

ABSTRACT

Python programming language is reliable enough to solve Machine Learning and Deep Learning problems. This paper describes how to solve earthquake prediction problems using the Python programming language that runs in the Jupyter Notebook environment. With the python library used, namely Keras. Deep Learning programming for this earthquake prediction system is the following programming sequence: data preparation, Keras model determination, Keras model compilation, Keras model adjustment, Keras model evaluation, and prediction system creation. From the test results of the earthquake prediction system using the python programming language, the results are quite satisfying. The simulation results show the results of the Deep Learning training process for the prediction system of b-value as an earthquake precursor with several iterations of 10,000 times, the results of MSE, RMSE, MAPE, and the percentage of successful predictions are 5.43×10^{-5} ; 0.00737; 0.80897 and 99.19% respectively. The results of the Deep Learning testing process for the b-value prediction system as an earthquake precursor which was carried out during the five tests obtained an average of MSE, RMSE, MAPE and the percentage of successful predictions was 0.03886; 0.19003; 23.96459, and 77.75%.

Introduction

Python is a multipurpose interpretive programming language. Unlike some other languages which are relatively difficult to read and understand, python places more emphasis on code readability to make it easier to understand the syntax. This makes Python very easy to learn for both beginners and those who have mastered other programming languages. Some many modules and libraries can be used to implement Machine Learning and Deep Learning in Python. As in this paper, the Keras library is used for deep learning programming. And in particular, deep learning in this paper is tested to solve the b-value prediction system as an earthquake precursor.

Nowadays machine learning is the branch of computer science that studies algorithm design that can be learned. Deep Learning is a sub-field of machine learning development which is a set of algorithms inspired by the structure and function of the brain. This algorithm is usually called an Artificial Neural Network (ANN). Deep learning is a development of ANN one of the hottest fields in data science with many case studies that have had amazing results in the fields of robotics, image recognition, and Artificial Intelligence (AI).

One of the most reliable and easy-to-use open-source Python libraries for developing and evaluating deep learning models in Keras; It wraps up the efficient numerical computation libraries Theano and TensorFlow. The advantage is that neural networks are easy to implement and fun. And it's possible to define and train a neural network model in just a few lines of code. Lots of deep learning programming applications use the Keras library, especially for prediction systems. Some of them can be mentioned among others: Deep learning-based prediction of species-specific protein Sglutathionylation sites (Li et al., 2020). Portfolio optimization with return prediction using deep learning and machine learning (Ma et al., 2021). The molecular structure incorporated a deep learning approach for accurate interfacial tension predictions (Yang et al., 2020). Prediction and analysis of COVID-19 positive cases using deep learning models: A descriptive case study of India (Arora et al., 2020). DeepPPSite: A deep learning-

based model for analysis and prediction of phosphorylation sites using efficient sequence information (Ahmed et al., 2021). An end-to-end model for rice yield prediction using deep learning fusion (Chu & Yu, 2020). Tool wear mechanism and prediction in milling TC18 titanium alloy using deep learning (Ma et al., 2020). Predictions for COVID-19 with deep learning models of LSTM, GRU, and Bi-LSTM (Shahid et al., 2020). ETH analysis and predictions utilizing deep learning (Zoumpakas et al., 2020). Occupant-centric miscellaneous electric loads prediction in buildings using state-of-the-art deep learning methods (Das et al., 2020).

Material and Methods

As described in the introduction, this paper will show that python is effective enough to complete deep learning programming, with an example of an earthquake prediction system. The prediction referred to in this paper is a prediction of the b-value as an earthquake precursor. The data used in this study are earthquake data from the catalog of the International Seismological Center (ISC) Sumatra-Andaman region, which includes the boundaries of 92 ° - 106 ° East Longitude (EL) and 6.5 ° South Latitude (SL) - 8 ° North Latitude (NL), the period January 1973 - November 2014. Magnitude greater than 3.0 SR, with a depth of less than 300 km (Rahmat et al., ____).

Under the dataset used, which consisted of a 444-month earthquake, 12 x 32 months, or 384 months were used for the training process. And the remaining 60 months are used for the validation process. Furthermore, from this data structure, each is designed for training flowcharts and deep learning testing flowcharts as in Figure 1 and Figure 2.

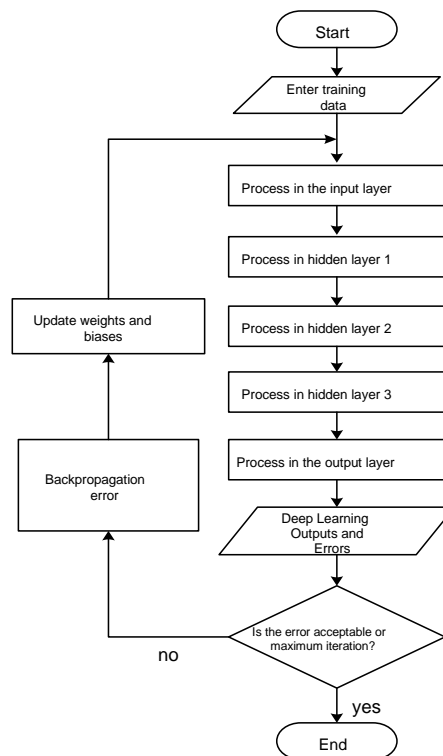


Figure 1. Deep Learning training process flow chart

From Figure 1 it can be described as follows. Where the Deep Learning network is trained using training data in the form of pairs of input and output data according to the training data model. In the input layer, the training data is processed with weights and input bias using the ReLu activation function. Proceed to the process in hidden layers 1, 2, and 3, with hidden and

biased layer weights using the ReLu activation function. Until the output layer is used the output and bias weights and the sigmoid activation function. Until the Deep Learning output is obtained.

Furthermore, the Deep Learning output is compared with the target or desired output, so a difference or error is obtained. Furthermore, this error is attempted to decrease (decrease) using gradient descent with the Error backpropagation (EBP) method. From the EBP new weights and biases are obtained. With these new weights and biases, the process is repeated for the next iteration. Thus, this process is repeated until the error or the difference between the output and the target is acceptable or the maximum iteration is reached.

Based on the flow chart of the Deep Learning training process in Figure-1, after the error or the difference between the Deep Learning output and the target has been accepted or the maximum iteration has been reached, then the weights and bias of the results of this training are stored. Henceforth, if the Deep Learning network is given new data, it is processed at each layer until the Deep Learning network output is obtained which is called prediction. The results of this prediction will be used later in this study to predict the b-value as a parameter that is believed to be a precursor to an earthquake. Thus, the flow chart of the Deep Learning testing process is as shown in Figure 2.

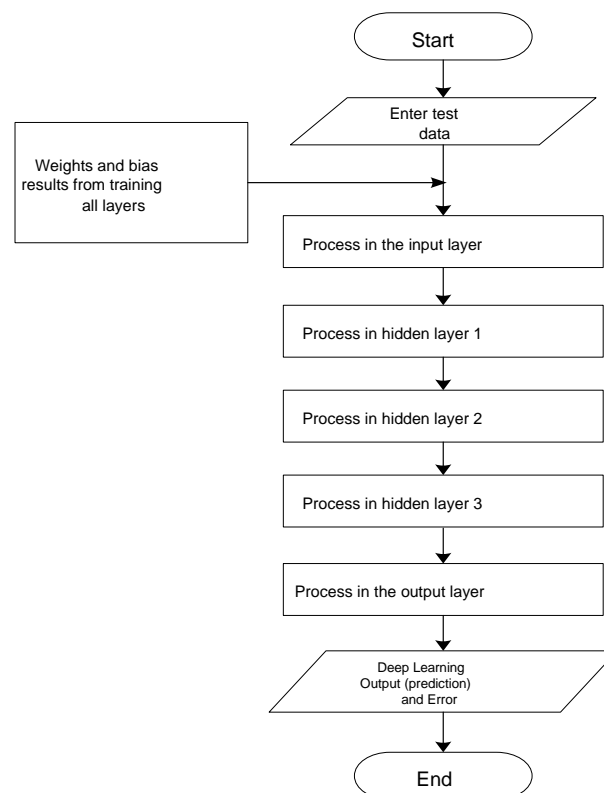


Figure 2. Deep learning testing process flow chart

Furthermore, as previously mentioned, the earthquake prediction program using the Deep Learning method will be created using the Python programming language with the Keras library. The design of the earthquake prediction program with Deep Learning using Keras is designed with the following programming sequence: Data Preparation, Determination of Keras Model, Compile Keras Model, Keras Model Adjustments, Keras Model Evaluation, and Making predictions, as shown in Figure 3.

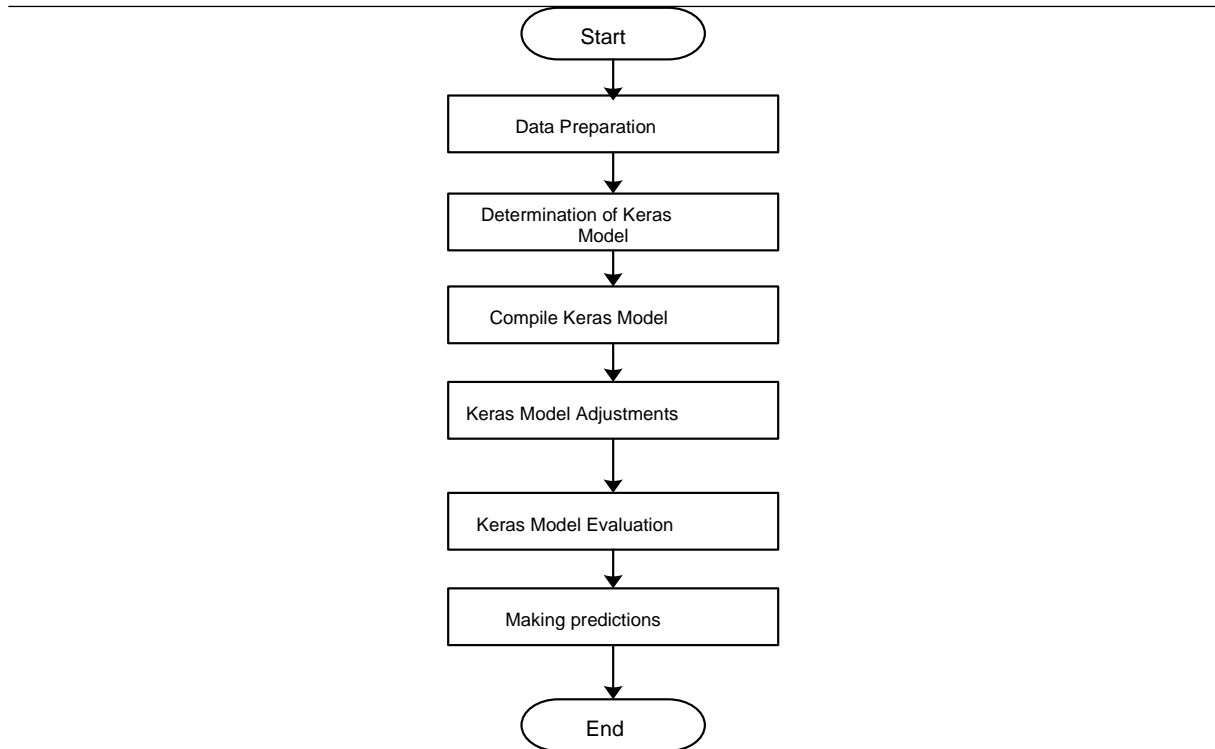


Figure 3. Earthquake prediction programming flowchart based on Deep Learning using Keras

Results and Discussion

For testing python programming for a deep learning-based earthquake prediction system with the Keras library, the results areas in the following description. The deep learning system network training process with 31 inputs and 1 output, using b-value input data from December 1973 to November 2008. The target data or the desired output is data from December 2008 to November 2009. Results the training process using 10,000 iterations, as shown in Figure 4.

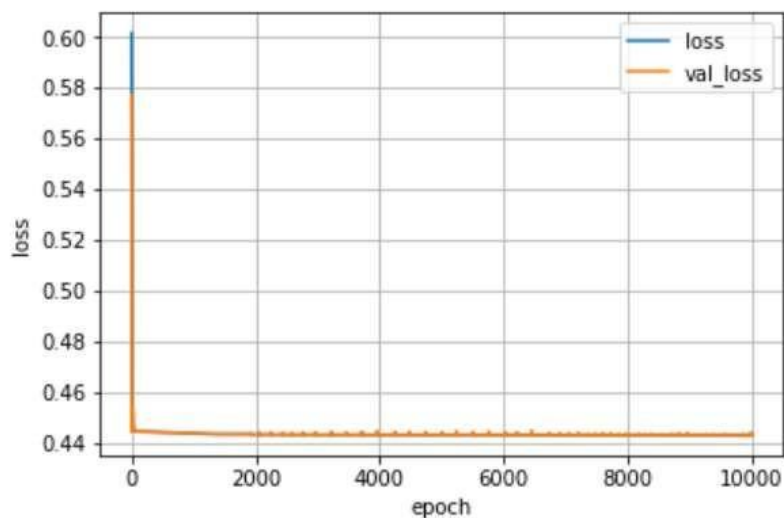


Figure 4. The curve of reducing error results from the Deep Learning training process

From the results of the Deep Learning training process for the b-value prediction system as an earthquake precursor with several iterations of 10,000 times. The results of Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and

the percentage of successful predictions are obtained 5.43×10^{-5} ; 0.00737; 0.80897 and 99.19% respectively. In summary, presented in tabular form, as shown in Table 1.

Table 1. The results of the Deep Learning training process

Number of Iterations	MSE	RMSE	MAPE	Success Percentage (%)
10.000	5.43×10^{-5}	0,00737	0,80897	99,19

Furthermore, for testing, tested 5 times. For example in the first test, the Deep Learning network system was given b-value input data from December 1974 to November 2009. It is used to predict the b-value as a precursor to earthquakes for the next year, from December 2009 to November 2010. An example of the prediction results is shown in Figure-5. And as a whole, a series of tests for the prediction of the b-value as an earthquake precursor which was carried out five times the test obtained an average of MSE, RMSE, MAPE, and the percentage of successful predictions were 0.03886; 0.19003; 23.96459, and 77.75%. Completely, each test result and the average test result are presented in Table 2.

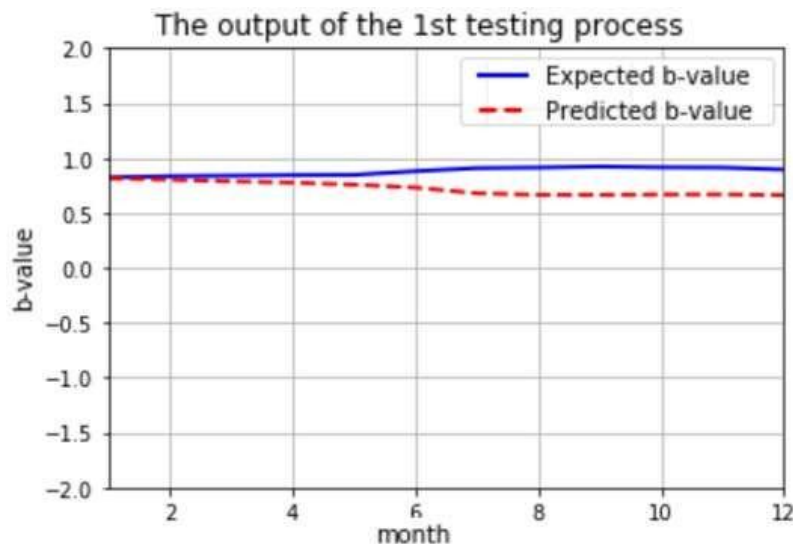


Figure 5. The output of the 1st Deep Learning testing process

Table 2. The results of the Deep Learning training process

Testing	MSE	RMSE	MAPE	Success Percentage (%)
1	0,03286	0,18128	17,21536	82,78
2	0,05779	0,24039	26,10020	82,46
3	0,01891	0,13752	17,53647	82,46
4	0,06755	0,25991	42,35427	57,65
5	0,01718	0,13106	16,61665	83,38
Average	0,03886	0,19003	23,96459	77,75

Conclusion

Python programming language is reliable enough to solve machine learning and deep learning problems. With the python library used, namely Keras, deep learning programming for the earthquake prediction system is implemented in the following stages: data preparation, determining the Keras model, compiling the Keras model, adjusting the Keras model, evaluating the Keras model, and making a prediction system. From the test results of the earthquake prediction system using the Python programming language, the results are quite satisfactory.