AGRICULTURAL PRODUCE DEFICIT PREDICTION

A Mini-Project Report

Submitted by

Deepshika Raghuraman	2019506026
Pradeep P	2019506062
Shankar N	2019506084

Under the supervision of

Dr. D Sangeetha & Dr. S Uma Maheswari

In partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

MADRAS INSTITUTE OF TECHNOLOGY CAMPUS

ANNA UNIVERSITY, CHENNAI – 600044

JANUARY 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this mini-project report titled **"AGRICULTURAL PRODUCE DEFICIT PREDICTION"** is the bonafide work of Deepshika Raghuraman (2019506026), Pradeep P (2019506062) and Shankar N (2019506084) who carried out the project work under my supervision.

Signature

Dr. Dhananjay Kumar

HEAD OF THE DEPARTMENT

Professor

Department of Information Technology

MIT Campus, Anna University

Chennai – 600044

Signature

Dr. D Sangeetha

SUPERVISOR

Professor

Department of Information Technology

MIT Campus, Anna University

Chennai – 600044

ACKNOWLEDGEMENT

It is essential to mention the names of the people, whose guidance and encouragement made us accomplish this project.

We express our thankfulness to our project supervisor and panel members **Dr. D Sangeetha,** and **Dr. S Uma Maheswari**, Department of Information Technology, MIT Campus, for providing invaluable support and assistance with encouragement which aided to complete this project.

Our sincere thanks to **Dr. Dhananjay Kumar**, Head of the Department of Information Technology, MIT Campus for catering all our needs giving out limitless support throughout the project phase.

We express our gratitude and sincere thanks to our respected Dean of MIT Campus, **Dr. T. Thyagarajan**, for providing excellent computing facilities throughout the project.

DEEPSHIKA RAGURAMAN 2019506026

PRADEEP P 2016506062

SHANKAR N 2016506084

ABSTRACT

Prediction of shortage of agricultural commodities is an everlasting and recurring problem that has gained the attention of the state and the stakeholders of the agricultural sector. Every other year, our country faces a fluctuation in the amount of agricultural produce that reaches the vegetable markets which also drastically increases the price of these basic commodities. The scenario is not ideal for the country as it poses a threat to the economy as well as the food security of the citizens. To help avoid such a situation, or to be well prepared, we propose this project which aims to develop a model that helps to predict the possibility of such a shortage in advance. The model makes use of the techniques offered by machine learning and applies them on the data available regarding contributing factors like climatic conditions, price and crop availability, to estimate the possibility of a deficit. To ensure the credibility of the result, the data is analysed and heuristics is performed on it to isolate the relevant features which are then used in the model. Several Machine Learning models like ANN, Decision Trees, Random Forest, SVM and XGBoost were evaluated to find the most optimum one. Further, to provide the stakeholders with a means by which they can access this information, the project is deployed in real time as an interactive website. The complete research and experiments showed that the Random Forest model and XGBoost gave the most accurate results out of the models that were employed.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

In 2019, due to an unexpected turn of events, the vegetable markets of India saw a sudden scarcity of onions. Consumers, who had till then taken the commodity for granted, found themselves shelling out more than twice the usual rate. In fact, the situation became serious to the point where The Government had to temporarily suspend the export of onions.

This incident has occurred multiple times over the past decade and is inevitable. Thus, it would be of help to have a means by which to predict such a scarcity well in advance for the benefit of the nation.

1.2 RESEARCH CHALLENGES

Several projects centred around crop price prediction and crop yield prediction have been carried out as they are equally pressing issues. Due to their popularity, there are several resources available for furthering research and study in these domains. Unfortunately, crop shortage prediction has not gained an equal amount of attention until recent years due to which there are no satisfactory resources available that would help make any significant advances. Hence, accumulating accurate and reliable datasets to develop a model to predict the possibility of a shortage poses a great challenge. Additionally, due the lack of accurate data, there are several difficulties in developing a model of high reliability and success rate.

1.3 OBJECTIVE

The major objective is to develop a model that can successfully predict the possibility of there being a shortage of any given agricultural commodity well in advance. The Government and the agricultural sector should be provided with the means to ensure food security in the nation.

1.4 SCOPE OF THE PROJECT

The proposed system design aims to make use of several available Machine Learning algorithms like ANN, Decision Trees, Random Forest, SVM and XGBoost to determine which one provides the best results for shortage prediction. The model also makes use of heuristics to filter out irrelevant data and choose the factors that have the most impact on the outcome. Furthermore, visualization techniques are used to graphically represent the data. The model makes use of the weather API to get real time information regarding the weather forecast of the selected center.

1.5 CONTRIBUTION

The project contributes to developing a model that helps make a prediction regarding whether a shortage in an agricultural product will occur. The general approach involves making use of the machine learning algorithm that provides the best result while integrating heuristics to this would increase the accuracy of the prediction. The model was deployed as a website and data was fetched in real time to make a prediction.

1.6 ARTIFICIAL NEURAL NETWORKS (ANNs)

The ANN algorithm is based on a collection of connected units or nodes called artificial neurons that model the neurons in a brain. A neuron receives a signal then processes it and can signal neurons connected to it. The signal is a real number and the output of each neuron is computed by some non-linear function of the sum of its inputs. Neurons and edges typically have a weight that adjusts as learning proceeds. Neurons may have a threshold such that a signal is sent only if the aggregate signal crosses that threshold. Different layers may perform different transformations on their inputs. Signals travel from the first input layer, to the last output layer after traversing the layers multiple times.

1.7 DECISION TREES

A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences. It is a way to display an algorithm that only contains conditional control statements. It is a flowchart-like structure in which each internal node represents a test on an attribute. The paths from root to leaf represent classification rules.

1.8 SUPPORT VECTOR MACHINE (SVM)

Support Vector Machine is one of the most popular supervised learning algorithms. It is used for classification as well as regression problems. It's aim is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.

1.9 RANDOM FOREST

A Random Forest is a machine learning technique that's used to solve regression and classification problems. It utilizes a technique that combines many classifiers to provide solutions to complex problems. It consists of many decision trees. It establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome.

1.10 XGBOOST

XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. It is an implementation of gradient boosted decision trees designed for speed and performance. The implementation of the algorithm was engineered for efficiency of compute time and memory resources. It is a perfect combination of software and hardware optimization techniques to yield superior results using less computing resources in the shortest amount of time.

1.11 FEATURE ENGINEERING

It is the process of adding, modifying and removing data into your dataset so it is optimised for machine learning. It Mainly revolves around feature selection, elimination and extraction. Feature selection and elimination is the process of picking which features you want to keep and use to train the model, excluding the ones that supposedly affect the outcome the least.Feature Extraction is adding in new features based on domain knowledge and knowledge of the dataset.

1.122 HEURISTICS

Rules-based model a.k.a. heuristics-based modelling reflects the current business processes or based on a set of rules. The hybrid approach of heuristics plus machine learning is very powerful and is essentially an extension on feature engineering. Heuristics can be incorporated into a machine learning model by preprocessing using the heuristics, creating a feature directly based off of the heuristic, mining the raw inputs of the heuristic or modifying the target variable. For this project, the Moving Average(MA) model is the most apt.

1.13 DATA VISUALIZATION

Data visualization is the process of representing information and data sets in the form of a graphical image such as a chart, diagram or picture. It is required to deal with the output of high-volume data sources. Data sets are classified according to their spatial distribution and according to data type. Two-dimensional data sets have values distributed over a surface, and three-dimensional data sets have values distributed over some region of space. Data types include scalars, vectors, tensors, and multivariate data.

1.14 WEATHER API

A weather API is an Application Programming Interface that allows weather data to be queried from scripts and code.

CHAPTER 2
LITERATURE SURVEY

2.1 CROP YIELD PREDICTION USING MACHINE LEARNING: A SYSTEMATIC LITERATURE REVIEW

In this paper, They have investigated studies and analyzed the methods and features used to predict crop yield and provided suggestions for further research through SLR. According to their analysis, the most used features are temperature, rainfall, and soil type, and the most applied algorithm is Artificial Neural Networks in these models.

2.2 CROP PRICE PREDICTION SYSTEM USING MACHINE LEARNING ALGORITHMS

This paper predicts the price of crops to help farmers. It uses linear regression and neural networks to predict the price. It also analyzes the yield vs. production and export vs. import of soybean between China and The USA. Data related to this trade was used to predict market price. Several algorithms were used to help predict the price of which XGBoost gave the most accurate results.

CHAPTER 3
SYSTEM ARCHITECTURE AND DESIGN

3.1 SYSTEM ARCHITECTURE

The module that helps make the prediction of crop shortage is constructed in Python using the Machine Learning models and correlated functions that are in-built. The working deliverable, which is the website, deploys the above module that has been trained to accurately predict the possibility of a shortage in a crop occurring using Flask in Python. The website has input fields which allows the user to interact with the site and provide data regarding the desired center/market, the crop's price in that particular market, and the quantity of the produce that has reached the market in a particular month. Apart from the data that is provided by the user, the backend Python program fetches real-time data of the weather in the selected center using a Weather API. This API provides the program with information regarding the maximum and minimum temperatures, humidity, cloud cover, and rainfall. The data that is submitted by the user and the data which is fetched using the API are wrapped together and is further processed. This wrapped data is first encoded to contain only numerical values after which it undergoes feature scaling. Finally, the data is submitted to the model to receive a prediction. The architecture diagram of the frontend of the proposed system is shown in Fig. 3.1. It depicts the various factors that are required to make a prediction.

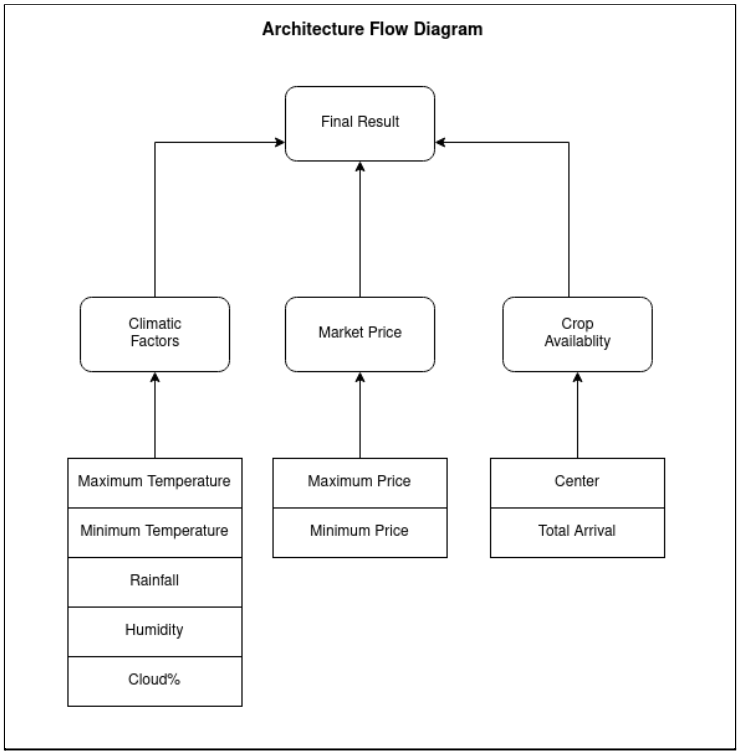

Figure 3.1 – System Architecture

3.2 FEATURE ENGINEERING

Feature engineering involves leveraging data mining techniques to extract features from raw data along with the use of domain knowledge. It involves several processes such as Feature Encoding, Feature Construction, Feature Scaling, and Feature Selection. After extracting row data from a resource, the dataset undergoes a sequential process for selecting the best features to get utmost accuracy during model creation. The categorical features are first encoded into a binary format and extra features are added if necessary. This process is called feature construction. The dataset may contain various ranges of values which are essential for the model but these ranges may take a lot of computational power and time. Thus the encoded dataset undergoes feature scaling which reduces the value range drastically and brings it to a normalized form. Feature scaling also helps in better comparison between features during model training. Finally the normalized dataset is tested in various aspects to

select the best features to train the model. The various steps involved in feature engineering are shown in Fig. 3.2.

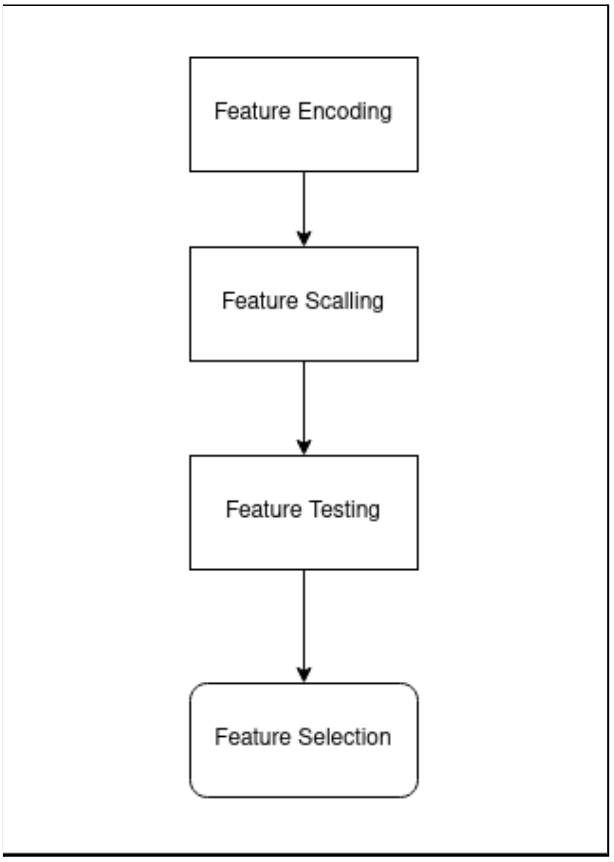

Figure 3.2 – Feature Engineering

3.3 MODEL DEVELOPMENT

The model to predict the occurrence of a shortage is built using the Machine Learning algorithms which are present in the Python packages. The input parameters of the model are decided by performing feature engineering on the dataset. The fields that have the most impact on the occurrence of a shortage are taken to be the input factors. After performing feature engineering on the used dataset, it was found that the factors to be considered are arrival of the crop, maximum temperature, minimum price, humidity, rainfall, minimum temperature, maximum price, cloud and center in decreasing order of impact on

the final result. Here, Fig. 3.3. depicts the overall process of constructing a machine learning model in the form of a data flow diagram.

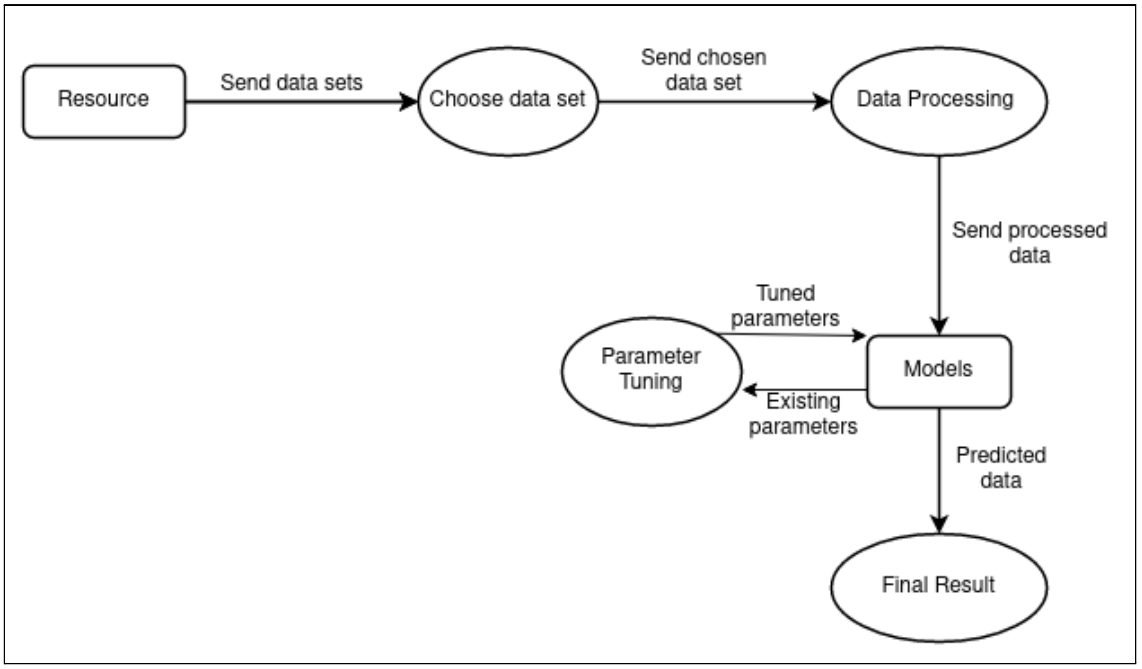

Figure 3.3 – Data Flow Diagram

The above features that have already been encoded and scaled are given as input to train the models. For the project, five different Machine Learning models were considered and experimented with to find the model that gave the most accurate prediction. The models that were implemented are Artificial Neural Networks, Decision Tree, Random Forest, SVM and XGBoost. Each of these models were trained with the given input dataset and parameter tuning was performed on them to determine the most accurate model. The output expected from the trained model is 'yield' where a value of 1 suggests that a shortage would occur while a value of 0 implies that there would be no shortage. The detailed process of developing a machine learning model is shown in Fig. 3.4 in the form of a swimlane diagram.

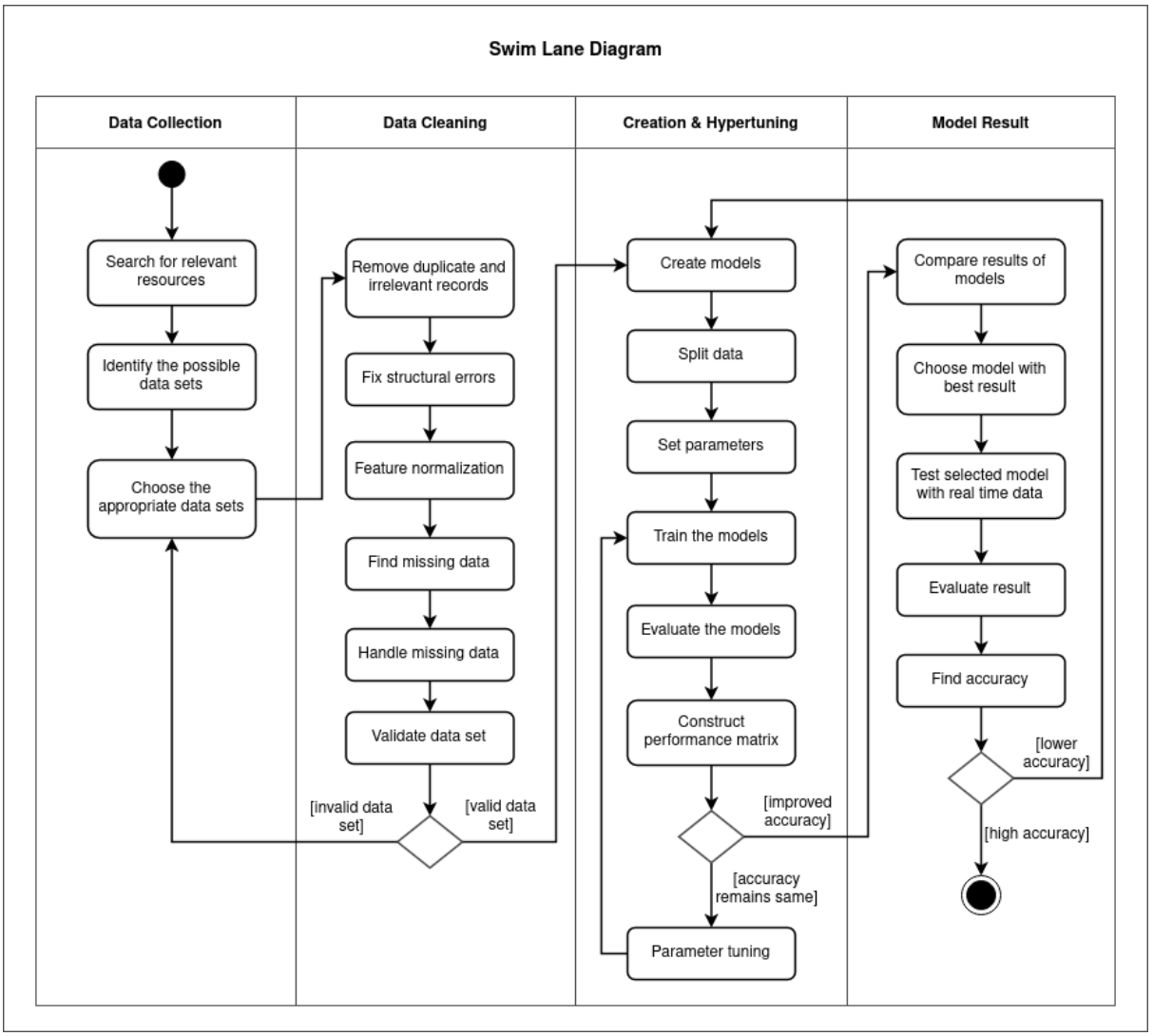

Figure 3.4 – Swim Line Diagram

CHAPTER 4
ALGORITHM DEVELOPMENT AND IMPLEMENTATION

4.1 MACHINE LEARNING BASED APPROACH

Machine Learning based approach is used in developing the model to make accurate predictions. Since the implementation requires a yes/no type of prediction we use semi-supervised classification learning models. The models used are Artificial Neural Networks, Decision Tree, Random Forest, SVM and XGBoost. Further, feature engineering methods are implemented on the dataset before constructing the model.

4.2 ALGORITHM - MODEL CONSTRUCTION AND SELECTION

1. Start
2. Read the raw data from the resource
3. Inspect the raw data for presence of invalid and null values
4. Clean the data to remove identified invalid and null values
5. Encode the data to numerical format
6. Create extra features
7. Normalize the values to a standardised form
8. Test the dataset to find the relationship between features and the final result
9. Identify features that have the most impact on the result
10. Split the dataset into training and testing sets
11. Normalize the training and testing data
12. Create and train the model with the training dataset
13. Perform parameter tuning on the model
14. Test the model by predicting with the testing dataset
15. Compute the classification and confusion matrix with predicted data
16. Evaluate the model
17. Identify the model with the best accuracy
18. Deploy the model by integrating within website
19. End

4.3 ALGORITHM - WEBSITE

1. Start
2. Receive input from the user for center, month, price, and arrival
3. Submit the data to the machine learning model which is in backend
4. Fetch data regarding the weather using the Weather API
5. Construct the input dataset
6. Encode the dataset to numerical format
7. Normalize the dataset into standardized form
8. Feed the normalized dataset to the model to get prediction
9. Print the prediction result to the website
10. Display the weather of the chosen center in the site
11. Generate line graph that visualizes the maximum price per center per year
12. Generate the bar graph that visualizes the total arrival per center per month for any given year
13. End

4.4 IMPLEMENTATION ENVIRONMENT

Python programming environment is used for implementation purposes since it provides a rich library support. The project was initially developed on Jupyter Notebook due to its user-friendly interface. Furthermore, the project makes use of HTML, CSS, and JavaScript to construct the website. In later stages, Spyder IDE was used to deploy the project due to its simplicity and ease-of-use. The machine learning model was integrated to the website using Flask. The final website was deployed on the localhost using Anaconda.

CHAPTER 5
RESULTS AND DISCUSSIONS

5.1 IMPLEMENTATION ENVIRONMENT

A website that helps predict the possibility of a crop shortage occurring in a particular center/market has been created. The user provides details regarding the center and the price and arrival of a crop to the website which is submitted to the machine learning model that is integrated to the site. The model then fetches the remaining details regarding the weather using the Weather API and feeds the collective encoded and scaled data to the prediction model. The model returns the prediction result back to the site where it is displayed along with the weather of the center. Further, visualization of maximum price of crop per center per year as a line graph and a bar graph that depicts the total arrival of crop per center per month for a given year are also displayed.

The proposed system is implemented and tested using the Python programming language. The machine learning models are constructed using the packages provided by Python. The website is created using HTML, CSS and JavaScript. The machine learning model is integrated in the website using Flask.

5.2 FEATURE SELECTION

The features to be taken as the input to the model based on the impact that they have on the final result are determined using F-Test. An F-test is any statistical test in which the test statistic has an F-distribution under the null hypothesis. It is most often used when comparing statistical models that have been fitted to a data set, in order to identify the model that best fits the

population from which the data were sampled. Fig. 5.1 shows the features that were determined to have the most impact on the final result after performing the F-Test.

```
fs.fit(xLabelEncN,Y)
fsScore=pd.DataFrame(fs.scores_,columns=['Score'])
fsCol=pd.DataFrame(xLabelEncN.columns,columns=['Category'])
fsDf3=pd.concat([fsCol,fsScore],axis=1)
fsDf3.nlargest(10,columns='Score')
```

	Category	Score
4	Year	22.150577
3	Total Arrival	18.876964
7	Max Temp °C	10.732223
1	Min Price	9.815931
10	Humidity(%)	8.300431
9	RainFall(mm)	6.638772
6	Min Temp °C	5.358360
2	Max Price	2.971594
11	Cloud(%)	2.593311
0	Center	0.918425

Figure 5.1 – Feature Selection using F-Test

5.3 MODEL COMPARISON

The project considered five machine learning algorithms to construct the model on the selected features. ANN, Decision Trees, Random Forest, SVM and XGBoost models were evaluated to find the most optimum one. The ANN model gave an accuracy of 47.89%, the SVM model gave an accuracy of 56.33%, the Decision Tree model gave an accuracy of 75.94%, the Random Forest model gave an accuracy of 83.01% and the XGBoost model gave an accuracy of 84.90%. On comparing these five models and their

accuracies it was found that the XGBoost model had the highest accuracy and hence was chosen to deploy in the final project.

5.3.1 Decision Tree

```
model.score(xTScale, yT)

0.7594339622641509

pred=model.predict(xTScale)
print(metrics.classification_report(yT,y_pred=pred))

              precision    recall  f1-score   support

           0       0.76      0.78      0.77       109
           1       0.76      0.74      0.75       103

    accuracy                           0.76       212
   macro avg       0.76      0.76      0.76       212
weighted avg       0.76      0.76      0.76       212
```

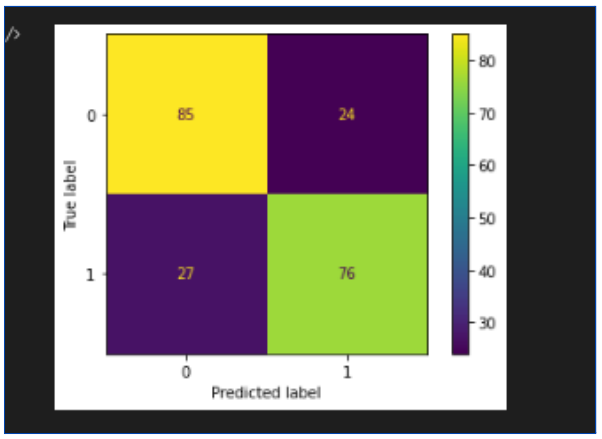

Figure 5.2 – Decision Tree Result

5.3.2 Random Forest

```
Accuracy:  0.8301886792452831
Precision:  0.8073394495412844
Recall:  0.8543689320388349
```

```
              precision    recall  f1-score   support

           0       0.85      0.81      0.83       109
           1       0.81      0.85      0.83       103

    accuracy                           0.83       212
   macro avg       0.83      0.83      0.83       212
weighted avg       0.83      0.83      0.83       212
```

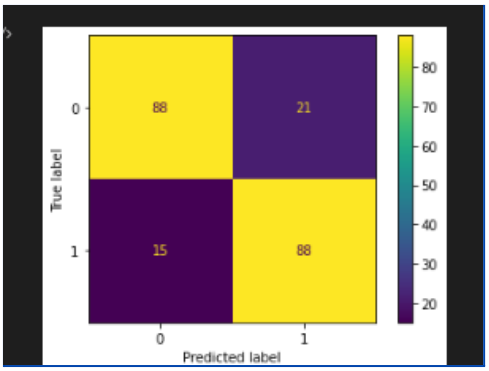

Figure 5.3 – Random Forest Result

5.3.3 XGBoost

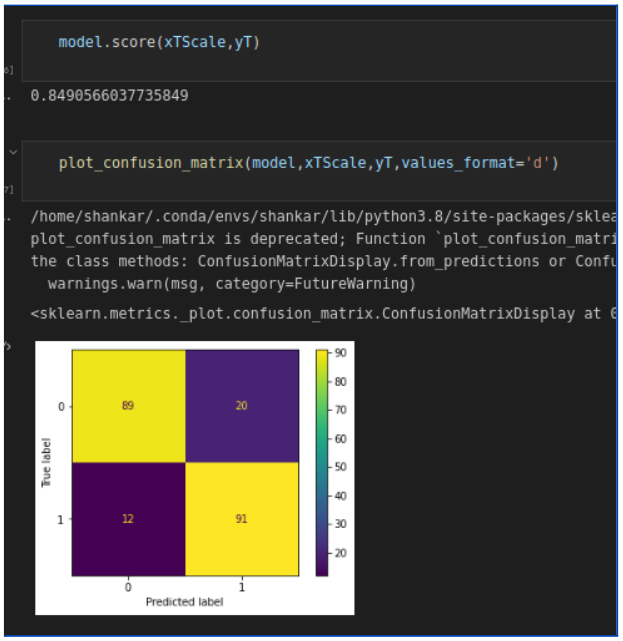

Figure 5.4 – XGBoost Result

5.3.4 SVM

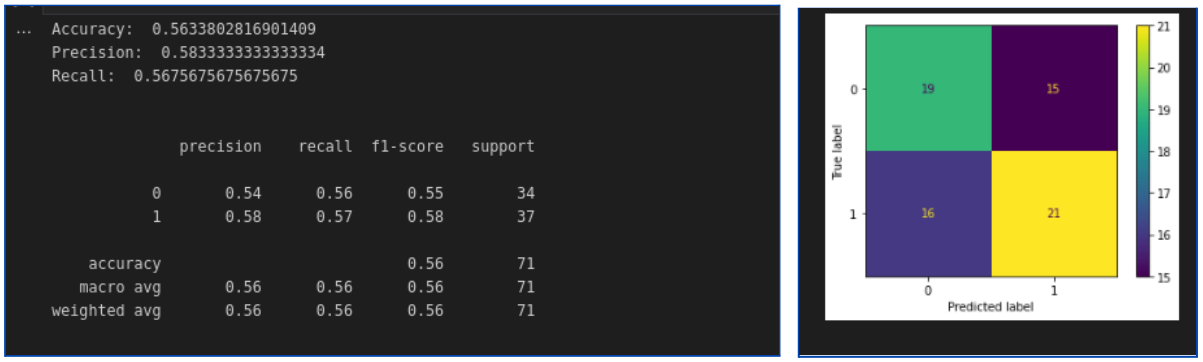

Figure 5.5 – SVM Result

5.3.5 ANN

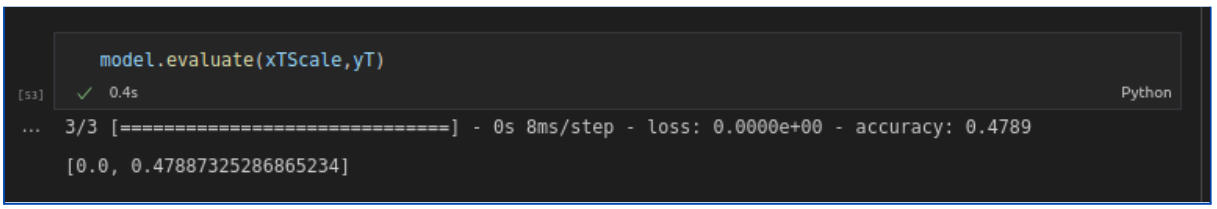

Figure 5.6 – ANN Result

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

A website that helps to predict the possibility of there occurring a shortage of a particular crop in a given center has been created and deployed using machine learning. The various challenges faced during the development of the model are analysed. The ability to be able to predict the possibility of a crop shortage is required for The Government and the agricultural sector of the nation to ensure food security. As the XGBoost model is used here along with feature engineering, the accuracy of the model is improved greatly. The proposed system generates the required prediction accurately using the input data provided.

6.2 FUTURE WORK

The proposed system was developed for a minimum number of centers. This can further be extended to incorporate all other centers in the country as well as a range of agricultural products. The system can also be deployed as an official website that is recognised by The Government. Furthermore, the accuracy of the model can be greatly improved by sourcing and using official and credible datasets. Doing so would also allow there to be more modules or categories based on which the prediction could be made.

REFERENCES

1. Crop Yield Prediction using Machine Learning: A Systematic Literature Review

2. Crop Price Prediction System using Machine Learning Algorithms 1

3. World Weather Online: 14 Day Weather Forecast

4. National Horticulture Board

5. Web Application Using Flask

6. Render HTML Using Flask

7. Build Web App Using Flask

8. Chart.js

9. OpenWeatherAPI