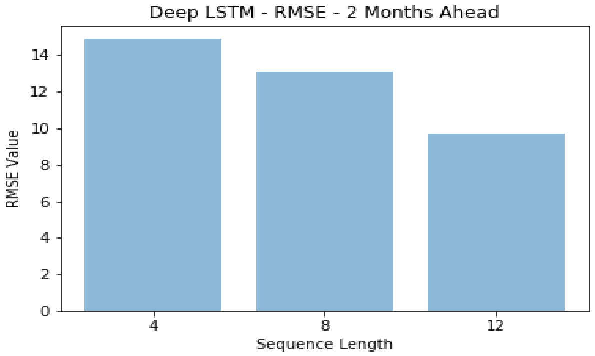


Fig. 18: Deep LSTM RMSE - 2 Month Ahead

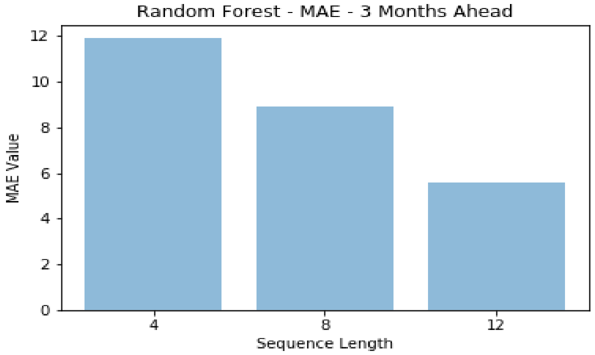


Fig. 19: Random Forest MAE - 3 Month Ahead

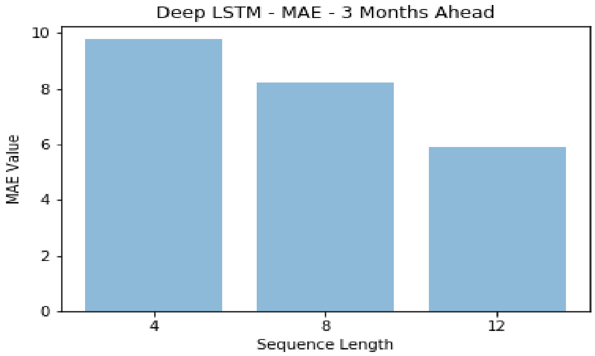


Fig. 20: Deep LSTM MAE - 3 Month Ahead

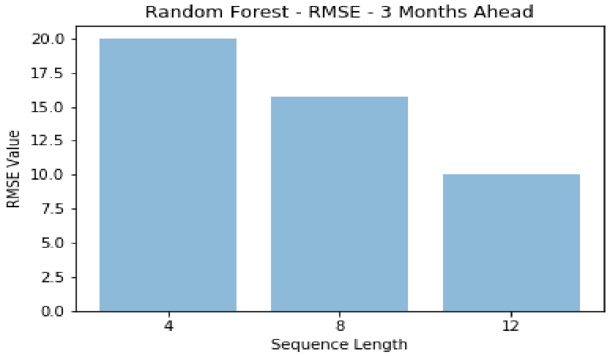


Fig. 21: Random Forest RMSE - 3 Month Ahead

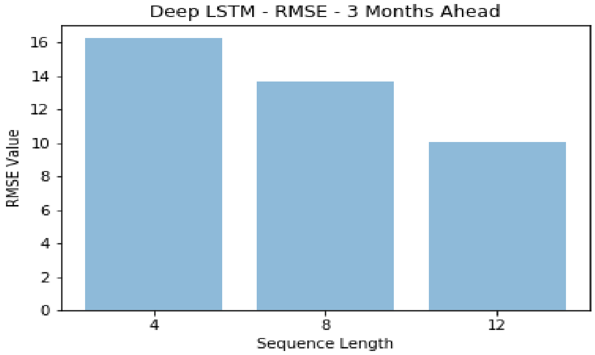


Fig. 22: Deep LSTM RMSE - 3 Month Ahead

relatively comparable showing considering our experiments, it appears that the Deep LSTM technique performed relatively better compared to the RF on the rainfall predictions for sequence lengths of 4, 8, and 12 when predicting 1, 2, and 3 months ahead.

VII. Business Value and Process

Data Science (DS) involves expertise and discipline areas to yield a complete, detailed, and distinguished outlook into raw information. Currently, DS is recognized as a method of climate prediction and it all depends on new methodologies to provide continuous improvement tactics to drive the industry into higher lengths. There is no well-positioned associate for data science than artificial intelligence and machine learning.

Data science commonly consist of five-stage development that entails the following:

1) Capturing: Data acquirement, data recording, signals reception, and extraction of data.
2) Maintaining: Data cleansing, Data warehousing, data processing, data architecture, data staging.
3) Processing: Data mining, clustering/classification, data modeling, data consolidation.
4) Communicating: Data reporting, business intelligence,

data visualization, and decision making.

5) Analysing: Exploratory/confirmatory, predictive analysis, regression, text mining, and qualitative analysis.

Data science can be used for the following:

1) Anomaly detection an example is a fraud;
2) Automation and decision-making example is background checks/creditworthiness;
3) Classification involves classifying rainfall images such as low, moderate, and high amounts of rainfall
4) Forecasting an example is rainfall prediction.
5) Pattern detection an example is weather patterns;
6) Recognition an example is recognizing someone's voice; and
7) Recommendations and examples could be based on learned preferences.

Therefore, data scientists ought to be knowledgeable in many aspects including but not limited to mathematics, data engineering, advanced computing statistics, and visualizations to possess an ability to efficiently examine complete multitudes of data and transfer only the utmost valuable information which will assist to lead to efficiency and innovation.

1) Improved industry resilience for weather tragedies: New regular tragedies and bad climate measures linked to weather variations occurring globally. Analyzing, gathering, and tracing climate data assists industries to detect climate developments and also assists their businesses to be extra resilient to dangerous climate occurrences. When the weather developments are detected and acknowledged, companies can forecast where and when the climate conditions can harmfully influence their tasks and also activate their readiness strategies. Appreciating the climate information assist to better prepare for bad natural disaster and weather strategies.
2) Climate predictions and warnings will help with employee safety and productivity: Climate data is utilized to protect the business's greatest significant assets like workers and physical assets. Climate prototypes can significantly take advantage of the institution of machine learning-based processes because this technology can develop huge quantities of climate information and also improve its situation for extra precise forecasts the further it gets utilized. It has been scientifically proven that when employees are feeling safe then they can be more productive, which does well for the business. If you utilize bad climate warning equipment to guard workers, and they are also aware of such measures, that will not assist them with their safety but with their efficiency as well.
3) Climate data can assist to make good business decisions and save money: Severe climate situations can create monetary difficulties and operational ineffectiveness for companies. The cash-flow can be severely impacted by bad weather conditions. When companies want to be masters of business continuity, fiscal, and HR departments, the business must begin to integrate and observe climate data, predictions, and also to fit their requirement. The most important part is that lots of different and dynamic weather tools and kits are available to companies and can be designed to fit their requirement.

VIII. Individual Experiences & Future Improvements

A. Keelan Govender

The results from a random forest approach could be improved by tweaking the hyper-parameters. Each of the values of the hyper-parameters has a direct impact on the model and therefore could potentially provide a more impactful result on rainfall prediction with adjustments to these values.

The training time for the random forest model was lengthy. This could be improved by training the model on a platform that aims at improving machine learning efficiency such as Apache Spark's machine learning library 'MLlib'.

B. Abdullah McDonald

The types of experience I have gained from this project can be looked at from two perspectives. The first one is in terms of the project itself, and the challenges that I faced. Some of the challenges I faced was setting up a Spark cluster so our group could train our predictive models in parallel and actually see the significant benefits of big data technologies. Another issue was not having enough memory to train my Deep LSTM model with more training examples. Due to this lack of resources, however, I was taught to improvise. This could be a valuable lesson that could be transferred to the implementations of other projects.

Looking at the project from a team perspective, we also had challenges. The main problem the group faced was having to deal with another group member who never contributed to our project at all. We merely decided to exclude that particular member from the project and manage their workload. This circumstance taught us that it is always a good idea to have a plan B when things don't go according to plan.

There are a number of improvements that could be done for the machine learning and deep learning sections of this project if the time frame was expanded, these are:

- Training the Random Forest and Deep LSTM models in parallel using Apache Spark or Apache Hadoop
- Using hyper-parameter optimization to improve the predictions of the respective models
- Random Forest and Deep LSTM models using Apache Spark MLib

C. Unathi Ntame

I have obtained new skills, knowledge, and a better understanding of important tools for reading and analyzing data. I have demonstrated the commitment, desire to improve and learn new things to ensure continuous self-development. There are better career prospects because of having learned new information from various fields. Therefore, the individual benefits acquired from the project are as follows:

- Having utilized spark for the first time, this project has provided an external and aerial view of how important the information can be viewed, explored, and manipulated.
- Based on the team dynamics where IFS and Computer Science students were combined to do this project, that has assisted to work well, patiently, and committed to the fellow group and team members.
- Based on the newly acquired knowledge on the installation of Anaconda, Python, Spark, and all other programs, this will support the career development strategies which can be of benefit to the industry.
- Since I'm not coming from a programming environment, this project has assisted with improving capabilities and competency level in order to deliver in industries big projects and increase customers satisfaction levels by simplifying lots of processes and procedures which can positively affect the morale and productivity of fellow teammates
- Based on the assignments that we had to do and the newly learned style of referencing, this project can assist to improve projects methodologies and approaches with a focus to reduce turnaround times.
- Based on the strict timelines, operating under immense pressure under the difficult time of the pandemic, this project will assist with an ability to deliver with speed, simplicity, and trust that will contribute positively to the image, brand, and credibility of the organization from external views.

IX. Conclusion

The aim of our paper was to predict monthly rainfall on images of 1,2, and 3 months ahead for sequence length of 4,8, and 12 images using a Random Forest and Deep LSTM model using the big data technologies of Hadoop and Spark. The data we used was stored and processed into our Random Forest and Deep LSTM models. For sequence lengths of 4 and 8 images, the results of our implementation showed that the Deep LSTM and Random Forest models performed similarly, however, for sequence length of 12, the Deep LSTM outperformed the Random Forest model. Therefore the Deep LSTM model was established to be the better of the two models.

References

[1] S. Aswin, P. Geetha, and R. Vinayakumar, "Deep Learning Models for the Prediction of Rainfall," Proc. 2018 IEEE Int. Conf. Commun. Signal Process. ICCSP 2018, pp. 657–661, 2018, doi: 10.1109/ICCSP.2018.8523829.

[2] E. Tuba, I. Strumberger, N. Bacanin, and D. Zivkovic, "Detection in Microscopic Digital Images," vol. 1, no. Iii, pp. 142–151, 2019, doi: 10.1007/978-3-030-26354-6.

[3] L. Chen, Y. Cao, L. Ma, and J. Zhang, "A Deep Learning-Based Methodology for Precipitation Nowcasting With Radar," Earth Sp. Sci., vol. 7, no. 2, 2020, doi: 10.1029/2019ea000812.

[4] Y. Cao et al., "Precipitation Nowcasting with Star-Bridge Networks," pp. 1–10, 2019.

[5] S. Agrawal, L. Barrington, C. Bromberg, J. Burge, C. Gazen, and J. Hickey, "Machine Learning for Precipitation Nowcasting from Radar Images," no. NeurIPS, pp. 1–6, 2019.

[6] V. C. B, J. S. B, and M. Bhavsar, "Weight Based Workflow Scheduling in Cloud Federation," Inf. Commun. Technol. Intell. Syst. (ICTIS 2017) - Vol. 1, vol. 83, no. Ictis 2017, pp. 1–7, 2017, doi: 10.1007/978-3-319-63673-3.

[7] [1] S. Cramer, M. Kampouridis, A. A. Freitas, and A. K. Alexandridis, "An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives," Expert Syst. Appl., vol. 85, pp. 169–181, 2017, doi: 10.1016/j.eswa.2017.05.029.

[8] [3] A. K. Sharma, S. Chaurasia, and D. K. Srivastava, "Supervised Rainfall Learning Model Using Machine Learning Algorithms," Adv. Intell. Syst. Comput., vol. 723, pp. 275–283, 2018, doi: 10.1007/978-3-319-74690-6_27.

[9] C. Zhang, H. Wang, J. Zeng, L. Ma, and L. Guan, "Tiny-RainNet: A Deep CNN-BiLSTM Model for Short-Term Rainfall Prediction," 2019.

[10] Y. M. Souto, F. Porto, A. M. Moura, and E. Bezerra, "A Spatiotemporal Ensemble Approach to Rainfall Forecasting," Proc. Int. Jt. Conf. Neural Networks, vol. 2018-July, 2018, doi: 10.1109/IJCNN.2018.8489693.

[11] Y. Pan, Y. Shen, J. Yu, A. Xiong, "An experiment of high-resolution gauge-radar-satellite combined precipitation retrieval based on the bayesian merging method," J. Meteorol. Vol. 73, pp. 177–186.

[12] C. Kondragunta, D. J. Seo, "Toward integration of satellite precipitation estimates into the multisensor precipitation estimator algorithm," Available online: https://www.google.com.hk/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=2ahUKEwj2jrnJ4Z3dAhUOHXAKHWqoAD8QFjABegQICBAC&url=https%3A%2F%2Fams.confex.com%2Fams%2Fpdfpapers%2F7102.pdf&usg=AOvVaw3O8d6_9aKJB8DEx0731J-P (accessed on 3 September 2004)

[13] H. Qian, P.Y. Li, T. Wang, "Precipitation prediction on shizuishan city in ningxia province based on moving average and weighted markov chain," J. North China Institution, vol. 31, pp 6–9.

[14] T. Wang, H. Qian, P.Y. Li, "Prediction of precipitation based on the weighted markov chain in yinchuan area," South-to-North Water Transfer. Water Science. vol. 8, pp 78–81.

[15] J.N.K Liu, Y. Hu, Y. He, P.W. Chan, I. Lai, "Deep neural network modeling for big data weather forecasting, information granularity, big data, and computational intelligence," springer international Publishing: Cham, Switzerland, 2015; pp. 389–408.

[16] S. Afshin, "Long term rainfall forecasting by integrated artificial, neural network-fuzzy logic-wavelet model in karoon basin," 2011.

[17] M. Valipour, "Optimization of neural networks for precipitation analysis in a humid region to detect drought and wet year alarms," pp. 91–100, 2016.

[18] J.L Du, Y.Y. Liu, "A Prediction of precipitation data based on support vector machine and particle swarm optimization algorithms," pp. 10 - 57, 2017.

[19] W. Liu, Z. Wang, X. Liu, "A survey of deep neural network architectures and their applications," Neurocomputing. pp. 11–26, 2016.

[20] A. Iaddad, "Basic understanding of LSTM," Good Audience, 13 March 2019. [Online]. Available: https://blog.goodaudience.com/basic-understanding-of-lstm-539f3b013f1e. [Accessed 19 June 2020].