CSCI 6515 Machine Learning with Big Data Final Project Report

Vigneshwari Ravichandran, *B00830787*

Index Terms—Machine Learning, Regression, Neural Networks, Decison Tree, Big Data, Volume, Variety, Score

1 Overview

MACHINE Learning is, currently, a revolutionary field in computer science. With the ever-demanding increase in the amount of data being generated every day, there comes a need to apply machine learning to work on this exponentially growing amount of data. The scope of this project is to pick up such a large dataset that conforms to the specifications of what big data is and apply suitable machine learning models to it to achieve the desired results. Five major steps were taken to develop this project and it is explained in detail in the below sections.

2 Step One: Project Idea and Relevance with Big Data

This section elaborates on how the project idea was chosen, how the data was resourced, and its conformance to big data standards. The further steps taken to work on this dataset are also explained here.

2.1 Idea

The knowledge of house rents when anyone moves to a new place is an important need for survival. As a student, I had faced the need for a platform that would predict the house rent based on my description of the living space that I wanted. With this in mind, I looked for datasets that could fit this goal and found a dataset that holds the rental information for different types of rental units in the United States. The data is present for all states in the country. The given problem statement is to choose and build an appropriate machine learning model that would predict the rent of a unit given the required specifications. Further, the same model, that is being developed in this paper, can be applied to any other country or city, if an appropriate data set containing prices and specifications about rental units in that place is presented.

2.2 Dataset

The selected dataset was retrieved from Kaggle's dataset collection. It contains the rental unit specifications such as the number of bedrooms and bathrooms, parking and laundry options, wheelchair, electric vehicle charge and pet privileges, unit area, price and type, and furnished status. The rental units listed in this dataset are those in the United States of America. The latitude and longitude of each unit are mentioned along with its description, region, and state.

2.3 Big Data V's

The given dataset conforms to most of the Vs from the five V's required to denote any data as Big Data. The size of the dataset is extremely large. The original file was 371 MB and after cleaning it was 10 MB. This satisfies the Volume constraint of the Big Data's Vs. The data is present not for any particular region, but all of the states in the United States of America. Also, the data is listed for different kinds of rental

units such as a house, apartment, townhouse, studio flat, and so on. This illustrates the Variety constraint for Big Data. The source of the data is mentioned in the datasets confirming its authenticity and also showing the veracity property of Big Data. And majorly this data has a lot of value as it can be transformed into a suitable third-party business using a mobile application or a website. This application can serve as a platform for anyone who is looking for a home to find all that they need in one centralized place.

2.4 Preprocessing

The dataset retrieved above was then cleaned and preprocessed before being applied to the machine learning models. The CSV file was processed as a Pandas Dataframe and unnecessary columns to the model such as columns containing URLs and description of the rental unit were dropped. The description was dropped as text analysis of the description is not compulsorily needed to predict the rent of a house.

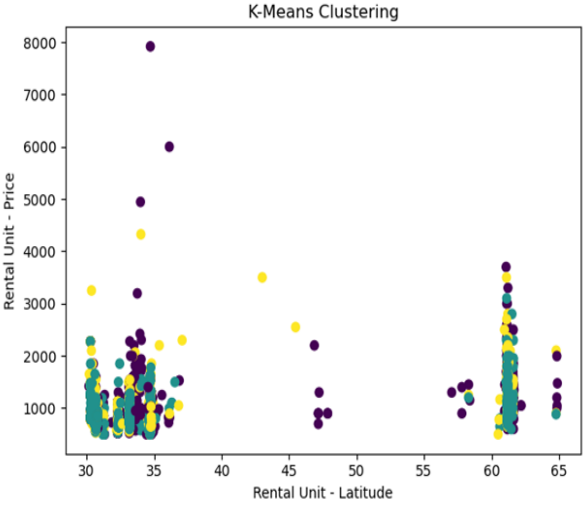
Next, any row with any of its column values as null was removed. This reduced the number of rows that are to be processed by a hundred thousand. For any machine learning algorithm to run, categorical i.e., text columns need to be either vectorized or encoded before applying the dataset to the model. Here the text columns were the ones the held information about the state, region, parking and laundry options, and unit type. To encode these, the label encoding technique was selected, and it was applied to each of the categorical columns. This encoding technique is part of python's sklearn library. It assigns a numerical label to each category of text available on that list of data. As the above columns' data has categorical values, this was a suitable technique to work with.

The final data retrieved from this had around a hundred and sixty thousand rows. This final dataframe was converted to a CSV and then passed to other steps of the project. Most machine learning models ran into a memory error with this. Hence, for this project, I have taken a thousand to five thousand rows to work with each of the machine learning models. Topics Involved.

Further investigation of this problem involved clustering the data, applying machine learning models to classify the data, and present the data in form of visualization. Matplotlib and Plotly libraries of python were used to provide appropriate data visualizations wherever necessary. Classification of data was done through regression as the data is in a numerical format. This is explained in detail in the sections below along with the machine learning models used as solutions to the given project idea. Clustering helps grouping relevant data together. This is briefly explained in the section below.

2.5 Clustering

The cleaned CSV obtained as a result of preprocessing was processed as a pandas dataframe. The K-Means clustering algorithm was applied to this dataset to cluster them into three cluster groups. A score was computed for the accuracy in clustering by using the silhouette score metrics from the sklearn library. The score improves with a larger number of clusters. The cluster results were presented as a dictionary with the name of the model, score, and the obtained cluster values. A scatter plot to show how the data is clustered with respect to its rental unit price and the location is displayed below.



After clustering, the data was then passed through the various machine learning models to predict the prices of the various households.

3 Step Two: Available Solutions

The given project involves predicting the price of a rental unit, given the various factors that might affect a user's decision and the price of that unit. The data is retrieved from preprocessing and clustering is in a numerical format. The best way to predict numerical data is by using regression. Regression is a supervised machine learning algorithm that focuses on predicting outcomes of continuous data (numeric). There are multiple kinds of regression algorithms available. These algorithms can be applied to the dataset using python's sklearn library. For this paper, I have considered the following machine learning regression algorithms to predict the house price. For each of the algorithms, the factors determining the price of the house are presented and a predicted value with an accuracy score is returned as a result. The algorithms that have been considered are listed as follows.

There are a lot of other regression models that will be considered as a good fit for prediction. I have chosen the most frequently used and easy to comprehend regression models for this project. The algorithms that have been considered are listed as follows

- Linear Regression
- Polynomial Regression
- Decision Tree Regression
- Random Forest Regression
- Neural Networks Regression

Each of these models, why it was chosen and the observed results are explained in detail below.

3.1 Linear Regression

Linear regression is a model that solves the given problem mathematically by a linear form. A straight line is plotted, and the predicted values conform to above and below this straight line. For the given data, linear regression achieved maximum accuracy.

Advantages

- It is fairly easy to implement and understand. The results can also be implemented fairly well.
- The relationship between variables can be easily visualized by applying this model.

Limitations

- Over-fitting is quite common with this model and this leads to it becoming a not a good model for real-world applications. My implementation of this model exhibited overfitting.
- Outliers can have an adverse effect, as the boundaries and values conform to a linear equation.

3.2 Polynomial Regression

The polynomial regression technique involves creating a set of polynomial features from the given data before applying linear regression to it. The curve generated for the model is a polynomial curve, computed mathematically from a polynomial expression.

Advantages

- It is easily comprehendible if one has a sufficient understanding of linear regression models and polynomial curves.
- It's a good approach for non-linear problems.

Limitations

- Unless the right polynomial degree is chosen, the variance calculated as part of the model will not yield the right predicted value.

3.3 Decision Tree Regression

A decision tree in general helps in choosing an outcome from each set of available options. It is in the form of a tree structure and it is one of the most common approaches for supervised learning. Hence it is used in regression, to predict a value as it traversed through the decomposed list of outcomes segregation.

Advantages

- It is easier to interpret as the concept of a decision tree allows one to predict outcomes

from available options.
• It works well with both linear and non-linear problems, thereby making it better than linear and polynomial regression.
Limitations
• Over-fitting is quite common with this model as well.
• It is not suitable for small datasets.

3.4 Random Forest Regression

A Random Forest is a combination of multiple decision trees obtained from the given dataset. The value is then computed as an average of the values from these randomly picked decision trees and a final result is presented from this average. Due to the randomness of this solution, it was not chosen for the given problem, as a more mathematical approach was needed to predict the rent for this problem.
Advantages
• It is a very powerful algorithm as it takes the value of multiple decision trees making it much more efficient as well.
• The accuracy of values predicted using a random forest machine learning model has proved to be accurate in most applications.
Limitations
• As in Decision Tree Regression, overfitting occurs as this is a combination of multiple decision trees.
• As the decisions are randomly chosen from a bunch of decision trees, interpretability is less.

3.5 Neural Network Regression

Neural networks emulate the way the neurons in the brain communicate with the nervous system in the human body. This property makes it a good choice for problems that deal with prediction. The neural network regressor predicts continuous numeric values. The data is organized in layers with the predicted values at the bottom-most layer and the forecasters as the topmost layer.
Advantages
• Predictions are quite faster with neural network models and any numeric data can be easily used for regression with this model.
• It is a good model to work with for a large number of inputs. The current problem to predict the rental unit's price has a lot of inputs for prediction. That makes this a right fit for this problem.
• Any number of layers and inputs can be trained with neural networks.
Limitations
• This model requires a lot of training data for it to work efficiently.
• This model is computationally expensive and time consuming for training.

3.6 Selected Solution Justification

I have implemented all the above models and observed their results with accuracies as I have mentioned. I have chosen the neural networks model as the best fit for the given problem. This is because the data does not get overfitted with the given model and it gives a more realistic approach to predict the data. The other models explained above, also give good accuracy and expected prediction values for the given problem of predicting the rental price of houses. Neural Networks is also one of the emerging fields in machine learning. Hence, I am taking this as the chosen method to implement this rent prediction problem although the above models have better accuracy. There are various ways to implement neural networks such as using Keras with Tensorflow, Pytorch, etc. I have chosen to use the neural networks regressor available with sklearn as it is much easier to comprehend and apply. Also, the other models that I have implemented are from sklearn and I would like to learn and observe the difference in the working of various regression models for prediction provided by sklearn. The way neural networks work is explained in detail in step 3.

4 Step Three: Selected Solution

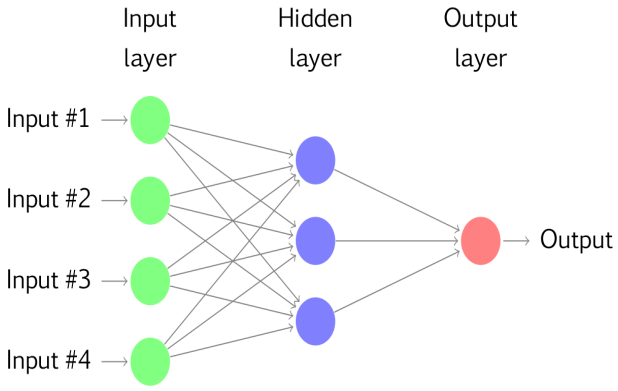
Prediction for numerical data can be done using regression. I have listed above a few regression techniques that I have applied to my given problem of predicting the price of a given rental

unit every month. Out of these, I will be choosing the Neural Networks Regression model for the above-mentioned reasons. This section gives a detailed explanation of how Neural Network models work concerning regression.

4.1 Neural Networks Model

Neural Networks emulate the way the human brain behaves while making decisions. It uses a mathematical model to correlate these decisions in the form of multiple layers and give an outcome or prediction. The input consists of the various factors that help in prediction, predictors. This is the bottom-most layer. The output is the top-most layer which represents the outcome, or the forecasts predicted from the various values of the predictors. Each node in a neural network is called a neuron.

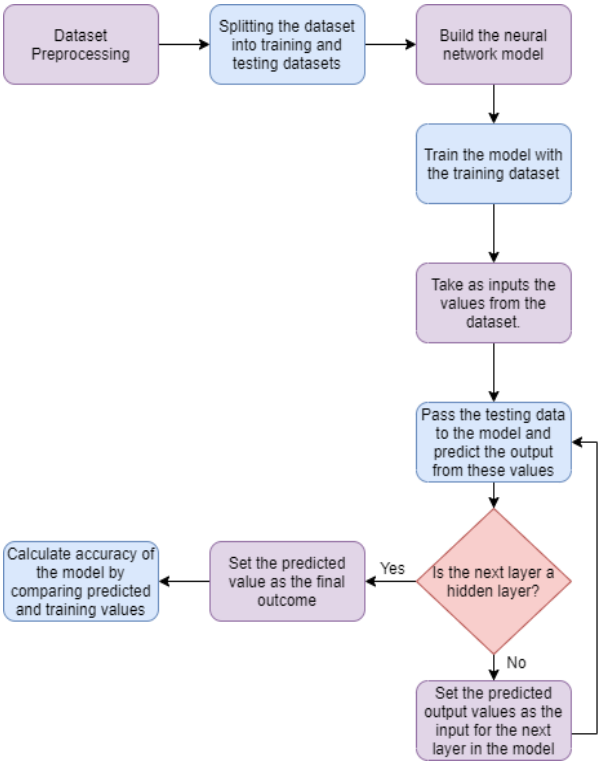
Most Neural networks have similar behavior to linear regression models with just the input layer and the output layer. Each of the inputs has a coefficient attached to it. This is called its weight. Initially, the weights take random values but proceed to take an observed value as the model traverses deeper into the neural network. If there are more layers other than the input and output layer, they are called the hidden intermediate layer. This layer consists of hidden neurons. Each neuron is a predictor for the next layer and a forecaster from the previous layer. Once these layers are added the model becomes non-linear. This architecture is termed a multilayer feed-forward network. It can be illustrated as shown below.



These models are deemed a good fit for prediction analysis as these intermediate hidden layers not only make the model non-linear but make it better at accurately predicting data. These hidden layers learn the way a human being does, based on the decisions made right or wrong. This is why, when considering datasets like rent prediction though several factors come into play, the foremost important thing a business would need to know without the model is how humans get what they want in a house at a price they are ready to pay. The advantage of neural networks is that it has a good design that makes it able to foreplay this behavior using its hidden layers, making it a good choice for the house rent prediction problem being discussed in this paper.

4.2 Flowchart

The following diagram shows how data is fed and processed through a neural network regression model in the sklearn library.



4.3 Available Implementations

Some of the ways to implement neural networks include using the Keras library of Ten-

sorFlow, PyTorch, and the Sklearn library provided by python. I tried implementing my current dataset with the Keras library and found that it takes a lot more time to train than the sklearn library. I have enclosed the code for this trial as a separate notebook in the code folder. Since the other models were implemented using sklearn, I finalized implementing the neural networks model with sklearn as well as it.

4.4 Applying the Dataset to the model

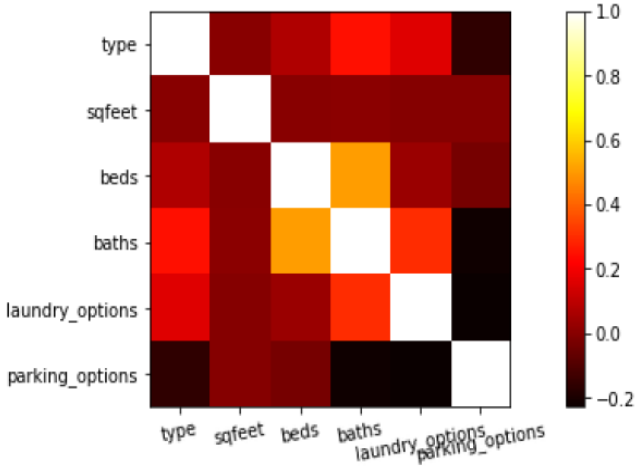
On running the model on the dataset of the given problem, it predicted values within the range expected as the price for the rental unit. The accuracy of the model was above 90 percent.

5 Step Four: Model Evaluation for Implemented Solution

This step outlines the evaluation of each of the models and an explanation of why neural networks are a better model. For any of the regression models, the steps to work with the model is the same. Split the dataset into training and testing data. Train the model, make the model learn about the data, with the training dataset. Then predict the required value by sending the testing dataset as the input. Then compute the score of that model by comparing the predicted values with the original value of the dataset.

5.1 Factors influencing Rental Unit Price Prediction

The different kinds of factors that influence the decision of a person to rent a unit include parking options, laundry options, the number of bedrooms and bathrooms, the area of the unit, the type of unit, pet tolerance, smoking tolerance, and few others. The dataset holds these data for various rental units available in different places in the US. To work further and apply these factors as predictors or inputs to a neural network model, we need to understand how these factors correlate with one another. This helps to see how they collectively influence the price of a rental unit. A heatmap denoting the correlation matrix of some of these inputs is visualized below.



We can see the level of correlation between multiple factors. For example, the correlation between the number of bedrooms and the number of bathrooms is quite high. This makes it one of the most primary choices while considering renting a unit.

5.2 Metric Evaluation

For all the available models listed above for this rental unit price prediction problem, I have chosen to compute the accuracy of the model using the Score method of the regression models available as part of the sklearn library. Each method, whether it be a neural network regressor model or a decision tree regressor model, has its own score method. This gives the accuracy of that model by comparing the predicted values with the original values in the training dataset and computes a score between 0 and 1. This accuracy can further be converted into a percentage by multiplying it by 100. I chose this metric as it has a better accuracy value as it's specific to each of the models being trained rather than from a separate library's class by itself.

5.3 Neural Network Score

The score of the neural network model was fine-tuned by setting the learning rate of the model to adaptive and avoiding a shuffle in the data to reduce randomness. The neural network regressor is called the MLPRegressor

in the sklearn library. It gave an accuracy of 99 percentage for the currently tested dataset for the house rent prediction problem.

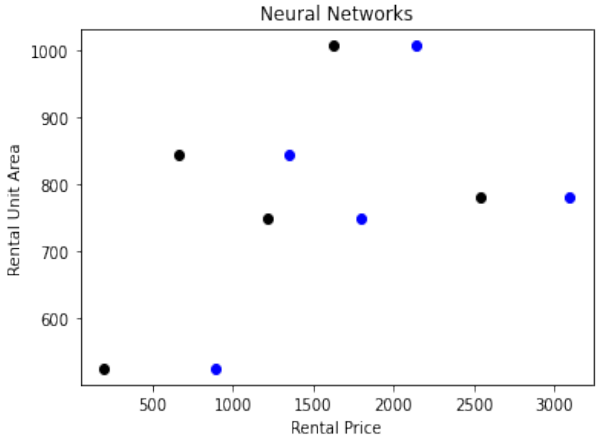
The Linear Regression model had an accuracy of 100 percent which could be attributed due to overfitting. Other models also face this overfitting problem. For instance, the polynomial regressor has an accuracy of 99 percent. The accuracy of the decision tree regressor was 92 percent. The random forest regressor had an accuracy of 96 percent, but this accuracy changes with every run owing to the randomness of the model. Also, a random forest chooses a result from multiple decision trees thereby attributing to this value as the score.

I still chose the neural network model as the best solution for this problem not mainly because of the score but because of the way the model works. The design of the model and the way it predicts the outcome can be justified as the right approach as it is similar to the way human beings tend to make decisions. The accuracy is good enough, without being affected by overfitting issues as well and hence it seemed the appropriate approach for prediction.

For this project, not solely the accuracy was taken into consideration as the best factor to determine the right model for the house rent prediction. Despite having good accuracy, the neural networks model holds well for large datasets. Real-world problems, such as this, require it to be efficient for constantly growing datasets. This and the design of the model, as mentioned above, led me to conclude that this would be the niche model for this experiment.

5.4 Visualized Result

The predicted data can be analyzed in the given graph as follows. For the Neural Network model, the predicted values have a weight attached to them. To look at the original value, I have assumed a weight and subtracted the value from the predicted value to show the predicted rent and plot it.



The Predicted value concerning the trained data is predicted in a scatter plot as shown below. The scatter plot below shows the price for one of the factors, the area of the rental unit. The blue dots indicate the estimated values and the black dots indicate the predicted values. Similarly, each of the factors can be compared with the rental price to see how one influences the other.

6 Step Five: Limitations and Future Work

This section involves analyzing the major limitations of using the neural networks regressor model and a justification as to why it is the right fit despite that. A footnote is also included on what more can be done with this model to improve or extend this project idea.

6.1 Drawbacks

As mentioned above the major drawback of using neural networks is that it takes a large amount of time to train, though the prediction time is less. Also, the processing speed required to run a model with Big Data with neural networks is large and this makes it computationally expensive to work with. Despite these drawbacks, neural networks are the right fit for this problem as it emulates human decisions and gives a good accuracy despite the high cost that it takes to build the model. As mentioned

before, this is the reason why this model can be particularly useful for this problem.

6.2 Future Prospects

Future work includes exploring the neural network regressors offered by libraries other than sklearn and implementing the same idea to see the difference among them. Also, neural networks are currently being used in many applications for image recognition and facial recognition. The look of the house is one of the other foremost factors that decide whether a person would rent it. Including a separate model that allows processing the pictures of the house to assess its quality and predict whether the user likes the rental unit based on the images will be a better extension to this project.

7 Conclusion

Prediction is one of the major machine learning problems that exist. By choosing this as the idea for the project and choosing a dataset that has a niche to become a potential business, the learning curve increased for me in this project. As a result of this paper and survey of available options, I can see from the results that the neural network model is a stable, reliable, and accurate model that can be used for the prediction of the data from the chosen dataset.

References

[1] Kaggle.com. 2020. House-Rent-Prediction-Dataset. [online] Available at: ¡https://www.kaggle.com/rkb0023/ houserentpredictiondataset¿ [Accessed 1 December 2020].

[2] Scikit-learn.org. 2020. Sklearn.Preprocessing.Labelencoder — Scikit-Learn 0.23.2 Documentation. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html¿ [Accessed 3 December 2020].

[3] Scikit-learn.org. 2020. Sklearn.Linear_Model.Linearregression — Scikit-Learn 0.23.2 Documentation. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html¿ [Accessed 5 December 2020].

[4] I. Implementation) and A. Sharma, "Polynomial Regression — Polynomial Regression In Python", Analytics Vidhya, 2020. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/03/polynomial-regression-python/. [Accessed: 8- Dec- 2020].

[5] "sklearn.tree.DecisionTreeRegressor — scikit-learn 0.23.2 documentation", Scikit-learn.org, 2020. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html. [Accessed: 10- Dec- 2020].

[6] "3.2.4.3.2. sklearn.ensemble.RandomForestRegressor — scikit-learn 0.23.2 documentation", Scikit-learn.org, 2020. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html. [Accessed: 11- Dec- 2020].

[7] Scikit-learn.org. 2020. Sklearn.Neural_Network.Mlpregressor — Scikit-Learn 0.23.2 Documentation. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html¿ [Accessed 13 December 2020].

[8] Scikit-learn.org. 2020. Sklearn.Metrics.Silhouette_Score — Scikit-Learn 0.23.2 Documentation. [online] Available at: ¡https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html¿ [Accessed 14 December 2020].

[9] GeeksforGeeks. 2020. Advantages And Disadvantages Of Different Regression Models - Geeksforgeeks. [online] Available at: ¡https://www.geeksforgeeks.org/advantages-and-disadvantages-of-different-regression-models/¿ [Accessed 14 December 2020].

[10] Otexts.com. 2020. 11.3 Neural Network Models — Forecasting: Principles And Practice. [online] Available at: ¡https://otexts.com/fpp2/nnetar.html¿ [Accessed 15 December 2020].

[11] "Build your first Neural Network to predict house prices with Keras — Hacker Noon", Hackernoon.com, 2020. [Online]. Available: https://hackernoon.com/build-your-first-neural-network-to-predict-house-prices-with-keras-3fb0839680f4. [Accessed: 15- Dec- 2020].

[12] Subscription.packtpub.com. 2020. Neural Networks Pros And Cons. [online] Available at: ¡https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781788397872/1/ch01lvl1sec27/pros-and- cons-of-neural-networks¿ [Accessed 15 December 2020].

[13] Express Analytics. 2020. Prediction Using Neural Networks - Express Analytics. [online] Available at: ¡https://expressanalytics.com/blog/prediction-using-neural-networks/¿ [Accessed 15 December 2020].

[14] "Flowchart Maker Online Diagram Software", App.diagrams.net, 2020. [Online]. Available: https://app.diagrams.net/. [Accessed: 17- Dec- 2020].