DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

COMPARISON OF BIG DATA TIME SERIES ANALYSIS METHODS

by

Fadıl Can MALAY

June, 2022

İZMİR

COMPARISON OF BIG DATA TIME SERIES ANALYSIS METHODS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of
Philosophy in Computer Engineering

by
Fadıl Can MALAY

June, 2022
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"COMPARISON OF BIG DATA TIME SERIES ANALYSIS METHODS"** completed by **FADIL CAN MALAY** under supervision of **ASSOC**. **PROF. DR. MEHMET HİLAL ÖZCANHAN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

....

Assoc. Prof. Dr. Mehmet Hilal ÖZCANHAN

Supervisor

....

Prof. Dr. Mehmet UNLUTURK

(Jury Member)

....

Asst. Prof. Dr. Feriştah DALKILIÇ

(Jury Member)

Prof. Dr. Okan FISTIKOĞLU

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor, Assoc. Prof. Mehmet Hilal ÖZCANHAN. Also thanks for conveying his own experience in many fields such as master's, thesis writing and supporting me in my studies.

I would also like to thank Murat EMEÇ who helped me in these studies. And special thanks to my fiance, Azize Yalçın who supports me throughout my life.

Fadıl Can MALAY

COMPARISON OF BIG DATA TIME SERIES ANALYSIS METHODS

ABSTRACT

Time series analysis is interpretation of data recorded at regular or irregular time intervals. The analysis is applied to earthquakes, sales, financial investment forecasting, early warning, decision making and many other areas. There are quantitative and qualitative methods of time series analysis of big data. The aim of this study is to compare two important quantitative methods: Autoregressive Integrated Moving Average and Long Short-Term Memory. For comparison, the accuracy of earthquake magnitude and location prediction has been used. Earthquake data (earthquake magnitude, longitude, latitude; sun and moon altitude-azimuth-distance to earth at the instant of the earthquake) between years 1970 and 2019 were utilized. Eighty percent of the United States Geological Survey data was used for training and the remaining for testing. Earthquakes of magnitude 4.0 and above were considered. The results of each method were compared with the results of previous works, using standard deviation, mean squared error, mean absolute error and median absolute error performances. ARIMA was inefficient in long term predictions. Therefore, LSTM was deemed as the appropriate method for the analysis of long term, irregular data, as supported in the literature too. Vanilla and Stacked models of LSTM were compared and Stacked LSTM provided the best results after being optimized by changing the dense, batch size and epoch parameters. The best results were obtained by using parameters 8 dense, 64 batch size and 64 epochs. While the longitude prediction was way off, the best predicted outcomes were the magnitude and latitude of the earthquakes.

Keywords: Time series analysis, forecasting, deep learning, earthquake prediction

BÜYÜK VERILERIN ZAMAN SERI ANALIZ METOTLARININ KARŞILAŞTIRILMASI

ÖZ

Zaman serisi analizi, düzenli veya düzensiz zaman aralıklarında kaydedilen verilerin yorumlanmasıdır. Analizler depremler, satışlar, finansal yatırım tahminleri, erken uyarı, karar verme ve diğer birçok alanda uygulanmaktadır. Büyük verilerin zaman serisi analizinin nicel ve nitel yöntemleri vardır. Bu çalışmanın amacı, iki önemli nicel yöntemi karşılaştırmaktır: Otoregresif Bütünleşik Hareketli Ortalama (ARIMA) ve Uzun Kısa Süreli Bellek (LSTM). Karşılaştırma için deprem büyüklüğü ve yer tahmininin doğruluğu kullanılmıştır. 1970-2019 yılları arasındaki deprem verilerinden (deprem büyüklüğü, boylam, enlem; güneş ve ayın yükseklik-azimut-deprem anındaki dünyaya olan uzaklığı) yararlanılmıştır. Amerika Birleşik Devletleri Jeolojik Araştırma verilerinin yüzde sekseni eğitim için, geri kalanı ise test için kullanılmıştır. 4.0 ve üzeri büyüklükteki depremler dikkate alınmıştır. Her yöntemin sonuçları, standart sapma, ortalama karesel hata, ortalama mutlak hata ve medyan mutlak hata performansları kullanılarak önceki çalışmaların sonuçlarıyla karşılaştırılmıştır. ARIMA, uzun vadeli tahminlerde yetersiz kalırken, LSTM'in literatürde de desteklendiği gibi uzun dönemli, düzensiz verilerin analizi için uygun bir yöntem olduğu görülmüştür. LSTM'nin Vanilla ve Stacked modelleri karşılaştırılmış ve Stacked LSTM yoğunluk, parti boyutu ve dönem parametreleri değiştirilerek optimize edildikten sonra en iyi sonuçları vermiştir. En iyi sonuçlar 8 yoğunluk, 64 parti boyutu ve 64 dönem parametreleri kullanılarak elde edilmiştir. Boylam tahmini çok uzak olsa da en iyi tahmin edilen sonuçlar depremlerin büyüklüğü ve enlemleri olmuştur.

Anahtar kelimeler: Zaman serisi analizi, öngörme, derin öğrenme, deprem tahmini

CONTENTS

Page

M.Sc THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ v
LIST OF FIGURES viii
LIST OF TABLES ix
LIST OF SYMBOLS x
ABBREVIATIONS xi

CHAPTER ONE - INTRODUCTION 1

1.1 General 1
1.2 Purpose ………….. 2
1.3 Problem Definition 2
1.4 Organization of the Thesis 2

CHAPTER TWO - LITERATURE REVIEW 4

2.1 Related Works 4
2.2 Materials and Method 5
2.2.1 Materials 5
2.2.2 Method 5

CHAPTER THREE - COMPARISON OF THE GENERAL PROPERTIES OF THE TWO METHODS 7

3.1 Time Series 7
3.2 ARIMA 8
3.3 LSTM 9
3.3.1 LSTM Models 10

CHAPTER FOUR - APPLICATION OF CHOSEN METHOD 14

4.1 Application Steps and Test Structures 14
4.2 Metrics 15
4.2.1 Mean Squared Error 15
4.2.2 Standard Deviation 15
4.2.3 Mean Absolute Error 16
4.2.4 Median Absolute Error 17

CHAPTER FIVE - RESULTS 18

CHAPTER SIX - DISCUSSION 36

6.1 Comparison of Results 36

CHAPTER SEVEN - CONCLUSION AND FUTURE WORK 39

7.1 Conclusion 39
7.2 Future Work 39
REFERENCES 41

LIST OF FIGURES

Page

Figure 1.1 Major fault lines...1
Figure 3.1 General structure of lstm...9
Figure 3.2 Architecture of a typical vanilla LSTM block...12
Figure 3.3 Stacked LSTM…………...13
Figure 4.1 Model Selection..………..14

LIST OF TABLES

Page

Table 5.1 8 dense-64 epoch-64 batch size full data (Best results)..............................18
Table 5.2 8 dense-64 epoch-64 batch size without depth (Best results).....................20
Table 5.3 8 dense-64 epoch-64 batch size without moon (Best results).....................21
Table 5.4 8 dense-64 epoch-64 batch size without sun (Best results)........................23
Table 5.5 32 dense-32 epoch-32batch size full data (Best results).............................24
Table 5.6 32 dense-32 epoch-32 batch size without depth (Best results)...................26
Table 5.7 32 Dense-32 Epoch-32 Batch Size without Moon (Best results)...............27
Table 5.8 32 dense-32 epoch-32 batch size without sun (Best results)......................29
Table 5.9 32 dense-32 epoch-64 batch size full data (Best results)……....................30
Table 5.10 32 dense-32 epoch-64 batch size without depth (Best results).................31
Table 5.11 32 dense-32 epoch-64 batch size without moon (Best results).................33
Table 5.12 32 dense-32 epoch-64 batch size without sun (Best results)....................34
Table 6.1 Hyperparameter tuning – results when all parameters used……………...36
Table 6.2 Hyperparameter tuning – results without the depth parameter…………...37
Table 6.3 Hyperparameter tuning - results without moon parameters........................37
Table 6.4 Hyperparameter tuning – results without sun parameters….......................38

LIST OF SYMBOLS

∅	: Coefficient of lag
∈	: Errors observed at respective lags
ϕ	: Respective Weights
μ	: Constant mean of the series
δ	: Coefficients of the estimated error
Σ	: To add up

ABBREVIATIONS

ARIMA	: Autoregressive Integrated Moving Average
LSTM	: Long Short-Term Memory
USGS	: United States Geological Survey
MAE	: Mean Absolute Error
MedAE	: Median Absolute Error
MSE	: Mean Squared Error
STD	: Standard Deviation
RNN	: Recurrent Neural Network
LAT	: Latitude
LONG	: Longitude
MAG	: Magnitude

CHAPTER ONE

INTRODUCTION

1.1 General

There are continuous earthquakes around the world. Earthquakes are divided into regions according to their occurrence in the world. In the beginning of the study, more than 3 million earthquake data belonging to 86 regions since 1970 have been gathered. The data include variables such as date time, latitude, longitude, magnitude, zones and depth. All data is cleaned and stored into MySQL database, by creating a proper database structure. Data of 29C-33A regions which are covering the major fault lines in Turkey, as shown in Figure 1.1.

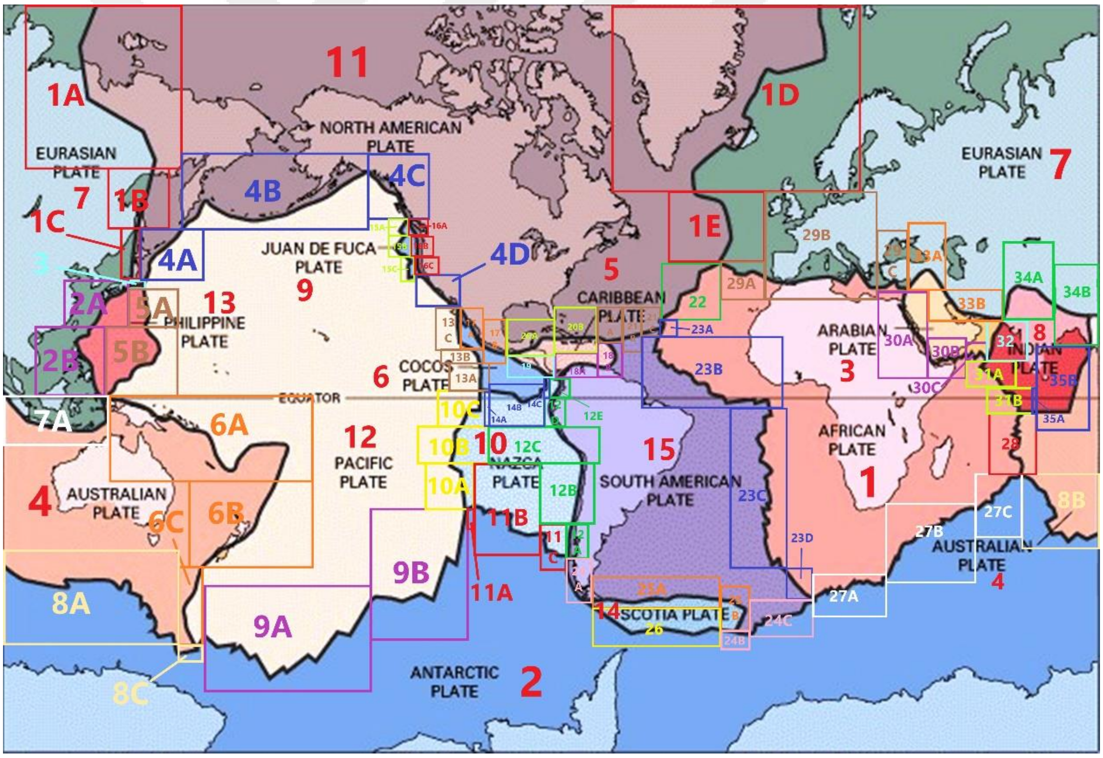

Figure 1.1 Major fault lines

Time series forecasting is used in many applications. It can be used in various areas such as airports, earthquakes, optical systems, exchanges, e-commerce sites. Several models are used in time series forecasting. In this study, especially Autoregressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) models were examined. According to the literature research, LSTM model used in other studies was chosen to compare with ARIMA (Alhirmizy, & Qader, 2019). LSTM is easy to use and suitable for tackling the earthquake prediction problem (Jozefowicz, Zaremba, & Sutskever, 2015). The general structure of LSTM is shown in Figure 1.1.

LSTM is a Recurrent Neural Network (RNN) variation. LSTM has also several variations such as univariate and multivariate models. Multivariate LSTM is used in other studies as well (Sorkun, İncel, & Paoli, 2020). Tensor Flow framework and Python's Keras library were used implementing LSTM (Du, Li, & Horng, 2018).

While preparing the algorithm, other studies using multivariate LSTM were examined and used in deciding the most effective algorithm. The data of the United States Geological Survey (USGS) were used in our study. The comparison of predictions are made using Mean-Squared-Error (MSE), Standart-Deviation (STD), Mean-Absolute-Error (MAE) and Median-Absolute-Error (MedAE) in daily latitude, longitude, magnitudes.

1.2 Purpose

In recent years, making predictions by using machine and deep learning have become popular. Comparing these methods and knowing which one would be more appropriate according to a given problem is a matter of curiosity. In this paper, the earthquakes experienced in Turkey between the years 1970-2019 will be examined. Earthquake prediction will be made using the chosen methods and the results will be compared to find out the most successful model.

1.3 Problem Definition

There are several Artificial Intelligence methods for forecasting in time series. But deciding which method fits a specific area is not trivial. Our work examines two methods for deciding which method is better in forecasting earthquakes.

1.4 Organization of the Thesis

The examined literature are given in Chapter 2. Studies similar to analysis of time series were studied. Answers to questions like on which data and what methods were used and what results were researched. The materials and methods used in the earthquake prediction studies are summarized in Chapter 2. Information about the datasets are also explained. The tools and equipment used are listed. In the Method section, ARIMA and LSTM methods are explained.

In Chapter 3, LSTM and ARIMA are introduced first then the comparison of general properties of the two methods is made. Then the internal structure of LSTM is detailed.

In Chapter 4, the designed vanilla LSTM and stacked LSTM models are compared. The reason for choosing stacked LSTM is explained. In addition, the metrics used are presented. The used metrics are Mean Squared Error, Standard Deviation, Mean Absolute Error, and Median Absolute Error.

In Chapter 5, test results with ARIMA and Stack LSTM for different Dense-Epoch-Batch size and models are given. The results are listed in detailed tables.

In Chapter 6, the results are compared to select the best performing method. The best and worst results are determined for each hyper parameter structure.

In Chapter 7, the conclusion of the study and future work are given.

CHAPTER TWO
LITERATURE REVIEW

2.1 Related Works

In study Wang, Guo, Yu, & Li, (2017) a deep learning LSTM networks is used to predict the timely and spatial relationship between earthquakes in different places. Using the simulation results, the authors showed that the LSTM network with two-dimensional input developed in the study was able to discover and use the temporal and spatial between earthquakes, thus making better predictions than before.

In their article, Berhich, Belouadha & Kabbaj, (2020) used the Moroccan data set to carry out earthquake prediction studies. The Moroccan seismic activity data set from 1900 to 2019 was obtained by the National Geophysical Institute of the National Centre for Scientific and Technical Research CNRST.

Nicolis, Plaza & Salas (2021) explored a new method based on LSTM networks and Convolutional Neural Network (CNN) to predict density function in a particular area and the probability of a seismic event of a certain density. Specifically, they first used the Chilean earthquake catalog to estimate the ETAS model on a grid of 1 × 1 degree from 2000 to 2017 using three different threshold sizes (≥ 4.0, ≥ 5.0 and ≥ 6.0). Then, LSTM was used to achieve the spatial maximum of the intensity and estimate the maximum rate at which the earthquake event occurred the next day.

Bhandarkar, Satish, Sridhar, Sivakumar & Ghosh (2019) did earthquake trend prediction with using LSTM. They trained model and predict the future trend of earthquakes. An ordinary Feed Forward Neural Network solution for the same problem was done for comparison. They found out that LSTM outperform the Feed Forward Neural Network.

In work González, Yu & Telesca, (2019), the authors proposed a novel earthquake magnitude prediction method. They modeled a time series of behavior using machine learning to measure seismic events over more than two decades. They used LSTM for time series forecasting.

2.2 Materials and Method

In this section, the materials that are used throughout the study and the methods of the times series analysis are introduced.

2.2.1 Materials

2.2.1.1 Dataset

We started the studies first by collecting data. From the United States Geological Survey (USGS), we used earthquake data from 1970-2019 in 35-47 latitude and 25-50 longitude within the scope of Turkey for 49 years. These data were divided into 86 regions according to their locations in the world and they were all created as a separate file. In addition to these data, the dates and locations of the solar and lunar eclipses that have been since 1970 have been collected in a separate file. In order to properly manage and use the collected data, it was inserted into a MySQL database in a suitable table structure. While the total number of data rows is 2.850.180, the number of rows belonging to the 29C-33A region which is used in this study is 18.918. The total data size is 555 MB. Tests on the earthquake in the regions of Turkey have in this study and experiments were performed. To use the database structure created from regions of Turkey earthquakes were exported from the database equivalent of pulling in a convenient format.

2.2.1.2 Tools

We installed the Anaconda platform for the Python language used in the algorithm that we designed in the study.

We imported the tensor flow and keras libraries in Python to be used in the designed algorithm.

All work was done on a computer with the following specifications: Intel i5-8250U CPU, 8GB DDR4 ram, AMD Radeon 530 graphics card and Lenovo v330-15IKB

2.2.2 Method

We made studies on ARIMA and LSTM used in time series analysis methods. As ARIMA works univariate, it does not fit our structure. The study conducted with ARIMA on the COVID-19 dataset was examined, it was found out that ARIMA's estimation results for shorter times are better but the results are deteriorated in monthly or longer time predictions, and the error rates increased. As a result of the examinations, we found that LSTM, which is used in other studies, is more suitable for our study (Alhirmizy, & Qader, 2019). In Study Jozefowicz et al., (2015), there is a multivariate structure. In addition, LSTM is preferred more than ARIMA for large and complex data sets (Zhang et al., 2013). After the LSTM selection, the models underneath were examined, and we made examinations on the remarkable Vanilla and Stacked LSTM. The Vanilla LSTM univariate study led us to the Stacked LSTM side. We have performed our tests by creating an algorithm in the multivariate Stacked LSTM structure with the data we have.

Like the other studies, we used different variables and Hyper parameters (Sorkun et al., 2020). Giving 3 variables from outside, tests were performed on 11 parameters in the data set. We analyzed the effect of estimation results by removing some parameters one by one from the data set, respectively. The tests resulted in approximately 10-15 hours.

CHAPTER THREE
COMPARISON OF THE GENERAL PROPERTIES OF THE TWO METHODS

3.1 Time Series

A time series is a series of data points ordered by time. There are aspects such as autocorrelation, seasonality and stationarity in time series. Auto correlation is the similarity of time-dependent results between observations over time. Seasonality shows us changes over time. A time series is said to be stationary if the mathematical properties do not change with time.

A time series with cyclical behavior but no trend or seasonality is stationary in some cases (Montgomery, Jennings, & Kulahci, 2015). A time series with independent variables and distributed with a mean of zero is called white noise. For this case, it may mean that the variables have the same variance and that the values have zero correlation with all other values in the series. This is unpredictable if a time series contains only noise. If the errors of a model contain only noise, it means that all bits of useful information are removed from the signal, leaving only random noise. Forecasting is an important step in time series analysis. These estimates can be used commercially. Estimates must be accurate. Even the smallest mistakes to be made can affect other structures of businesses using this estimation and affect business planning activities to occur incorrectly.

For example, in seasonal time series forecasts, when seasonal airport usage is estimated, errors in the forecast will affect all businesses. Time series estimations can be divided into two parts, univariate and multivariate. In univariate time series, future values can be predicted by looking at past values. In multivariate time series, on the other hand, it provides us with a forecast by using different variables other than the time series, such as future exogenous variables.

3.2 ARIMA

ARIMA is a model used for univariate time series estimation. ARIMA allows us to model time series that exhibit some kind of pattern and do not contain random white noise. The first step we need to do here is to make the time series stationary. We do this to use the delays of Arima's linear regression model as an estimator. These linear regression models are known to work best with independent and uncorrelated estimators. In non-stationary time series, it shows the situation where the variance and mean are constant over time. Time series that have come to this state are easy to predict. Time series delays and delayed prediction errors are used for implementation in ARIMA. Therefore, past values can be used to predict future values. The ARIMA model has three features: p, d, and q. The autoregression(p) part in ARIMA says that the auto aggressive model regresses the variable of interest at its lagged values. Parameter d is the minimum number of differencing needed to make the series stationary. And parameter q is the order of MA. That means that number of lagged forecast errors that should have the ARIMA model.

The corresponding AR (p) model of Y_t series, which is the generalizations of autoregressive model, can be expressed as:

$$AR(p): Y_t = \emptyset_0 + \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \cdots + \emptyset_p Y_{t-p} + \in t \ldots \quad (3.1)$$

Where, Y_t is the response variables at time t, Y_t-1, Y_t-2……Y_t-p is the respective variables at different time with lags; ϕo, ϕ1,….. ϕp are the coefficients; and εt is the error factor. Similarly, the MA (q) model which is again the generalizations of moving average model may be specified as:

$$Y_t = \mu_t + \in_1 + \delta_1 \in_{t-1} + \cdots + \delta_q \in_{t-q} + vt \ldots \quad (3.2)$$

Where, μ_t is the constant mean of the series; δ1… δq is the coefficients of the estimated error term; εt is the error term. Combining both the model is called as ARIMA models, which has general form as:

$$Y_t = \emptyset_0 + \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \cdots + \emptyset_p Y_{t-p} + \in_t + \delta_1 \in_{t-q} + \cdots + \delta_q \in_{t-q} + vt \quad (3.3)$$

If Y_t is stationary at level or I(0) or at first difference I(1) determines the order of integration, which is called as ARIMA model (Padhan, 2012).

3.3 LSTM

Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture that Hochreiter & Schmidhuber (1997) used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points (such as images), but also entire sequences of data (such as speech or video). For example, LSTM is applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and anomaly detection in network traffic or IDS's (intrusion detection systems) (AGMLS, 2009;Li, 2015;Sak, 2014;).

A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

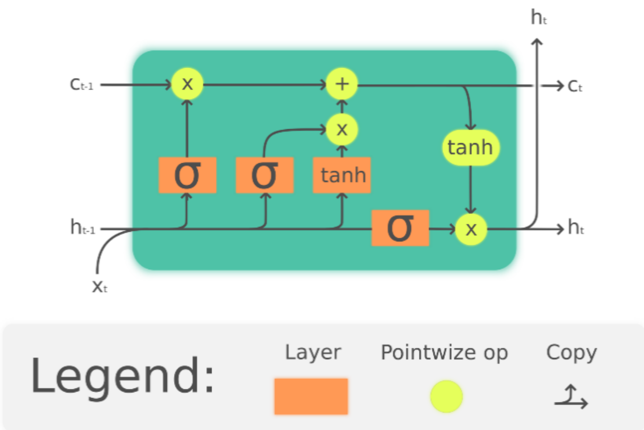

Figure 3.1 General structure of LSTM (Github, 2020)

LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. The vanishing gradient problem is a situation encountered in the training of neural networks where the gradients used to update the weights shrink exponentially. As a consequence, the weights are not updated anymore, and learning stalls. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications (Zhang et al., 2013).

3.3.1 LSTM Models

According to Brownlee (2018), there are 4 categories of LSTM models:

1. Univariate LSTM Models
    a. Vanilla LSTM
    b. Stacked LSTM
    c. Bidirectional LSTM
    d. CNN LSTM
    e. Conv LSTM
2. Multivariate LSTM Models
    a. Multiple Input Series
    b. Multiple Parallel Series
3. Multi-Step LSTM Models
    a. Vector Output Model
    b. Encoder-Decoder Model
4. Multivariate Multi-Step LSTM Models
    a. Multiple Input Multi-Step Output
    b. Multiple Parallel Input and Multi-Step Output

3.3.1.1 Bidirectional LSTM

Bidirectional LSTM is the LSTM model's ability to learn the input sequence both in its observations and correctly, and it is a combination of both learning. Bidirectional LSTMs are extensions of traditional LSTMs that can improve the performance of models in sequence classification problems. In problems where all time steps of the input sequence are available, a bidirectional LSTM trains two LSTMs on the input sequence instead of one. The first is in the input array and the second is in the reverse copy of the input array. This can provide additional context to the network and lead to a faster and more complete understanding of the problem.

3.3.1.2 CNN LSTM

CNN is a kind of neural network developed on two-dimensional image data. It is very effective in automatically extracting and learning features from one-dimensional array data.

3.3.1.3 Conv LSTM

Conv LSTM is a type of LSTM related to CNN LSTM. The convolution reading of the input is placed directly in each LSTM unit. It was developed to read two-dimensional spatial-temporal data, but can be adapted for use with univariate time series forecasting.

3.3.1.4 Multiple Input Series

A problem may have more than one input time series and an output time series connected to the input time series. The time series entered are parallel because they are in the same time steps.

3.3.1.5 Multiple Parallel Series

Multiple parallel series are used when there is more than one parallel time series and a value must be estimated for each.

3.3.1.6 Vector Output Model

LSTM can also output a vector that can be directly interpreted as a multi-step prediction. In these cases, the vector output model is used.

3.3.1.7 Encoder-Decoder Model

The Encoder-Decoder LSTM is used to predict variable length output sequences. The model is designed for translation operations and prediction problems where both input and output sequences from sequence to sequence are involved.

3.3.1.8 Multiple Input Multi-Step Output

It is used in multivariate time series where the output series is separate but dependent on the input time series and the output series needs more than one step.

3.3.1.9 Multiple Parallel Input and Multi-Step Output

It is used in parallel time series when each time series requires estimation of more than one time step.

3.3.1.10 Vanilla LSTM

Vanilla LSTM is the LSTM model with a single LSTM layer, taking an input and outputting an output. It can be used in forecasting studies that will take a single input and output. Since it is a single variable series, the number of features is one for a variable. The number of timesteps as input is the number we chose when preparing our dataset as an argument to the split_sequence() function. The shape of the input for each instance is specified in the input_shape argument in the definition of the first hidden layer.

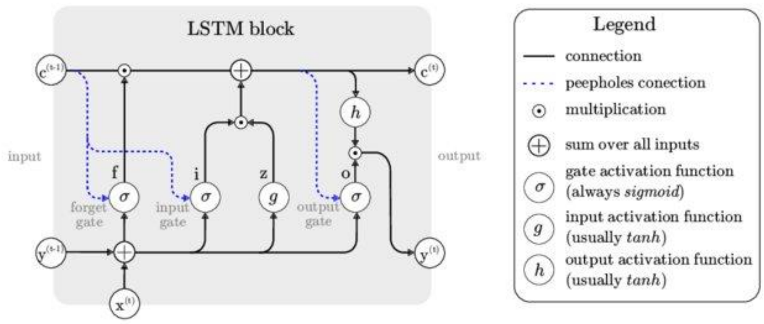

Figure 3.2 Architecture of a typical vanilla LSTM block. (G. V. Houdt, 2020)

3.3.1.11 Stacked LSTM

Multiple hidden LSTM layers are used in the stack LSTM structure. These LSTM layers are superimposed. Thanks to Stack LSTM, we don't have to have a single input and output as in vanilla LSTM. It is used in studies that will contain multiple inputs and outputs. We can solve this problem by setting the return_sequences=True argument in the layer and outputting a value for each time step in the LSTM's input data. This allows us to take 3D output from the hidden LSTM layer as input to the next one.

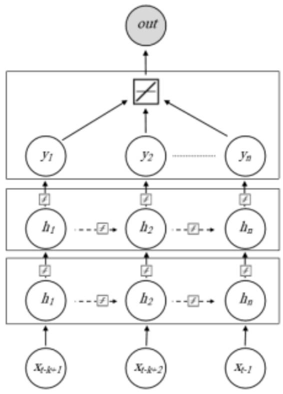

Figure 3.3 Stacked LSTM. (A. Khaled, 2018)

CHAPTER FOUR
APPLICATION OF THE CHOSEN METHOD

4.1 Application Steps and Test Structures

When we examine the literature studies on timeseries forecasting, we observed that ARIMA and LSTM models are generally used. The study with ARIMA on the COVID-19 dataset was reviewed. The ARIMA model analysis results showed that the model was correct for short-term forecasts. Significant errors occur in monthly or annual forecasts. When analyzing the data on a country-by-country basis, the margin of error has increased to the point where it is not possible to make an accurate calculation in the long run. It was concluded that the ARIMA model was able to give short-term forecast figures with very little error. It has been said that ARIMA is an acceptable model for short to medium term forecasts. In our LSTM study, earthquake data are data that are analyzed at long intervals and whose time intervals are not regular. (Er, Emeç, & Özcanhan, 2020)

In reviewed studies, it was stated that the LSTM model gave better results on complex data (Alhirmizy, 2019; Jozefowicz et al., 2015). Our work has a multivariate structure. Since ARIMA is univariate, it was not compatible with our structure.

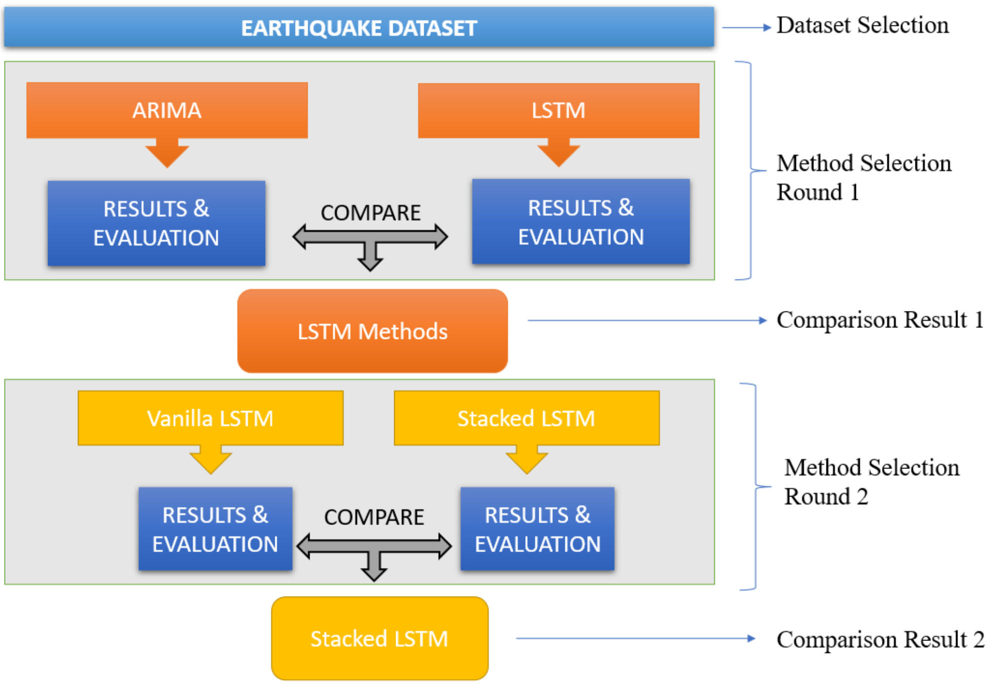

Figure 4.1 Model Selection (Personal archive, 2022)

After the LSTM selection was made, we started to examine the models under it. First of all, vanilla LSTM was used. However, vanilla LSTM did not fit our structure as it takes only one input and one output.

Since we have multiple data inputs and multiple data outputs in our study, using Stacked LSTM was the most convenient for us.

After the stacked LSTM model was selected, the best results were looked for performing hyper parameter tuning as in similar studies in the literature (Sorkun et all., 2020).

Here are 3 LSTM parameters that affect the results. These are dense, batch size and epoch values. Dense is a classic fully connected neural network layer. Each input node is connected to each output node. The batch size is a number of samples processed before the model is updated. The number of epoch is the number of complete passes through the training dataset Experiments were made on these values and the best results were obtained.

While the tests were being carried out, as mentioned before, we extracted some of the values with the hyper parameter tuning study and obtained results.

4.2 Metrics

We needed metrics to measure the results of our study and to make sense of the values. In the literature research, we preferred the metrics that are generally used in other studies like ours.

4.2.1 Mean Squared Error

Mean Squared Error (MSE) is a model evaluation metric that shows how close a fitted line is to data points, often used in regression models (Sammut & Webb, 2011). The mean squared error of a model over a test set is the mean of the squared prediction errors over all samples in the test set. Estimation error is the difference between the true value of a sample and its predicted value. N is the number of data points, Y_i the value returned by the model and $\hat{Y}_i$ the actual value for data point i.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{4.1}$$

4.2.2 Standard Deviation

The standard deviation (STD) gives a measure of the distance of the average from the mean and is used to show and explain whether scores cluster around the mean or are widely dispersed (Gravetter & Wallnau, 2017).

When you're running an experiment (or test, or survey), you're usually working with a sample— a small fraction of the population. The formula to find the standard deviation (s) when working with samples is:

$$s = \sqrt{\frac{\sum(x - \bar{x})^2}{n - 1}} \tag{4.2}$$

The Σ sign in the formula means "to add up". To solve the formula,

1. Add the numbers,
2. Square them,
3. Then divide.

N is the number of data points, x is each of the values of data and $\bar{x}$ is the mean of x.

Though the formula seems simple, it gets tedious when working with larger sample sizes as squared multiple times.

4.2.3 Mean Absolute Error

Mean Absolute Error (MAE) is an essential measure used for forecast error in the analysis of time series (Hyndman & Koehler, 2005). The mean error is a different measure that varies in two. The MAE is the average vertical distance between its true value and the best line crossings to the data. MAE is also the mean horizontal distance in terms of best horizontal line with each data point. Since the MAE value is interpretable, it is used in transportation in regression and time series problems. It is a well-mean average over a series of average-sized predictions. The MAE value can range from 0 to ∞. Scores with negative visuals as well as estimators with higher values perform better. $\frac{1}{n}\sum_{i=0}^{n}$ is the test set, f_i is the predicted value and y_i is the actual value.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| \tag{4.3}$$

4.2.4 Median Absolute Error

According to Bonnin (2017, p.53),

"The median absolute error is particularly interesting because it is robust to outliers. The loss is calculated by taking the median of all absolute differences between the target and the prediction. If ŷ is the predicted value of the ith sample and yi is the corresponding true value, then the median absolute error estimated over n samples is defined as follows:"

$$MedAE(y,\hat{y}) = median(|y_1 - \hat{y}_1|, \ldots, |y_n - \hat{y}_n|) \tag{4.4}$$

CHAPTER FIVE

RESULTS

In this section, the best 7 results of hyper parameter tuning tests made with different Dense, Epoch and Batch sizes are presented. Prediction errors are given for each of the latitude, longitude, and magnitude values.

Table 5.1 8 dense-64 epoch-64 batch size full data (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
Date	2014_05_24			2006_10_25			2015_05_2		
predict	40.289	31.813	4.393	39.562	28.998	4.497	40.651	29.306	4.301
Real	40.289	25.389	4.100	40.408	29.004	5.000	40.871	32.458	4.300
deviation ratio	**0.000**				**0.006**				**0.001**
Date	2007_04_16			2008_07_4			2014_08_11		
predict	38.962	28.365	4.234	39.749	29.240	4.375	40.797	29.888	4.302
Real	38.960	25.920	4.000	37.000	29.232	4.300	40.373	25.824	4.300
deviation ratio	**0.002**				**0.008**				**0.002**
Date	2008_07_14			2007_04_11			2007_05_8		
predict	40.558	29.485	4.361	40.281	30.899	4.401	40.218	28.818	4.397
real	40.563	48.084	4.100	37.991	30.908	4.000	40.817	27.512	4.400
deviation ratio	**0.005**				**0.009**				**0.003**
date	2007_10_8			2015_10_28			2007_10_25		
predict	38.602	27.979	4.245	39.762	27.784	4.209	39.650	28.049	4.196
real	38.610	25.720	4.100	40.820	27.765	4.100	39.335	27.731	4.200
deviation ratio	**0.008**				**0.019**				**0.004**
date	2017_02_7			2017_09_21			2006_12_10		
predict	39.518	29.007	4.292	39.811	30.512	4.330	39.351	27.759	4.395
real	39.528	26.099	4.700	39.778	30.483	4.300	38.924	27.840	4.400
deviation ratio	**0.010**				**0.029**				**0.005**
date	2017_07_22			2008_07_15			2008_05_27		
predict	40.093	30.132	4.329	40.338	27.478	4.323	39.238	28.148	4.305
real	40.110	27.183	4.500	40.369	27.448	4.300	38.370	25.450	4.300
deviation ratio	**0.017**				**0.030**				**0.005**
date	2008_04_26			2007_02_8			2006_05_6		
predict	39.806	28.382	4.224	39.795	28.676	4.383	40.925	30.194	3.994
real	39.789	25.625	4.200	40.840	28.709	4.800	41.590	25.380	4.000
deviation ratio	**0.017**				**0.033**				**0.006**
Mean Squared Error	4,184				98,06				0,1323
Standart Deviation	1,179				6,525				0,255
Mean Absolute Error	1,672				7,449				0,2594
Median Absolute Error	1,506				4,672				0,2082

The deviation ratio values specified separately for each of the three columns (latitude, longitude, magnitude) show the difference between the prediction and real values.

When full data is used with 8 Dense, 64 Epoch and 64 Batch Size parameters (Table 5.1), the best realistic results appear to be magnitude, latitude and longitude, respectively. In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the parameters given in Table 5.1, while latitude and magnitude values are close to reality, longitude values have a big margin of error. Especially for mean squared error, the margin of error is very high.

When we subtract depth values and use 8 Dense, 64 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.2). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the parameters in Table 5.2, while latitude and magnitude values are close to reality, longitude values have a big margin of error. Especially for mean squared error, the margin of error is very high.

Table 5.2 8 dense-64 epoch-64 batch size without depth (Best results)

Value	LATITUDE			LONGITUDE			MAGNITUDE		
date	2008_06_2			2009_08_5			2006_07_5		
predict	40.168	29.127	4.397	39.675	28.696	4.380	40.082	29.057	4.299
real	40.170	26.990	4.100	43.447	28.692	4.200	40.739	25.343	4.300
deviation ratio	**0.002**				**0.004**				**0.001**
date	2006_06_11			2007_12_1			2007_07_16		
predict	40.457	27.388	4.051	38.554	29.214	4.056	39.656	28.113	4.099
real	40.460	25.900	4.000	37.101	29.204	4.000	38.800	25.710	4.100
deviation ratio	**0.003**				**0.010**				**0.001**
date	2008_03_26			2008_06_20			2009_09_7		
predict	40.550	28.741	4.251	40.045	28.593	4.375	40.424	32.107	4.498
real	40.540	26.810	4.100	39.368	28.582	5.500	42.660	43.443	4.500
deviation ratio	**0.010**				**0.011**				**0.002**
date	2017_09_21			2007_02_8			2015_11_14		
predict	39.788	27.811	4.324	40.127	28.724	4.391	39.916	31.893	4.303
real	39.778	30.483	4.300	40.840	28.709	4.800	43.692	38.943	4.300
deviation ratio	**0.010**				**0.015**				**0.003**
date	2008_02_14			2014_06_29			2008_08_9		
predict	39.547	28.714	4.317	39.816	27.188	4.117	40.232	31.075	4.203
real	39.533	25.976	4.600	38.857	27.172	4.600	42.841	32.539	4.200
deviation ratio	**0.014**				**0.016**				**0.003**
date	2008_01_23			2019_02_12			2006_10_21		
predict	40.697	29.715	4.330	39.909	32.983	4.365	40.248	28.094	4.196
real	40.720	25.660	4.400	40.581	32.960	4.100	40.288	28.001	4.200
deviation ratio	**0.023**				**0.023**				**0.004**
date	2007_03_25			2007_08_25			2007_05_10		
predict	38.844	28.797	4.136	40.061	29.060	4.363	40.042	30.672	4.305
real	38.870	26.180	4.000	39.538	29.088	4.200	40.654	29.007	4.300
deviation ratio	**0.026**				**0.028**				**0.005**
Mean Squared Error	4,396				94,9				0,1373
Standart Deviation	1,18				6,395				0,2565
Mean Absolute Error	1,733				7,349				0,2674
Median Absolute Error	1,601				4,162				0,2183

Table 5.3 8 dense-64 epoch-64 batch size without moon (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	2007_12_17			2006_10_17			2014_09_4		
predict	39.343	26.466	4.112	38.879	26.902	4.251	40.503	32.305	4.502
real	39.349	26.191	4.100	39.210	26.900	4.200	41.644	32.512	4.500
deviation ratio	**0.006**				**0.002**				**0.002**
date	2008_03_15			2006_12_10			2013_09_14		
predict	39.512	28.648	4.330	39.247	27.838	4.404	40.423	31.030	4.503
real	39.501	32.951	4.100	38.924	27.840	4.400	41.335	48.638	4.500
deviation ratio	**0.011**				**0.002**				**0.003**
date	2017_02_7			2016_06_25			2006_06_12		
predict	39.543	28.926	4.319	40.617	29.219	4.299	40.562	27.654	4.397
real	39.528	26.099	4.700	40.703	29.215	4.100	40.190	25.200	4.400
deviation ratio	**0.015**				**0.004**				**0.003**
date	2006_11_18			2007_08_25			2006_12_10		
predict	39.331	27.901	4.626	39.590	29.099	4.325	39.247	27.838	4.404
real	39.350	25.340	4.600	39.538	29.088	4.200	38.924	27.840	4.400
deviation ratio	**0.019**				**0.011**				**0.004**
date	2014_08_11			2010_10_7			2006_07_5		
predict	40.348	32.636	4.479	39.613	27.552	4.210	39.492	31.736	4.304
real	40.373	25.824	4.300	43.065	27.574	4.100	40.739	25.343	4.300
deviation ratio	**0.025**				**0.022**				**0.004**
date	2017_07_22			2009_09_12			2019_01_1		
predict	38.907	27.828	4.347	39.763	27.218	4.490	39.373	28.880	4.296
real	38.880	26.050	4.000	37.515	27.075	4.900	39.423	40.564	4.300
deviation ratio	**0.027**				**0.143**				**0.004**
date	2008_04_26			2007_08_27			2018_11_30		
predict	39.761	26.957	4.364	39.383	28.949	4.395	40.149	29.107	4.305
real	39.789	25.625	4.200	37.627	29.334	4.300	40.595	28.955	4.300
deviation ratio	**0.028**				**0.385**				**0.005**
Mean Squared Error	4,366				96,09				0,1312
Standart Deviation	1,198				6,416				0,2507
Mean Absolute Error	1,712				7,411				0,2615
Median Absolute Error	1,546				4,518				0,2138

When we subtract moon values and use 8 Dense, 64 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.3). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When Table 5.3 is examined for all success metrics, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with these given parameters, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

When we subtract sun values and use 8 Dense, 64 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.4). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the parameters given in Table 5.4, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

When full data is used with 32 Dense, 32 Epoch and 32 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively /Table 5.5). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude.

Table 5.4 8 dense-64 epoch-64 batch size without sun (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	2014_05_24			2006_10_25			2015_05_2		
predict	40.289	31.813	4.393	39.562	28.998	4.497	40.651	29.306	4.301
real	40.289	25.389	4.100	40.408	29.004	5.000	40.871	32.458	4.300
deviation ratio	**0.000**				**0.006**				**0.001**
date	2007_04_16			2008_07_4			2014_08_11		
predict	38.962	28.365	4.234	39.749	29.240	4.375	40.797	29.888	4.302
real	38.960	25.920	4.000	37.000	29.232	4.300	40.373	25.824	4.300
deviation ratio	**0.002**				**0.008**				**0.002**
date	2008_07_14			2007_04_11			2007_05_8		
predict	40.558	29.485	4.361	40.281	30.899	4.401	40.218	28.818	4.397
real	40.563	48.084	4.100	37.991	30.908	4.000	40.817	27.512	4.400
deviation ratio	**0.005**				**0.009**				**0.003**
date	2007_10_8			2015_10_28			2007_10_25		
predict	38.602	27.979	4.245	39.762	27.784	4.209	39.650	28.049	4.196
real	38.610	25.720	4.100	40.820	27.765	4.100	39.335	27.731	4.200
deviation ratio	**0.008**				**0.019**				**0.004**
date	2017_02_7			2017_09_21			2006_12_10		
predict	39.518	29.007	4.292	39.811	30.512	4.330	39.351	27.759	4.395
real	39.528	26.099	4.700	39.778	30.483	4.300	38.924	27.840	4.400
deviation ratio	**0.010**				**0.029**				**0.005**
date	2017_07_22			2008_07_15			2008_05_27		
predict	40.093	30.132	4.329	40.338	27.478	4.323	39.238	28.148	4.305
real	40.110	27.183	4.500	40.369	27.448	4.300	38.370	25.450	4.300
deviation ratio	**0.017**				**0.030**				**0.005**
date	2008_04_26			2007_02_8			2006_05_6		
predict	39.806	28.382	4.224	39.795	28.676	4.383	40.925	30.194	3.994
real	39.789	25.625	4.200	40.840	28.709	4.800	41.590	25.380	4.000
deviation ratio	**0.017**				**0.033**				**0.006**
Mean Squared Error	4,184				98,06				0,1323
Standart Deviation	1,179				6,525				0,255
Mean Absolute Error	1,672				7,449				0,2594
Median Absolute Error	1,506				4,672				0,2082

If we want to reach a general conclusion with the parameters given in Table 5.5, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

Table 5.5 32 dense-32 epoch-32batch size full data (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	2008_06_2			2008_02_20			2007_04_20		
predict	40.170	30.062	4.424	39.950	28.860	4.435	40.308	28.635	4.400
real	40.170	26.990	4.100	42.684	28.860	5.200	43.188	36.036	4.400
deviation ratio	**0.000**				**0.000**				**0.000**
date	2008_07_14			2008_07_4			2018_08_4		
predict	40.562	30.064	4.302	39.771	29.221	4.291	39.241	27.640	4.301
real	40.563	48.084	4.100	37.000	29.232	4.300	37.419	36.367	4.300
deviation ratio	**0.001**				**0.011**				**0.001**
date	2014_11_24			2011_07_25			2009_07_7		
predict	38.818	28.701	4.082	40.328	27.730	4.397	39.614	28.659	4.502
real	38.820	25.660	4.700	40.802	27.752	4.400	38.236	38.765	4.500
deviation ratio	**0.002**				**0.022**				**0.002**
date	2016_10_15			2006_12_10			2008_05_23		
predict	39.000	27.925	4.289	39.419	27.818	4.348	39.991	29.437	4.303
real	38.990	27.769	4.400	38.924	27.840	4.400	43.428	47.547	4.300
deviation ratio	**0.010**				**0.022**				**0.003**
date	2008_02_14			2007_12_11			2013_08_13		
predict	39.544	27.955	4.258	39.455	28.186	4.268	40.505	29.462	4.303
real	39.533	25.976	4.600	40.862	28.163	4.000	43.122	27.491	4.300
deviation ratio	**0.011**				**0.023**				**0.003**
date	2013_11_27			2007_01_19			2011_07_25		
predict	40.830	29.687	4.344	39.405	28.270	4.384	40.328	27.730	4.397
real	40.851	27.920	4.700	37.016	28.295	5.400	40.802	27.752	4.400
deviation ratio	**0.021**				**0.025**				**0.003**
date	2017_07_22			2016_06_25			2014_03_11		
predict	40.135	27.856	4.326	40.108	29.251	4.299	40.686	31.460	4.404
real	40.110	27.183	4.500	40.703	29.215	4.100	38.394	31.810	4.400
deviation ratio	**0.025**				**0.036**				**0.004**
Mean Squared Error	4,293				116,7				0,1356
Standart Deviation	1,206				7,073				0,2638
Mean Absolute Error	1,685				8,168				0,257
Median Absolute Error	1,493				4,634				0,1823

When we subtract depth values and use 32 Dense, 32 Epoch and 32 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.6). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion using Table 5.6, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

When we subtract moon values and use 32 Dense, 32 Epoch and 32 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.7). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result.

When Table 5.7 is examined for all success metrics, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with these given parameters, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

Table 5.6 32 dense-32 epoch-32 batch size without depth (Best results)

Value	LATITUDE			LONGITUDE			MAGNITUDE		
date	2011_08_16			2006_08_24			2019_06_2		
predict	39.083	28.514	4.063	39.771	26.680	4.258	40.136	28.082	4.300
real	39.080	35.900	4.100	38.180	26.690	4.600	40.782	30.784	4.300
deviation ratio	**0.003**				**0.010**				**0.000**
date	2008_03_26			2008_07_15			2013_08_13		
predict	40.544	27.571	4.178	40.449	27.458	4.317	40.439	29.065	4.301
real	40.540	26.810	4.100	40.369	27.448	4.300	43.122	27.491	4.300
deviation ratio	**0.004**				**0.010**				**0.001**
date	2017_02_7			2016_07_17			2007_10_14		
predict	39.533	27.381	4.237	40.210	29.195	4.243	40.568	29.240	4.398
real	39.528	26.099	4.700	40.712	29.180	4.700	44.103	27.117	4.400
deviation ratio	**0.005**				**0.015**				**0.002**
date	2014_06_29			2008_07_5			2007_04_20		
predict	38.863	27.846	4.197	40.799	29.141	4.310	40.427	29.518	4.402
real	38.857	27.172	4.600	37.074	29.156	4.000	43.188	36.036	4.400
deviation ratio	**0.006**				**0.015**				**0.002**
date	2006_06_12			2007_05_10			2006_03_25		
predict	40.181	29.597	4.428	40.442	28.992	4.386	40.142	30.082	4.498
real	40.190	25.200	4.400	40.654	29.007	4.300	44.889	26.912	4.500
deviation ratio	**0.009**				**0.015**				**0.002**
date	2008_06_2			2017_09_21			2010_11_7		
predict	40.183	29.399	4.372	40.456	30.454	4.206	39.526	29.311	4.297
real	40.170	26.990	4.100	39.778	30.483	4.300	39.499	40.212	4.300
deviation ratio	**0.013**				**0.029**				**0.003**
date	2006_12_27			2016_06_25			2007_08_27		
predict	40.309	29.073	4.436	40.462	29.248	4.113	39.330	27.830	4.303
real	40.329	25.975	4.600	40.703	29.215	4.100	37.627	29.334	4.300
deviation ratio	**0.020**				**0.033**				**0.003**
Mean Squared Error	4,324				113,8				0,1358
Standart Deviation	1,169				6,888				0,2601
Mean Absolute Error	1,719				8,144				0,2611
Median Absolute Error	1,624				4,816				0,2066

Table 5.7 32 Dense-32 Epoch-32 Batch Size without Moon (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	2008_06_20			2006_06_28			2017_03_6		
predict	39.369	28.095	4.296	39.763	29.372	4.346	40.092	29.569	4.300
real	39.368	28.582	5.500	37.911	29.373	4.700	43.355	45.812	4.300
deviation ratio	**0.001**				**0.001**				**0.000**
date	2008_07_15			2007_05_10			2006_08_7		
predict	40.368	29.106	4.322	40.207	29.012	4.407	39.418	26.909	4.301
real	40.369	27.448	4.300	40.654	29.007	4.300	39.680	26.070	4.300
deviation ratio	**0.001**				**0.005**				**0.001**
date	2007_12_17			2007_06_20			2012_03_11		
predict	39.347	27.452	4.289	40.343	30.011	4.341	40.413	29.437	4.299
real	39.349	26.191	4.100	44.129	30.017	4.500	41.481	46.853	4.300
deviation ratio	**0.002**				**0.006**				**0.001**
date	2010_05_22			2007_10_25			2006_04_20		
predict	39.299	28.303	4.287	39.360	27.703	4.186	40.786	29.870	4.301
real	39.320	41.211	4.000	39.335	27.731	4.200	41.709	32.510	4.300
deviation ratio	**0.021**				**0.028**				**0.001**
date	2006_11_18			2006_10_17			2006_04_13		
predict	39.373	27.507	4.307	38.991	26.930	4.217	40.044	29.905	4.001
real	39.350	25.340	4.600	39.210	26.900	4.200	38.193	26.482	4.000
deviation ratio	**0.023**				**0.030**				**0.002**
date	2007_10_25			2015_07_25			2013_08_13		
predict	39.360	27.703	4.186	40.372	28.776	4.408	40.640	29.396	4.298
real	39.335	27.731	4.200	37.148	28.738	4.400	43.122	27.491	4.300
deviation ratio	**0.025**				**0.038**				**0.002**
date	2015_10_9			2012_05_8			2010_10_7		
predict	40.655	31.516	4.452	39.482	28.463	4.394	39.593	28.967	4.102
real	40.685	36.690	4.100	37.041	28.507	5.600	43.065	27.574	4.100
deviation ratio	**0.030**				**0.044**				**0.002**
Mean Squared Error	4,309				116,1				0,134
Standart Deviation	1,219				7,027				0,2659
Mean Absolute Error	1,681				8,165				0,2517
Median Absolute Error	1,47				4,571				0,1831

When we subtract sun values and use 32 Dense, 32 Epoch and 32 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.8). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the standard deviation this time. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is median absolute error. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with Table 5.8, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

When full data is used with 32 Dense, 32 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.9). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the parameters given in Table 5.9, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

Table 5.8 32 dense-32 epoch-32 batch size without sun (Best results)

Value	LATITUDE			LONGITUDE			MAGNITUDE		
date	1993_01_7			1993_09_1			2001_08_26		
predict	40.368	28.964	4.100	40.126	27.617	4.074	40.469	30.692	4.366
real	40.366	42.215	6.400	39.114	27.618	5.300	40.951	31.573	4.700
deviation ratio	**0.002**				**0.001**				**0.334**
date	2008_10_22			1991_06_2			1997_03_7		
predict	40.751	28.678	4.103	40.289	28.522	4.098	40.154	33.210	4.281
real	40.748	29.174	5.600	41.161	28.525	4.900	39.567	29.568	4.700
deviation ratio	**0.003**				**0.003**				**0.419**
date	2003_06_12			1984_02_25			1999_11_26		
predict	40.171	28.814	4.130	40.273	27.921	4.099	40.255	31.057	4.280
real	40.177	25.143	4.700	39.363	27.915	5.400	40.770	27.435	4.700
deviation ratio	**0.006**				**0.006**				**0.420**
date	2008_06_29			1990_11_19			2002_04_11		
predict	40.384	30.721	4.114	40.032	28.170	4.095	40.020	28.980	4.275
real	40.390	25.770	5.100	37.007	28.154	5.200	41.657	44.650	4.700
deviation ratio	**0.006**				**0.016**				**0.425**
date	2017_02_7			1990_10_24			1985_09_20		
predict	39.535	28.846	4.116	39.860	30.230	4.079	40.256	32.861	4.323
real	39.528	26.099	4.700	39.847	30.251	5.400	40.776	42.588	4.800
deviation ratio	**0.007**				**0.021**				**0.477**
date	1991_06_26			2006_10_25			1998_09_5		
predict	39.605	27.246	4.106	40.307	28.982	4.076	39.882	29.234	4.185
real	39.595	27.816	5.300	40.408	29.004	5.000	39.158	29.445	4.700
deviation ratio	**0.010**				**0.022**				**0.515**
date	1993_10_24			1996_01_28			2016_07_17		
predict	39.189	27.058	4.110	40.568	28.245	4.065	40.603	29.822	4.183
real	39.202	27.579	4.900	40.788	28.269	5.000	40.712	29.180	4.700
deviation ratio	**0.013**				**0.024**				**0.517**
Mean Squared Error	2,538				71,98				1,342
Standart Deviation	1,071				6,51				0,4989
Mean Absolute Error	1,179				5,441				1,046
Median Absolute Error	0,8012				1,859				0,9044

Table 5.9 32 dense-32 epoch-64 batch size full data (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	2007_10_4			2018_11_30			2007_10_22		
predict	40.174	29.125	4.341	39.867	28.996	4.351	39.164	27.677	4.199
real	40.170	25.040	4.200	40.595	28.955	4.300	38.391	43.408	4.200
deviation ratio	**0.004**				**0.041**				**0.001**
date	2008_07_15			2008_06_20			2006_04_9		
predict	40.356	29.269	4.303	39.709	28.634	4.363	40.253	30.414	4.399
real	40.369	27.448	4.300	39.368	28.582	5.500	42.982	48.054	4.400
deviation ratio	**0.013**				**0.052**				**0.001**
date	2010_05_22			2007_01_28			2007_03_25		
predict	39.307	27.594	4.044	39.550	28.377	4.289	38.903	29.725	3.999
real	39.320	41.211	4.000	37.870	28.290	4.500	38.870	26.180	4.000
deviation ratio	**0.013**				**0.087**				**0.001**
date	2016_10_15			2016_06_25			2007_10_25		
predict	38.969	28.374	4.271	40.372	29.121	4.328	39.434	28.494	4.202
real	38.990	27.769	4.400	40.703	29.215	4.100	39.335	27.731	4.200
deviation ratio	**0.021**				**0.094**				**0.002**
date	2016_10_15			2015_07_25			2008_04_15		
predict	38.969	28.374	4.271	40.031	28.633	4.271	39.472	28.681	4.298
real	38.990	27.769	4.400	37.148	28.738	4.400	37.280	44.940	4.300
deviation ratio	**0.021**				**0.0105**				**0.002**
date	2006_10_17			2008_07_4			2008_07_15		
predict	39.184	27.262	4.196	39.384	29.343	4.281	40.356	29.269	4.303
real	39.210	26.900	4.200	37.000	29.232	4.300	40.369	27.448	4.300
deviation ratio	**0.026**				**0.111**				**0.003**
date	2007_03_25			2015_10_28			2007_09_11		
predict	38.903	29.725	3.999	40.500	27.637	4.177	39.001	28.890	4.103
real	38.870	26.180	4.000	40.820	27.765	4.100	41.156	48.321	4.100
deviation ratio	**0.033**				**0.128**				**0.003**
Mean Squared Error	4,298				114,4				0,1368
Standart Deviation	1,193				6,991				0,2676
Mean Absolute Error	1,695				8,092				0,2553
Median Absolute Error	1,592				4,873				0,1906

Table 5.10 32 dense-32 epoch-64 batch size without depth (Best results)

Value	LATITUDE			LONGITUDE			MAGNITUDE		
date	2008_07_9			2008_08_3			2008_04_15		
predict	40.468	29.022	4.352	39.207	29.955	4.039	39.822	29.398	4.301
real	40.466	28.721	4.500	37.179	29.961	4.100	37.280	44.940	4.300
deviation ratio	**0.002**				**0.006**				**0.001**
date	2006_12_27			2008_06_20			2007_07_16		
predict	40.326	28.844	4.380	40.414	28.600	4.462	40.199	28.198	4.102
real	40.329	25.975	4.600	39.368	28.582	5.500	38.800	25.710	4.100
deviation ratio	**0.003**				**0.018**				**0.002**
date	2007_01_30			2008_01_11			2007_05_8		
predict	40.644	28.398	4.219	40.011	28.747	4.365	40.515	29.727	4.402
real	40.640	25.700	4.000	37.957	28.765	4.700	40.817	27.512	4.400
deviation ratio	**0.004**				**0.018**				**0.002**
date	2008_06_29			2008_11_5			2018_06_9		
predict	40.384	29.679	4.468	40.147	27.633	4.264	40.304	31.504	4.403
real	40.390	25.770	5.100	42.932	27.653	4.000	43.040	46.219	4.400
deviation ratio	**0.006**				**0.020**				**0.003**
date	2006_10_21			2015_04_29			2011_09_30		
predict	40.279	27.849	4.177	40.101	29.255	4.241	39.883	29.118	4.303
real	40.288	28.001	4.200	42.039	29.308	4.200	39.653	38.678	4.300
deviation ratio	**0.009**				**0.053**				**0.003**
date	2017_02_7			2017_09_21			2015_11_14		
predict	39.513	28.734	4.299	40.182	30.412	4.317	40.228	29.911	4.297
real	39.528	26.099	4.700	39.778	30.483	4.300	43.692	38.943	4.300
deviation ratio	**0.015**				**0.071**				**0.003**
date	2008_02_28			2013_08_13			2006_12_28		
predict	40.725	29.229	4.452	40.616	27.571	4.254	40.128	29.494	4.203
real	40.740	27.270	4.700	43.122	27.491	4.300	40.631	33.049	4.200
deviation ratio	**0.015**				**0.080**				**0.003**
Mean Squared Error	4,304				114,7				0,1363
Standart Deviation	1,184				6,947				0,2692
Mean Absolute Error	1,704				8,15				0,2527
Median Absolute Error	1,572				4,468				0,1805

When we subtract depth values and use 32 Dense, 32 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.10). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the given parameters in Table 5.10, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

When we subtract moon values and use 8 Dense, 64 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.11). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the mean squared error. This is the opposite for latitude. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value again is standard deviation. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with these given parameters (Table 5.11), while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

Table 5.11 32 dense-32 epoch-64 batch size without moon (Best results)

Value	LATITUDE			LONGITUDE			MAGNITUDE		
date	2006_05_31			2017_10_5			2012_03_11		
predict	40.691	30.671	4.390	39.964	28.029	4.307	40.012	28.476	4.300
real	40.697	34.631	4.800	37.169	28.042	4.100	41.481	46.853	4.300
deviation ratio	**0.006**				**0.013**				**0.000**
date	2008_03_15			2007_12_15			2007_01_28		
predict	39.487	28.289	4.347	39.577	26.570	4.376	40.114	29.488	4.500
real	39.501	32.951	4.100	38.229	26.593	4.100	37.870	28.290	4.500
deviation ratio	**0.014**				**0.023**				**0.000**
date	2008_10_19			2008_10_5			2007_04_20		
predict	39.151	28.561	4.365	39.696	28.971	4.368	40.123	28.726	4.400
real	39.136	29.389	4.500	40.650	29.017	4.400	43.188	36.036	4.400
deviation ratio	**0.015**				**0.046**				**0.000**
date	2008_07_14			2012_06_7			2013_11_16		
predict	40.579	29.983	4.357	39.480	27.978	4.320	40.870	31.559	4.400
real	40.563	48.084	4.100	40.848	27.921	4.200	42.376	40.909	4.400
deviation ratio	**0.016**				**0.057**				**0.000**
date	2014_05_24			2018_11_30			2007_08_23		
predict	40.263	28.518	4.344	40.485	28.876	4.296	39.753	28.877	4.500
real	40.289	25.389	4.100	40.595	28.955	4.300	40.636	48.518	4.500
deviation ratio	**0.026**				**0.079**				**0.000**
date	2017_06_16			2007_05_10			2016_12_13		
predict	40.615	30.162	4.329	39.431	29.101	4.319	40.140	30.927	4.400
real	40.641	47.805	4.300	40.654	29.007	4.300	40.951	48.647	4.400
deviation ratio	**0.026**				**0.094**				**0.000**
date	2007_08_25			2014_06_29			2016_11_23		
predict	39.567	28.958	4.350	39.535	27.279	4.436	39.779	29.024	4.401
real	39.538	29.088	4.200	38.857	27.172	4.600	38.546	43.777	4.400
deviation ratio	**0.029**				**0.107**				**0.001**
Mean Squared Error	4,297				116,4				0,1367
Standart Deviation	1,204				7,048				0,269
Mean Absolute Error	1,688				8,166				0,2537
Median Absolute Error	1,452				4,717				0,1883

Table 5.12 32 dense-32 epoch-64 batch size without sun (Best results)

Value	**LATITUDE**			**LONGITUDE**			**MAGNITUDE**		
date	1989_09_15			1997_12_7			1998_01_23		
predict	40.180	28.117	4.105	40.403	28.065	4.052	39.999	29.376	4.235
real	40.175	25.081	5.600	41.359	28.061	5.000	40.352	29.269	4.700
deviation ratio	**0.005**				**0.004**				**0.465**
date	1990_07_23			1979_04_28			2005_12_16		
predict	39.826	26.439	4.109	40.755	27.736	4.065	40.565	32.574	4.212
real	39.820	25.567	5.000	40.809	27.729	5.100	41.274	43.758	4.700
deviation ratio	**0.006**				**0.007**				**0.488**
date	2000_02_2			1991_03_31			2012_09_22		
predict	40.378	29.099	4.046	39.173	29.543	4.116	40.183	30.042	4.205
real	40.371	28.845	5.300	37.093	29.553	5.100	38.405	46.682	4.700
deviation ratio	**0.007**				**0.010**				**0.495**
date	2005_05_8			1994_07_7			2013_03_31		
predict	39.442	28.543	4.097	40.069	29.158	4.061	40.481	32.009	4.202
real	39.450	26.160	4.800	39.296	29.169	5.000	42.694	46.798	4.700
deviation ratio	**0.008**				**0.011**				**0.498**
date	1994_03_21			1999_08_9			2013_03_26		
predict	40.453	28.768	4.079	40.426	28.916	4.087	40.522	33.318	4.200
real	40.466	29.167	5.100	41.175	28.934	4.900	43.219	41.637	4.700
deviation ratio	**0.013**				**0.018**				**0.500**
date	1986_10_30			1990_10_2			2008_01_11		
predict	39.780	29.327	4.101	39.174	27.623	4.091	38.906	30.253	4.199
real	39.761	28.761	5.000	39.101	27.643	5.100	37.957	28.765	4.700
deviation ratio	**0.019**				**0.020**				**0.501**
date	1998_09_17			1998_04_18			1997_03_7		
predict	39.682	29.235	4.106	39.422	27.235	4.125	40.397	29.789	4.190
real	39.663	29.437	4.700	37.663	27.264	4.700	39.567	29.568	4.700
deviation ratio	**0.019**				**0.029**				**0.510**
Mean Squared Error	2,507				70,16				1,346
Standart Deviation	1,084				6,464				0,4998
Mean Absolute Error	1,154				5,328				1,047
Median Absolute Error	0,7302				1,731				0,9011

When we subtract sun values and use 32 Dense, 32 Epoch and 64 Batch Size parameters, the best realistic results appear to be magnitude, latitude and longitude, respectively (Table 5.12). In addition, when we look at the success metrics for the magnitude, it seems to get the best result with the standard deviation. Looking at the success metrics for Latitude, the worst value is mean squared error, while the best value is median absolute error. Likewise, mean squared error gave the worst value for longitude, while median absolute error gave the best result. When the table for all success metrics is examined, all the lowest error rates are gathered in magnitude. If we want to reach a general conclusion with the given parameters in Table 5.12, while latitude and magnitude values are close to reality, longitude values have quite a margin of error. Especially for mean squared error, the margin of error is very high.

CHAPTER SIX

DISCUSSION

6.1 Comparison of Results

The tests were done with different epoch, dense and batch sizes. In the tests performed, we made predictions for latitude, longitude and magnitude in earthquakes above 4. All tests work on %80 training. These tests and their results can be seen below.

Table 6.1 Hyperparameter tuning – results when all parameters used.

Model Number	Number of neurons	Number of epochs	Batch Size	MSE	STD	MAE	MEDAE	Order
1	32	32	32	Lat[5]:4,293 Long[6]:116,7 Mag[7]:0,135	Lat:1,206 Long:7,073 Mag:0,26	Lat:1,685 Long:8,16 Mag:0,26	Lat:1,493 Long:4,63 Mag:0,18	Inaccurate
2	32	64	32	Lat:4,298 Long:114,4 Mag:0,136	Lat:1,193 Long:6,99 Mag:0,26	Lat:1,695 Long:8,09 Mag:0,27	Lat:1,592 Long:4,87 Mag:0,19	Average
3	8	64	64	Lat:4,184 Long:98,06 Mag:0,132	Lat:1,179 Long:6,525 Mag:0,255	Lat:1,672 Long:7,44 Mag:0,25	Lat:1,5 Long:4,67 Mag:0,20	Accurate

In our first tests, we worked with different neuron, epoch and batch size values without removing any parameters. Results show that, model number 3 gave us the best results for Mean Squared Error in the test results used. In addition, it gave better results in all latitude, longitude and magnitude values compared to Mean Squared Error. When looking for Standard Deviation, model 3 gave us the best results. When we examined the results, it gave better results in all latitude, longitude and magnitude values according to Standard Deviation. When looking for the Mean Absolute Error, model 3 gave us the best results. When we examined the results, it gave better results in all latitude, longitude and magnitude values according to Mean Absolute Error. When looking for the Median Absolute Error, model 1 gave us the best results. When we examined the results, it gave better results in all latitude, longitude and magnitude values according to Median Absolute Error. When we look at the horizontal success metrics, model 3 has the best performances for in MSE, STD, MAE. Unfortunately, MEDAE is not the best. When we look at the vertical success metrics longitude has a very large error rate. Magnitude has best performance in model 3.

Table 6.2 Hyperparameter tuning – results without the depth parameter

Model Number	Number of neurons	Number of epochs	Batch Size	MSE	STD	MAE	MEDAE	Order
2	32	64	32	Lat:4,304 Long:114,7 Mag:0,136	Lat:1,184 Long:6,947 Mag:0,26	Lat:1,704 Long:8,15 Mag:0,26	Lat:1,572 Long:4,46 Mag:0,20	Inaccurate
1	32	32	32	Lat:4,324 Long:113,8 Mag:0,135	Lat:1,169 Long:6,888 Mag:0,26	Lat:1,719 Long:8,14 Mag:0,25	Lat:1,624 Long:4,81 Mag:0,18	Average
3	8	64	64	Lat:4,396 Long:94,9 Mag:0,132	Lat:1,18 Long:6,395 Mag:0,256	Lat:1,733 Long:7,34 Magn:0,25	Lat:1,601 Long:4,16 Mag:0,21	Accurate

In our second tests, we removed the depth parameter from the data set and worked with different neuron, epoch and batch size values. The worst performance is in longitude prediction. Because it has got the worst error rates. Again model 3 is the best in predicting magnitude and latitude.

Table 6.3 Hyperparameter tuning - results without moon parameters.

Model Number	Number of neurons	Number of epochs	Batch Size	MSE	STD	MAE	MEDAE	Order
2	32	64	32	Lat:4,297 Long:116,4 Mag:0,136	Lat:1,204 Long:7,047 Mag:0,26	Lat:1,688 Long:8,16 Mag:0,25	Lat:1,452 Long:4,71 Mag:0,18	Inaccurate
1	32	32	32	Lat:4,309 Long:116,1 Mag:0,135	Lat:1,218 Long:7,027 Mag:0,26	Lat:1,681 Long:8,16 Mag:0,25	Lat:1,47 Long:4,57 Mag:0,18	Average
3	8	64	64	Lat:4,366 Long:96,09 Mag:0,132	Lat:1,198 Long:6,416 Mag:0,250	Lat:1,712 Long:7,41 Mag:0,26	Lat:1,546 Long:4,51 Mag:0,21	Accurate

In our third tests, we removed the moon parameters from the dataset and worked with different neuron, epoch and batch size values. The worst performance is in longitude prediction. Because it has got the worst error rates. Again model 3 is the best in predicting magnitude and latitude.

Table 6.4 Hyperparameter tuning – results without sun parameters.

Model Number	Number of neurons	Number of epochs	Batch Size	MSE	STD	MAE	MEDAE	Order
3	8	64	64	Lat:2,751 Long:65,13 Mag:1,22	Lat:1,095 Long:6,017 Mag:0,54	Lat:1,246 Long:5,378 Mag:0,96	Lat:0,93 Long:2,31 Mag:0,81	Average
1	32	32	32	Lat:2,538 Long:71,98 Mag:1,34	Lat:1,071 Long:6,51 Mag:0,49	Lat:1,179 Long:5,44 Mag:1,04	Lat:0,80 Long:1,85 Mag:0,90	Average
2	32	64	32	Lat:2,507 Long:70,16 Mag:1,34	Lat:1,084 Long:6,46 Mag:0,49	Lat:1,154 Long:5,32 Mag:1,04	Lat:0,73 Long:1,73 Mag:0,90	Accurate

In our fourth tests, we removed the sun parameters from the dataset and worked with different neuron, epoch and batch size values. Without the sun parameter model 1 and 3 are same but model 2 has the best error rates. The MEDAE error is not the best in model 2. But the difference in error rates is around 0.01. Which does not affect our model choice.

When the average of MSE, STD, MAE and MEDAE is taken model 3 is the best model given the accurate results, in this study on earthquake data.

CHAPTER SEVEN
CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this study, we compared the results of four different parameter which is full data, without depth values, without sun values and without moon values, structures with three different which is using by 8 dense, 64 epochs and 64 batch size, 32 dense, 32 epochs and 32 batch size and 32 dense, 64 epochs and 32 batch size study techniques. When each parameter change was made, the tests were repeated for all 4 hyper parameter states and the results were obtained. The best results were obtained by using six input parameters (earthquake magnitude, longitude, latitude, depth, moon distance and sun distance to earth). The best values came in 8 dense, 64 epochs, 64 batch sizes. While the longitude prediction was way off the real location of the earthquakes, the best predicted parameter was the magnitude and latitude of the earthquake. As a conclusion, in the hyperparameter tests, the removal of the depth, sun and moon parameters adversely affected the test results, while the tests in which all parameters were kept gave the best results. Increasing the dense parameter with epoch and batch size affected the overall results badly. Keeping the dense value small and increasing the epoch and batch size gave the best results.

7.2 Future Work

In this study, first of all, LSTM and ARIMA comparisons were made on the earthquake dataset. Since the earthquake data are not events that occurred at a certain time, LSTM was selected and LSTM models were examined. Stacked LSTM is used because it is found appropriate, but there are many models that can be used under LSTM. With different models and different LSTM layers, studies can be done again and results can be obtained. In addition, various improvements can be made in the parameters and optimization studies can be carried out. Dense, epoch and batch size parameters can be increased. Increasing these parameters will require a more powerful machine. Different parameters can be tested by using highly equipped machines. The range of 1970 and 2019 used can be narrowed and new studies can be tried by getting faster results in a shorter range. In addition, not only the Turkish region, but also different earthquake regions can be included in the study separately and region-based results can be obtained.

REFERENCES

Alhirmizy, S., & Qader, B. (2019, March). Multivariate time series forecasting with lstm for madrid, spain pollution. *International Conference on Computing and Information Science and Technology and Their Applications (ICCISTA)* 1-5. https://doi.org/10.1109/ICCISTA.2019.8830667

Althelaya, K. A., El-Alfy E. M., & Mohammed S. (2018) Stock Market Forecast Using Multivariate Analysis with Bidirectional and Stacked (LSTM, GRU). *Saudi Computer Society National Computer Conference (NCC),* 1-7. https://doi.org/10.1109/NCG.2018.8593076.

Berhich, A., Belouadha, F. Z., & Kabbaj, M. I. (2020, March). LSTM-based models for earthquake prediction. *Proceedings of the 3rd International Conference on Networking, Information Systems & Security* (NISS2020). Association for Computing Machinery, New York, USA, 1–7. https://doi.org/10.1145/3386723.3387865

Bonnin, R. (2017). *Machine learning for developers: Uplift your regular applications with the power of statistics, analytics, and machine learning.* Packt Publishing Ltd.

Browlee, J. (2020, December 21). *Deep learning for time series forecasting.* https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/

Chang, W., Ji, X., Wang, L., Liu, H., Zhang, Y., Chen, B., & Zhou, S. (2021). A machine-learning method of predicting vital capacity plateau value for ventilatory pump failure based on data mining. *Multidisciplinary Digital Publishing Institute, 9*(10), 1306. https://doi.org/10.3390/healthcare9101306

Du, S., Li, T., & Horng, S. J. (2018, December). Time series forecasting using sequence-to-sequence deep learning framework. *9th International Symposium on Parallel Architectures, Algorithms and Programming* (PAAP), 171-176. https://doi.org/10.1109/PAAP.2018.00037

Er, B., Emeç, M., & Özcanhan, M. H. Analysis of covid-19 data using arima time series model. *V. International Scientific and Vocational Studies Congress – Engineering*. https://www.researchgate.net/publication/348282868_Analysis_of_COVID-19_Data_Using_Arima_Time_Series_Model

Gravetter, F. J., Wallnau, L. B., Forzano, L. A. B., & Witnauer, J. E. (2020). *Essentials of statistics for the behavioral sciences*. Cengage Learning.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2008). A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, *31*(5), 855-868. https://doi.org/10.1109/TPAMI.2008.137

Guillaume C. (2020, December 15). *Linear attention recurrent neural network.* https://github.com/guillaume-chevalier/Linear-Attention-Recurrent-Neural-Network

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hyndman, R. & Koehler A. (2005). Another look at measures of forecast accuracy. *International journal of forecasting*, *22*(4), 679-688.

Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015, June). An empirical exploration of recurrent network architectures. *International Conference on Machine Learning, 37*, 2342-2350.

Li, X., & Wu, X. (2015, April). Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. *International Conference on Acoustics, Speech and Signal Processing*, 4520-4524. https://doi.org/10.1109/ICASSP.2015.7178826

Montgomery, D. C., Jennings, C. L., & Kulahci, M. (2015). *Introduction to time series analysis and forecasting*. John Wiley & Sons.

Moon Calc (2020, December 15). http://www.mooncalc.org

Nicolis, O., Plaza, F., & Salas, R. (2020). Prediction of intensity and location of seismic events using deep learning. *Spatial Statistics*, *42*, 100442. https://doi.org/10.1016/j.spasta.2020.100442

Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*. https://doi.org/10.48550/arXiv.1402.1128

Sammut, C., & Webb, G. I. (Eds.). (2011). *Encyclopedia of machine learning*. Springer Science & Business Media. https://doi.org/10.1007/978-0-387-30164-8

Sorkun, M. C., İncel, Ö. D., & Paoli, C. (2020). Time series forecasting on multivariate solar radiation data using deep learning (LSTM). *Turkish Journal of Electrical Engineering & Computer Sciences*, *28*(1), 211-223. https://doi.org/10.3906/elk-1907-218

Van Houdt, G., Mosquera, C., & Napoles, G. (2020). A Review on the Long Short-Term Memory Model. *Artificial Intelligence Review*, *53*(1). https://doi.org/10.1007/s10462-020-09838-1

Wang, Q., Guo, Y., Yu, L., & Li, P. (2017). Earthquake prediction based on spatio-temporal data mining: an LSTM network approach. *Transactions on Emerging Topics in Computing*, *8*(1), 148-158. https://doi.org/10.1109/TETC.2017.2699169

Zhang, X., Liu, Y., Yang, M., Zhang, T., Young, A. A., & Li, X. (2013). Comparative study of four-time series methods in forecasting typhoid fever incidence in China. *PloS one*, *8*(5), e63116. https://doi.org/10.1371/journal.pone.0063116