VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590018, Karnataka

A **Project Phase-II** report on

"CROP YIELD PREDICTION USING DEEP LEARNING"

Submitted in fulfillment for the requirements of VIII semester degree of

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

By

Kundan Kumar Prasad (1DB17IS018)

Shantideepa Samanta (1DB17IS035)

Under the Guidance of

Mrs. Mamatha K.

Assistant Professor,

Dept of Information Science and Engineering

Don Bosco Institute of Technology

Department of Information Science and Engineering

DON BOSCO INSTITUE OF TECHNOLOGY

Kumbalagodu, Mysore road, Bangalore-560074

2020-2021

Don Bosco Institute of Technology

Kumbalgodu Mysore Road

Bangalore-560074

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

Certified that the Project on topic **" CROP YIELD PREDICTION USING DEEP LEARNING"** has been successfully presented at **Don Bosco Institute of Technology** by **KUNDAN KUMAR PRASAD (1DB17IS018), and SHANTIDEEPA SAMANTA (1DB17IS035),** in partial fulfillment of the requirements for the VIII Semester degree of **Bachelor of Engineering in Information Science and Engineering** of Visveshvaraya Technological University, Belagavi during academic year 2020-2021. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The Project report has been approved as it satisfies the academic requirements in respect of Project work for the said degree.

Signature of Guide	Signature of HOD	Signature of Principal
Mrs. Mamatha K	Mrs. Gowramma GS	Dr. Hemadri Naidu
Assistant Professor	Associate Professor, HOD	Principal, DBIT
Dept. of ISE, DBIT	Dept. of ISE, DBIT	

EXTERNAL VIVA

Name of the External	**Signature with date**
1. _____	_____
2. _____	_____

ACKNOWLEDGMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidanceand encouragement crowned the efforts with success.

I would like to profoundly thank **Management** of **Don Bosco Institute of Technology** for providing such a healthy environment for the successful completion of Project work.

I would like to express my thanks to the Principal **Dr. Hemadri Naidu** for their encouragement that motivated me for the successful completion of Project work.

It gives me immense pleasure to thank **Professor Gowramma G S**, Head of Department for his constant support and encouragement.

Also, I would like to express my deepest sense of gratitude to **Mrs. Mamatha K**, Assistant Professor, and Department of Information Science & Engineering for the constant support and guidance throughout the Project work.

I would also like to thank all other teaching and non-teaching staff of Information Science Department who has directly or indirectly helped me in the completion of the Project work.

ABSTRACT

Agriculture provides a living for around 58 percent of India's population. Agriculture, forestry, and fisheries were expected to generate ₹19.48 lakh crore (US$ 276.37 billion) in FY20. According to the World Bank's collection of development indicators gathered from formally approved sources, agricultural land (percentage of land area) in India was recorded at 60.43 percent in 2018. Given the significance of agriculture in India, farmers might benefit from early forecasting of agricultural yields when selecting which crops to cultivate. In terms of agriculture output, India stands second in the world. Agriculture and related industries like as forestry and fisheries accounted for 15.4 percent of GDP (gross domestic product) in 2016 and employed around 31 percent of the workforce in 2014.

The study focuses on predicting agricultural yields for Karnataka state using neural network regression modeling. The crop cycle data for summer, Kharif, Rabi, fall, and the entire year is used. The main challenge encountered when assembling the work was the lack of a single source dataset to train the suggested model on. To address these issues, all dispersed data is collected and relevant feature engineering and data pre-processing steps are employed.

The final constructed dataset takes parameters like cultivation area, crop, state, district, season, year, district wise annual rainfall (mm), district wise maximum and minimum temperature (˚C) and production or yield for the period of 1998 to 2014. To obtain data for the state of Karnataka, the dataset is filtered using Python Pandas and Pandas Profiling tools. The underlying model is built utilizing a Multilayer Perceptron Neural Network, a ReLu Activation function, an Adam Optimizer, and 50 epochs with a batch size of 200. Several additional well-known regression algorithms such as Multinomial Linear Regression, Random Forest Regression and Support Vector Machine are also constructed and trained using the same dataset in order to compare their performance to the base model.

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

Agriculture in India stretches back to the Indus Valley Civilization Era, and maybe much earlier in some regions of Southern India. In terms of agriculture output, India stands second in the world. While agriculture's proportion of the Indian economy has gradually decreased to less than 15% due to the rapid expansion of the industrial and service sectors, the sector's importance in India's economic and social fabric extends far beyond this metric. The reason for this deterioration in the agriculture industry is because farmers are not empowered, and there is a lack of application of information technology in the farming sector. Farmers are less knowledgeable about the crops they cultivate. We typically overcome this challenge by utilizing appropriate deep learning algorithms to forecast crop output and name based on a variety of parameters like as temperature, rainfall, season, and location. Based on the dataset supplied by the Indian government, this study presents a Neural Network model to forecast agricultural production and crop success rate. The main challenge encountered when assembling the work was the lack of a single source dataset to train the suggested model on. To address these issues, all dispersed data is collected and relevant feature engineering and data pre-processing steps are employed. The dataset is massive, comprising data for all areas of India that were filtered to acquire data for Karnataka state, resulting in 12000 entries. The crop cycle data for summer, Kharif, Rabi, fall, and the entire year is used. To obtain data for the state of Karnataka, the dataset is filtered using Python Pandas and Pandas Profiling tools. The crop yield forecast model employs an artificial neural network's back propagation technique. The technology of multilayer perceptrons is employed. The proposed work has a wide range of applications in improving real-world farming conditions. Every year, a large amount of crop is damaged owing to a lack of understanding of weather patterns such as temperature, rainfall, and so on, which have a significant impact on crop output. This initiative not only aids in forecasting these characteristics throughout the year, but it also aids in projecting agricultural yields in various seasons based on historical trends. As a result, it enables farmers to select the best crop to plant in order to incur the fewest losses. Different regression models are also constructed using machine learning, and their efficiency and accuracy are compared to the Neural Network model in order to provide some tangible results.

1.1 Problem Statement:

Crop yield forecasting will undoubtedly benefit farmers. The farmer may make crop selection decisions and contribute more to the farm's earnings. There are several crop production prediction models available, some of which may make use of meteorological data. Parameters that are genuine, as opposed to parameters that are static. We typically overcome this challenge by utilizing appropriate deep learning algorithms to forecast crop output and name based on a variety of parameters like as temperature, rainfall, season, and location. Based on the dataset supplied by the Indian government, this study presents a Neural Network model to forecast agricultural production and crop success rate.

1.2 Objectives:

The main challenge encountered when assembling the work was the lack of a single source dataset to train the suggested model on. To address these issues, the following objectives are proposed in order to rectify the problems in the existing system.

- To gather all dispersed data is collected and perform relevant feature engineering and data pre-processing steps are employed.
- Design a neural network model and optimize it with appropriate selection of activation function, epochs, batch size and optimizer in order to increase the success rate.
- Compare the performance of the designed neural network with other classic Machine Learning models.
- To add additional module that would provide smart solutions to better the yield (if low)
- Incorporate a module that would project the profit that one can expect from the predicted yield.

1.3 Existing System:

Many models were earlier designed to resolve the current problem statement. While some focused on working with static data and performing relevant data cleaning process to enhance the model accuracy, others tried to create a model that could process remote sensing data and eradicate the hassle with the dispersed dataset and the ETL process associated with it.

The classic machine learning models were inclined to limit down its capabilities to only one or two features that would impact the crop yield directly. Other features were studied using different approach separately.

On the other hand, research were conducted in order to eliminate the trouble of dealing with static and scattered datasets and shift the attention to image processing while utilising satellite images to identify and anticipate crop output. However, it was determined that this technique required a lengthy training period with no certainty of consistency.

Considering the following issues, the existing system has the following disadvantages:

- All the features and attributes that affect the crop yield are not taken into account together.
- The issue of scattered data was not considered.
- Remote sensing data cannot be used to provide consistent result.
- Classic machine learning models were not able to detect complex patterns in the data.

1.4 Proposed Idea:

Considering the issues discussed above, the proposed system aims to rectify it by gathering the dispersed data from different sources, performing relevant feature engineering to merge those datasets in order to obtain a single source of data, Use Neural Network to make the model consistent and capable to understanding the complex correlations and provide other add on modules in order to aid the better result of the yield.

Figure 1 depicts the modules constructed in the proposed work. It is made up of three major components. One module is dedicated to early yield forecast, utilising characteristics such as crop area, yearly rainfall, temperature data, and Karnataka state output history from 1998 to 2014. The second module is the fertiliser recommendation system, which takes into account the quantity of three major nutrients in the soil, namely nitrogen, phosphorus, and potassium, as well as the crop to be sown, and suggests the fertiliser that may be utilised to increase crop output. The third module is a crop-specific WPI trends indicator that depicts the whole index price over the following 12 months graphically.

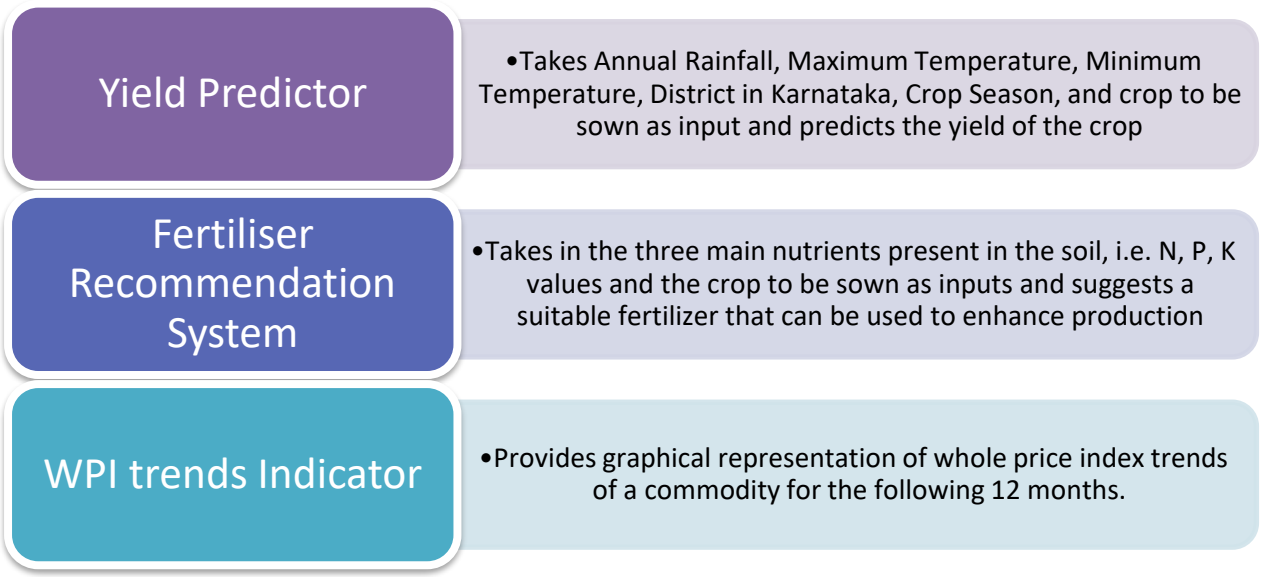


Figure 1. System Modules

1.5 Datasets:

The data for this study was obtained from the Indian government's website. The datasets are freely accessible for research and scholarly purposes. The collection contains information spanning the years 1997 to 2014. All the required datasets are collected and pre-processed to obtain a final dataset which will be used to train the model. For the experiment in this investigation, the following parameters are used.

- **Crop** – The dataset contains a number of crops such as Sunflower, Bajara, Jowar, Season, groundnut, rice, cottonseed, tur etc.
- **State**-Karnataka
- **District** – 'BAGALKOT', 'BENGALURU RURAL', 'BELLARY', 'BELGAUM', 'CHIKMAGALUR', 'CHITHADURGA', 'DHARWAD', etc
- **Season** - Kharif, Rabi, autumn, whole year
- **Year**- 1998 to 2014
- **Rainfall** - Monthly rainfall data (mm) for each district of Karnataka State, whose sum is taken to evaluate annual rainfall and concatenated to the final dataset.
- **Temperature** - District wise maximum and minimum temperature (˚C), who's mean is calculated and appended to the final dataset
- **Production** - It is given in tons per hector in lakh

- **Fertilizers** - Describes the amount of N, P, K required in the soil in order to grow a specific crop in a region.

The data in the dataset provided by the government has been examined for outliers and noise. The variables were also transformed to category and numerical formats as needed by the model. Figure 2 displays a bar graph from 1998 to 2014 that correlates the season and produce of all Karnataka districts. According to the graph below, the bulk of the crops cultivated in Karnataka are year-round crops.

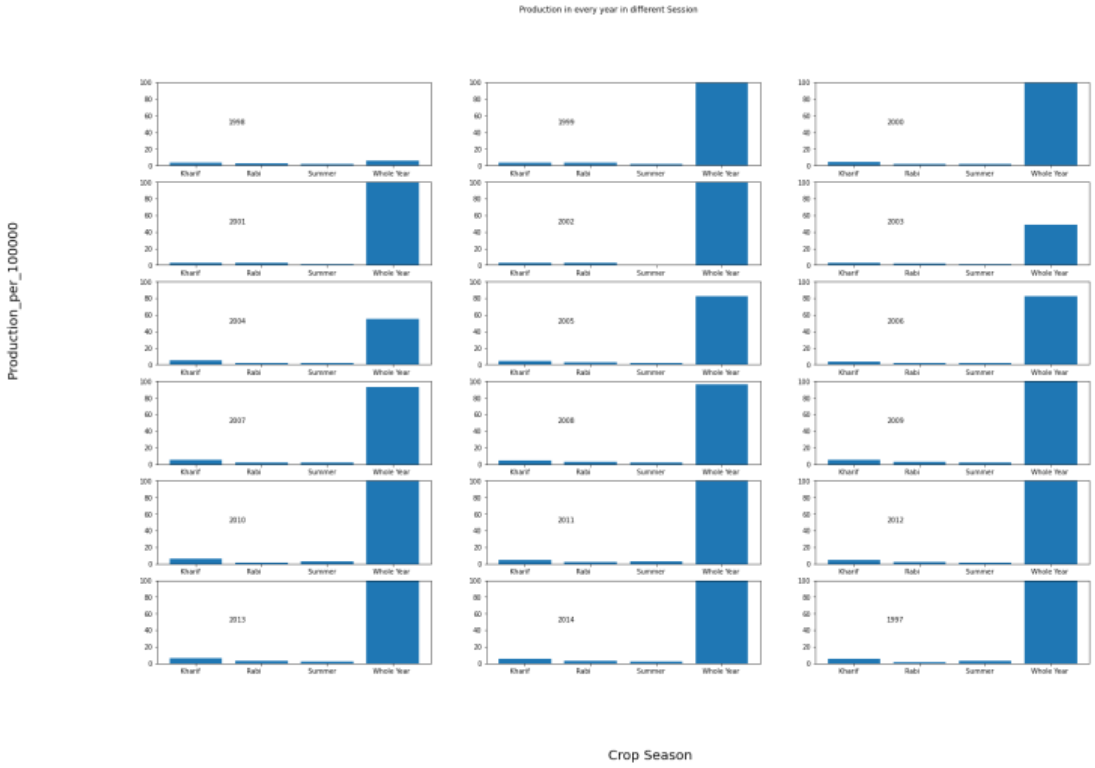


Figure 2. Year-specific Bar graph to depict the correlation of season vs. Production

Chapter 2

LITERATURE SURVEY

1. **Crop Yield Prediction to maximize profit using Machine Learning: [3]**

   Authors: Gabhane Srushti, Shaikh Naushinnaaz, Sadavarte Shivani, Khan Huda, A.I. Waghmare

   Abstract: The proposed system applies machine learning and prediction algorithms to suggest the best suitable crops for the farmers. The aim of the system is to reduce the losses due to drastic climatic changes and increase the yield rates of crops. The system integrates the data obtained from the past prediction, current weather and soil condition due to this farmers gets the idea and list of crops that can be cultivated. Machine Learning methods are widely used in prediction techniques like SVM (Support Vector Machine), linear regression. This in return gives the best crop for cultivation based on the current environment condition. The proposed system considers the rainfall amount of past, current and future and also the type of soil the farmer have. Based on this parameters the suitable crops for the given condition is predicted using the machine learning algorithms more accurate prediction results are produced.

   Disadvantages: The following System integrated various parameters but would fail to detect complex patterns if new data is provided. This issue could be resolved by using suitable Deep Learning Algorithm that would provide accurate, efficient and consistent results.

2. **Correlation Of Climatic Factors With Cereal Crops Yield: A Study From Historical Data Of Morang District, Nepal. [4]**

   Author: Badri Khanal

   Abstract: The present study is based on the secondary sources of information on temperature, rainfall and productivity of four major cereals (Rice, Maize, Finger Millet and Wheat) in Morang district of Nepal. A total of 17 years data (1995-2011) on yield of crops, annual total rainfall, annual mean maximum temperature and annual mean minimum temperature is analysed. The suitability analysis of crops shows that all the four cereal found to be suitable for cultivation in temperature range of Morang district, whereas irrigation is required in addition to recorded rainfall in case of rice and wheat. The production of three

cereals except millet (which is almost stable) has increased during the study period. The analysis of correlation coefficient shows that maize yield and minimum temperature have strong positive correlation (0.7755). The linear regression analysis showed that the yield of maize was significant and highly sensitive to combined effect of all three climatic factors (R2 0.7414)

Disadvantages: The study focuses on a large geographical area, thereby hindering the model accuracy. Also, it takes into account only two commodities i.e. rice and wheat and deduces a correlation of the same with climatic factors.

3. **Agro based crop and fertilizer recommendation system using machine learning [5]**

Authors: Preethi G, Rathi Priya V, Sanjula S M, Lalitha S D, Vijaya Bindhu B

Abstract: The paper explains how the amount of soil vitamins and environmental factors followed by the pointers for cropping and special fertilization of the site can be established. 1 The selection of the best crop for the soil and the sowing of it to provide the full yield is one of the key problems in agriculture. The proposed method takes the soil and PH samples as the input and helps to predict the crops that can be recommended suitable for the soil and fertilizer that can be used as the solution in the form of the webpage.So, the soil information is collected through sensors an the data transmitted from the Arduino through Zigbee and WSN (Wireless Sensor Network) to MATLAB and analyzing the soil data and processing is done with help of ANN (Artificial Neural Network) and crop recommendations is done using SVM (Support Vector Machine) .

Disadvantages: The following model takes account only the soil parameter and recommends suitable fertilizer to enhance the soil quality. This study suggests the ways to better the yield of the crop by indirect relation to soil and its nutrient values. However this module can be used in association to the main model.

4. **Rice Crop Yield Prediction Using Artificial Neural Networks : [6]**

Authors: Niketa Gandhi, Owaiz Petkar, Leisa J. Armstrong

Abstract: This study aimed to use neural networks to predict rice production yield and investigate the factors affecting the rice crop yield for various districts of Maharashtra state in India. Data were sourced from publicly available Indian Government's records for 27 districts of Maharashtra state, India. The parameters considered for the present study were

precipitation, minimum temperature, average temperature, maximum temperature and reference crop evapotranspiration, area, production and yield for the Kharif season (June to November) for the years 1998 to 2002. The dataset was processed using WEKA tool.

Disadvantages: This approach has been demonstrated by forecasting of rice crop yield prediction for Kharif season from year 1998 to 2002 for Maharashtra state of India, on the basis of different predictor variables including precipitation, minimum temperature, average temperature, maximum temperature, reference crop evapotranspiration and yield. Artificial Neural Networks with Multilayer Perceptron were considered for the present research.

5. **Rice Crop Yield Prediction in India using Support Vector Machines [7]**

Authors: Niketa Gandhi, Owaiz Petkar, Leisa J. Armstrong, Amiya Kumar Tripathy

Abstract: This paper presents the review on use of such machine learning technique for Indian rice cropping areas. This paper discusses the experimental results obtained by applying SMO classifier using the WEKA tool on the dataset of 27 districts of Maharashtra state, India. The dataset considered for the rice crop yield prediction was sourced from publicly available Indian Government records. The parameters considered for the study were precipitation, minimum temperature, average temperature, maximum temperature and reference crop evapotranspiration, area, production and yield for the Kharif season (June to November) for the years 1998 to 2002. For the present study the mean absolute error (MAE), root mean squared error (RMSE), relative absolute error (RAE) and root relative squared error (RRSE) were calculated. The experimental results showed that the performance of other techniques on the same dataset was much better compared to SMO.

Disadvantages: The following study demonstrates the use of Support Vector machine which is a classic machine learning algorithm to depict the rice production using Weather as well as historic data. However, the model can become inconsistent due to the static nature of dataset and the incapability of the model to learn from newly added data.

6. **Wheat Yield Prediction: Artificial Neural Network based Approach: [8]**

Authors: Muhd Khairulzaman Abdul Kadir, Mohd Zaki Ayob, Nadaraj Miniappan

Abstract: In this study, our wheat yield prediction model is designed using a Multi-Layer Perceptron (MLP) backpropagation-based- feed forward artificial neural network (ANN). The data used was weather data including: sun, frost, rain and temperature as the input parameters

from year 1997-2007. The output parameter of the model is using the wheat yield data for the years 1997 – 2007. The data is divided into three separate sets; – for training, validation and testing. Our MLP was able to predict, wheat yield with an accuracy of 98 %. Hence our MLP based wheat yield prediction model shows great promise as a tool which will be able to provide relatively accurate wheat yield prediction and may be applied to other crops.

Disadvantages: The model is perfectly defined to suit for one commodity. However, a similar model can be used to include various types of crops with similar accuracy.

7. **Understanding Satellite-Imagery-Based Crop Yield Predictions [9]**

Authors: Mark Sabini, Gili Rusak and Brad Ross

Abstract: We aim to improve upon and better understand [26]'s methodology and results. In line with their work, we use nine spectral and temperature bands from relatively low resolution satellite images as our features for predicting county-level corn and soybean yields. To ease training, we reduce the dimensionality of our data by assuming that the position of pixels doesn't impact the average yield (the permutation invariance assumption), which allows us to use pixel intensity histograms as features. By making [26]'s model deeper, we achieve better prediction accuracy, showing that there is still signal to extract from the data. To better understand whether our models can distinguish between crops, we compute saliency maps for each image/crop pair and compare maps for various crops. We find that our model distinguishes between crops, and that, in line with previous yield prediction research, the infrared and temperature bands of images taken during peak growing season contribute the most to discrimination ability.

Disadvantages: The following study utilizes the satellite imagery, and implements a convolution neural network, to detect the crop yield. However the training period of the model can be extremely time consuming and may require sophisticated hardware requirements to be deployed in full time.

Chapter 3

REQUIREMENTS

Software Requirement Specification

A Software Requirements Specification (SRS) -a requirements specification for a software system – is a complete description of the behaviour of a system to be developed. In addition to a description of the software functions, the SRS also contains non-functional requirements. Software requirements are a sub-field of software engineering that deals with the elicitation, analysis, specification, and validation of requirements for software.

Requirements:

Hardware Requirements

- System : Pentium IV 2.4 GHz or more
- Hard Disk : 40 GB.
- Monitor : 15 VGA Color.
- Mouse : Logitech.
- Ram : 512 Mb

Software Requirements

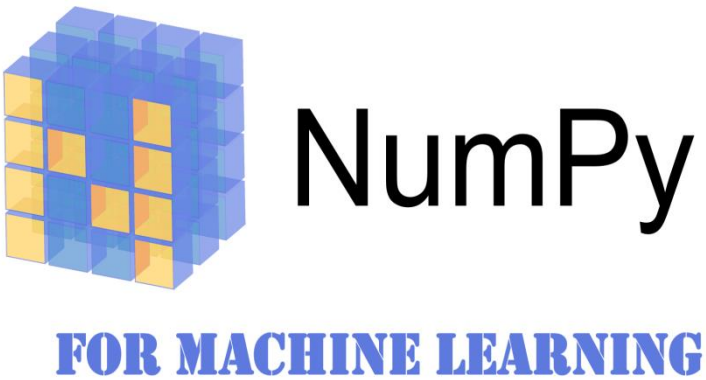
- Keras
- Tensorflow
- Visual Studio
- Anaconda Navigator

Libraries and Packages:

Numpy:

NumPy (pronounced /ˈnʌmpaɪ/ (*NUM-py*) or sometimes /ˈnʌmpi/[3][4] (*NUM-pee*)) is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.[5] The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy targets the CPython reference implementation of Python, which is a non-optimizing bytecode interpreter. Mathematical algorithms written for this version of Python often run much slower than compiled equivalents. NumPy addresses the slowness problem partly by providing multidimensional arrays and functions and operators that operate efficiently on arrays; using these requires rewriting some code, mostly inner loops, using NumPy.

Using NumPy in Python gives functionality comparable to MATLAB since they are both interpreted,[19] and they both allow the user to write fast programs as long as most operations work on arrays or matrices instead of scalars. In comparison, MATLAB boasts a large number of additional toolboxes, notably Simulink, whereas NumPy is intrinsically integrated with Python, a more modern and complete programming language. Moreover, complementary Python packages are available; SciPy is a library that adds more MATLAB-like functionality and Matplotlib is a plotting package that provides MATLAB-like plotting functionality. Internally, both MATLAB and NumPy rely on BLAS and LAPACK for efficient linear algebra computations.



Pandas:

pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals.[3] Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007. Pandas is mainly used for data analysis. Pandas allows importing data from various file formats such as comma-separated values, JSON, SQL, Microsoft Excel. Pandas allows various data manipulation operations such as merging,[9] reshaping, selecting, as well as data cleaning, and data wrangling features

Developer Wes McKinney started working on pandas in 2008 while at AQR Capital Management out of the need for a high performance, flexible tool to perform quantitative analysis on financial data. Before leaving AQR he was able to convince management to allow him to open source the library.Another AQR employee, Chang She, joined the effort in 2012 as the second major contributor to the library.In 2015, pandas signed on as a fiscally sponsored project of NumFOCUS, in the United States.

Matplotlib:

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib.

Matplotlib was originally written by John D. Hunter. Since then it has an active development community[4] and is distributed under a BSD-style license. Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012[5] and was further joined by Thomas Caswell.[6][7]

Matplotlib 2.0.x supports Python versions 2.7 through 3.10. Python 3 support started with Matplotlib 1.2. Matplotlib 1.4 is the last version to support Python 2.6.[8] Matplotlib has pledged not to support Python 2 past 2020 by signing the Python 3 Statement.

Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.

Seaborn:

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.For a brief introduction to the ideas behind the library, you can read the introductory notes. Visit the installation page to see how you can download the package and get started with it. You can browse the example gallery to see what you can do with seaborn, and then check out the tutorial and API reference to find out how.

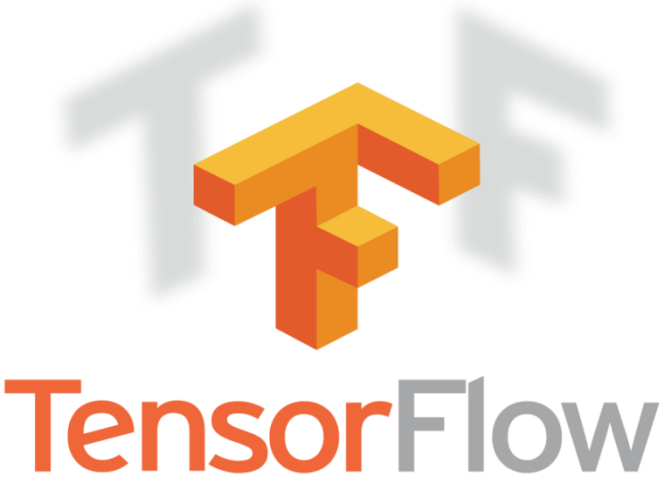
Data visualization has been one of the most important driving forces in the field of Data Analytics. It powers millions of businesses across the globe and provides in-depth insights into the data at hand. This makes it vital that you know about the latest and the most popular data visualization libraries out there: Seaborn in Python is one of these

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures.

Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.

TensorFlow:

TensorFlow is a free and open-source software library for machine learning. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. Tensorflow is a symbolic math library based on dataflow and differentiable programming. It is used for both research and production at Google. TensorFlow was developed by the Google Brain team for internal Google use. Its flexible architecture allows for the easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. TensorFlow computations are expressed as stateful dataflow graphs. The name TensorFlow derives from the operations that such neural networks perform on multidimensional data arrays, which are referred to as *tensors*. During the Google I/O Conference in June 2016, Jeff Dean stated that 1,500 repositories on GitHub mentioned TensorFlow, of which only 5 were from Google.[16]In December 2017, developers from Google, Cisco, RedHat, CoreOS, and CaiCloud introduced Kubeflow at a conference. Kubeflow allows operation and deployment of TensorFlow on Kubernetes. In May 2016, Google announced its Tensor processing unit (TPU), an application-specific integrated circuit (ASIC, a hardware chip) built specifically for machine learning and tailored for TensorFlow. A TPU is a programmable AI accelerator designed to provide high throughput of low-precision arithmetic (e.g., 8-bit), and oriented toward using or running models rather than training them. Google announced they had been running TPUs inside their data centers for more than a year, and had found them to deliver an order of magnitude better-optimized performance per watt for machine learning.



Keras:

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Up until version 2.3, Keras supported multiple backends, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidML.[2][3][4] As of version 2.4, only TensorFlow is supported. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System),[5] and its primary author and maintainer is François Chollet, a Google engineer. Chollet is also the author of the XCeption deep neural network model. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. The code is hosted on GitHub, and community support forums include the GitHub issues page, and a Slack channel.

In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling.Keras allows users to productize deep models on smartphones (iOS and Android), on the web, or on the Java Virtual Machine.[3] It also allows use of distributed training of deep-learning models on clusters of Graphics processing units (GPU) and tensor processing units.

Chapter 4

PROJECT DETAILS

4.1 System Design:

System design is the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements. System design could see it as the application of systems theory to product development. Theory is some overlap with the disciplines of system analysis, systems architecture and systems engineering.

If the broader topic development “blends the perspective of marketing, design, and manufacturing into a single approach to product development,” then design the act of talking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user.

Until the 1990s systems design had crucial and respected role in the data processing industry. In the 1990s standardization of hardware and software resulted in the ability to build modular systems. The increasing importance of software running on generic platforms has enhanced the discipline of software engineering.

Object-oriented analysis and design methods are becoming the most widely used methods for computer systems design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modelling software systems and is increasingly used for high designing non- software systems and organizations.

System design is one of the most important phases of software development process. The purpose of the design is to plan the solution of a problem specified by the requirement documentation. In other words the first step in solution is the design of the project.

The design of the system is perhaps the most critical factor affecting the quality of the software. The objective of the design phase is to produce overall design of the software. It aims to figure out the modules that should be in the system to fulfil all the system requirements in efficient manner.

The design will contain the specification of all the modules, their interaction with other modules and the desired output from each module.

4.2 Data flow diagram

A data flow diagram (DFD) is a graphical representation of the flow of the visualization of data processing.On a DFD, data items flow from an external data source or internal data source to internal data source or external data sink via an internal process. DFD provides no information about the timing of process or about whether process will operate in sequence or in parallel.

The diagram below depicts the flow of data through the system. The flow of all modules stays constant, with the only variation being the final result. Inputs for the relevant modules, such as yearly rainfall, temperature, district, crop name, season, and fertiliser data, are obtained via a web-based application by the user. A JSON data object is returned, which has been scaled with the sklearn package. The categorical data, such as district, season, and crop name, is again one hot encoded, and the data object is ultimately transformed to a numpy array. This information is subsequently put into the Neural Network model.

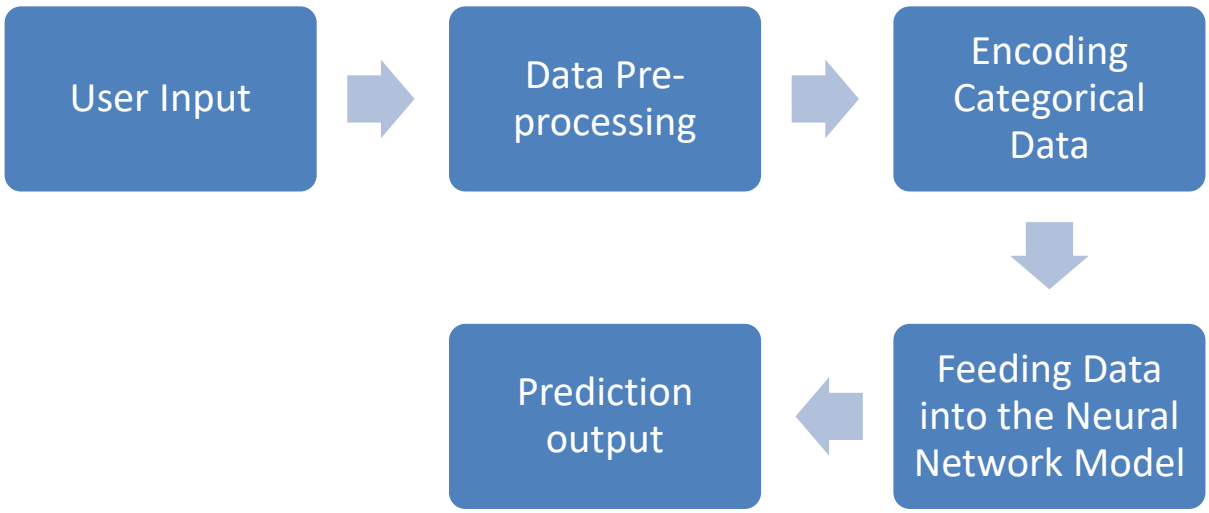


Figure 3. Dataflow Diagram of the proposed System

Chapter 5

SYSTEM IMPLEMENTATION

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy. In other words, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment. Many implementations may exist for a given specification or standard.

Implementation is one of the most important phases of the Software Development Life Cycle (SDLC). It encompasses all the processes involved in getting new software or hardware operating properly in its environment, including installation, configuration, and running, testing, and making necessary changes. Specifically, it involves coding the system using a particular programming language and transferring the design into an actual working system. This phase of the system is conducted with the idea that whatever is designed should be implemented; keeping in mind that it fulfils user requirements, objective and scope of the system. The implementation phase produces the solution to the user problem.

5.1 Pseudo code

Pseudo code is an informal high-level description of the operating principle of a computer program or other algorithm. It uses the structural conventions of a programming language, but is intended for human reading rather than machine reading. Pseudo code typically omits details that are not essential for human understanding of the algorithm, such as variable declarations, system-specific code and some subroutines. The programming language is augmented with natural language description details, where convenient, or with compact mathematical notations. The purpose of using pseudo code is that is easier for people to understand than conventional programming language code, and that it is an efficient and environment independent description of the key principles of an algorithm. It is commonly used in textbooks and scientific publications that are documenting various algorithms, and also in planning of computer program development, for sketching out the structure of the program before the actual coding takes place. No standard for pseudo code syntax exists, as a program in pseudo code is not an executable program. Pseudo code resembles, but should not be confused with skeleton programs, including dummy code, which can be compiled without errors. Flowcharts and

Unified Modelling Language (UML) charts can be thought of as a graphical alternative to pseudo code, but are more spacious on paper.

The Project is divided into 3 different modules:

1. Crop Yield Predictor (Base Module)
2. Fertilizer Recommendation system
3. Whole price index trend analysis

All the module follow the same Data pre-processing steps. The processed data is then fed to the respective Deep Learning models in order to obtain the required results.

5.2 Data Pre-processing:

[1]. The raw data set was then collated in single sheet which consisted of the following columns in Microsoft Excel: sr. no, name of the state, name of the district, year, precipitation, minimum temperature, average temperature, maximum temperature, soil type, area, production and yield.

[2]. For some of the districts particular year's climatic parameters or production data was not available hence those records were omitted. That particular year's data was not used for the current research. Record number was added for each record.

[3]. For preparing the data set for applying multilayer perceptron technique, unrequited columns were removed. They were sr. no, name of the district and year.

[4]. The data set was then sorted on the basis of area. Area less than 100 hectares were not considered for the present research. So those records were omitted.

[5]. the data which is present in label from converted to encoding using sklearn .

[6]. The dataset was then sorted on the basis production.

[7]. we considered production as output parameter and features like: crop, area, district, season.

[8]. This data set was then saved in .csv format for further application of the multilayer perceptron technique in Python TensorFlow.

5.3 Crop Yield Predictor:

This is the base module, which includes comparison to other classic Machine Learning Algorithm as well. This module deals with scattered datasets which is cleaned and processed using the data pre-processing step discussed above. The final dataset is then fed to the neural network model, with the following training parameters.

- batch_size=100,
- epochs=50
- Layer : 3
- Neuron at each layer : Layer 1, Layer 2 = 20
- Layer 3 = 1
- Oprimizer = Adam
- Activation : ReLu
- kernel_initializer='uniform
- lr rate : 0.01

The algorithm used for the following module is the Artificial Neural Network. An Artificial Neuron is basically an engineering approach of biological neuron. It has device with many inputs and one output. ANN is consisting of large number of simple processing elements that are interconnected with each other and layered also. [10]. An ANN begins with a training phase in which it learns to detect patterns in data, whether visually, audibly, or textually. During this supervised phase, the network compares its actual output to what it was supposed to produce—the expected output. Backpropagation is used to correct the discrepancy between the two results. This implies that the network works backward, from the output unit to the input units, adjusting the weight of its connections between the units until the discrepancy between the actual and planned outcome generates the smallest mistake possible.

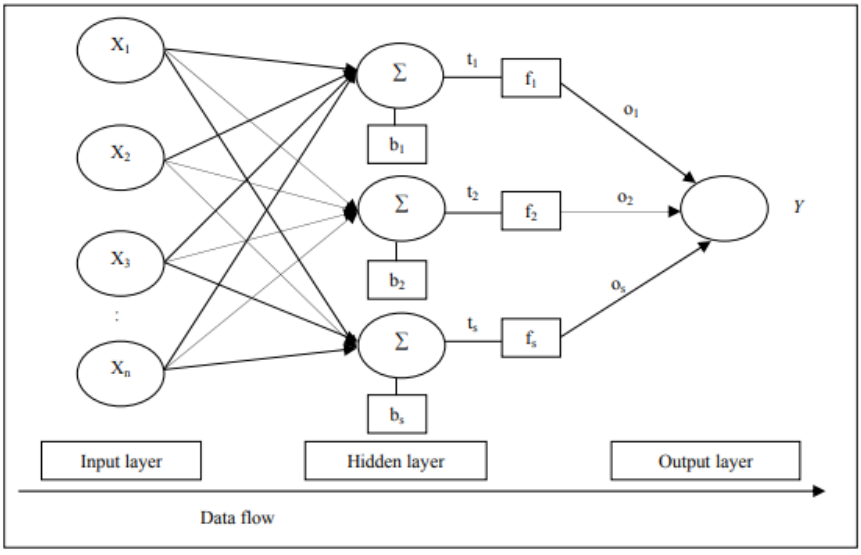


Figure 4.Layers and connections of ANN model

5.4 Fertilizer Recommendation System:

The following module uses three essential nutrients required for a healthy soil i.e. Nitrogen, Phosphorous, and Potassium. The amount of these three nutrients is taken from the user in order to determine, the soil health. After analyzing the soil nutrient contents the module suggest an appropriate fertilizer to balance the soil quality for a better yield. The following module uses Logistic Regression with Gradient Descent whose Pseudo code is given below. Logistic regression is a traditional and classic statistical model, which has been widely used in the academy and industry. Unlike linear regression, which is used to make a prediction on the numeric response, logistic regression is used to solve a classification problem.

1.Initialize the parameters

2. Repeat

    2.2. Make a prediction on y

    2.3 Calculate cost function

    2.4. Get gradient for cost function

    2.5. Update parameters

5.5 Whole Price Index Analysis

The following module uses Decision tree to analyze the ongoing Price of different commodity grown through the breath and length of Karnataka. It provides a 12 month analysis of the Whole price trends, thereby aiding the agronomic workers have a rough idea on the profit one must expect by cultivating a certain crop. The pseudo code for the algorithm used is given below:

1. It begins with the original set S as the root node.
2. On each iteration of the algorithm, it iterates through the very unused attribute of the set S and calculates **Entropy(H)** and **Information gain(IG)** of this attribute.
3. It then selects the attribute which has the smallest Entropy or Largest Information gain.
4. The set S is then split by the selected attribute to produce a subset of the data.
5. The algorithm continues to recur on each subset, considering only attributes never selected before.

Chapter 6

PERFORMANCE EVALUATION

Model evaluation aims to estimate the generalization accuracy of a model on future (unseen/out-of-sample) data.Methods for evaluating a model's performance are divided into 2 categories: namely, holdout and Cross-validation. Both methods use a test set (i.e data not seen by the model) to evaluate model performance. It's not recommended to use the data we used to build the model to evaluate it. This is because our model will simply remember the whole training set, and will therefore always predict the correct label for any point in the training set. This is known as overfitting.

We utilise scatter plots to compare the actual test data output to the predictions generated by the model on the test data. The graph below illustrates a linear connection between the actual and predicted results. A positive slope with a strong correlation between the actual and anticipated results indicates a greater success rate. It can also be stated that for the majority of the test inputs, the model was able to forecast a yield with extremely low error to the actual yield output.

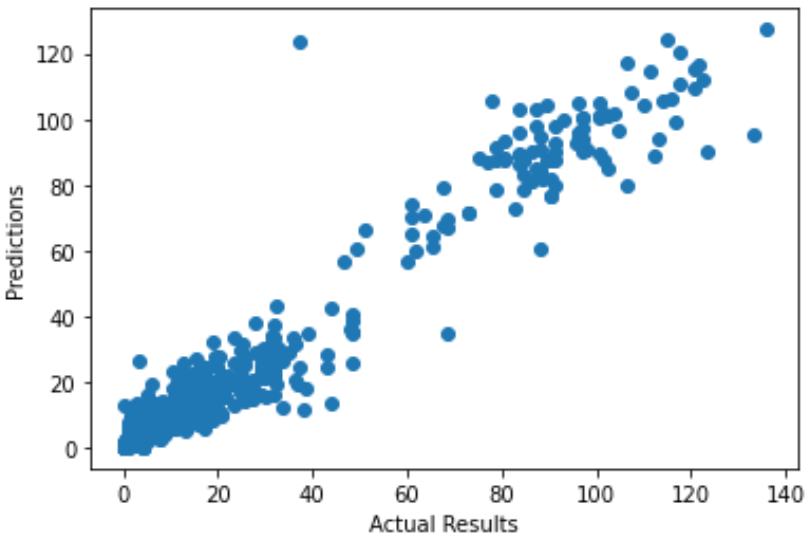


Figure 4.Linear Correlation between actual and predicted results

The algorithm's performance is measured using the two metrics listed below.

6.1 Mean Absolute Error:

In the context of machine learning, absolute error refers to the magnitude of difference between the prediction of an observation and the true value of that observation. MAE takes the average of absolute errors for a group of predictions and observations as a measurement of the magnitude of errors for the entire group. MAE can also be referred as L1 loss function.

As one of the most commonly used loss functions for regression problems, MAE helps users to formulate learning problems into optimization problems. It also serves as an easy-to-understand quantifiable measurement of errors for regression problems.

MAE measures the average magnitude of absolute differences between N predicted vectors S ={x1, x2, ..., xN} and S_ = {y1, y2, ..., yN}, the corresponding

loss function is defined as:

$$L_{MAE}(S, S^*) = \frac{1}{N} \sum_{i=1}^{N} || \ x_i - \ y_i \ ||$$

where || · || denotes L1 norm. [11].

6.2 R-Squared:

R-squared (R2) is a statistical metric that indicates the proportion of the variation explained by an independent variable or variables in a regression model for a dependent variable. Whereas correlation describes the strength of the link between an independent and dependent variable, R-squared explains how well one variable's variation explains the variance of the other. R-squared is commonly defined as the percentage of a fund's or security's movements that can be explained by changes in a benchmark index. The R-Squared score for the proposed work was able to reach approximately 0.9645 within 50 epochs. The score can be calculated using the formula below:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Where,

R^2 is coefficient of determination

RSS is Sum of Squares of residual

TSS is total sum of squares

The graph below depicts the metric scores discussed above for our neural network model. It plots the Accuracy calculated using R-Square metrics and the mean absolute error with the numbers of epochs takes to train the model.

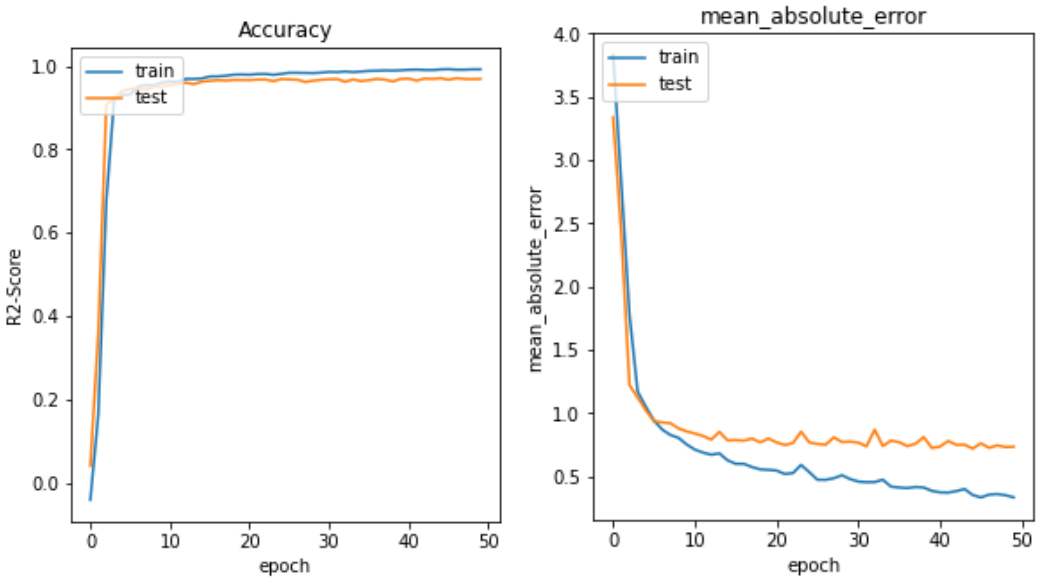


Figure 5. Performance graphs of the model

Various other Machine Learning model is also trained and evaluated using the metrics discussed above. A bar graph is plotted for the performance of all the models trained to predict the yield results. A side by side comparison clearly tells that Neural Network has outperformed classic Machine Learning models in terms of Accuracy and Minimum error.

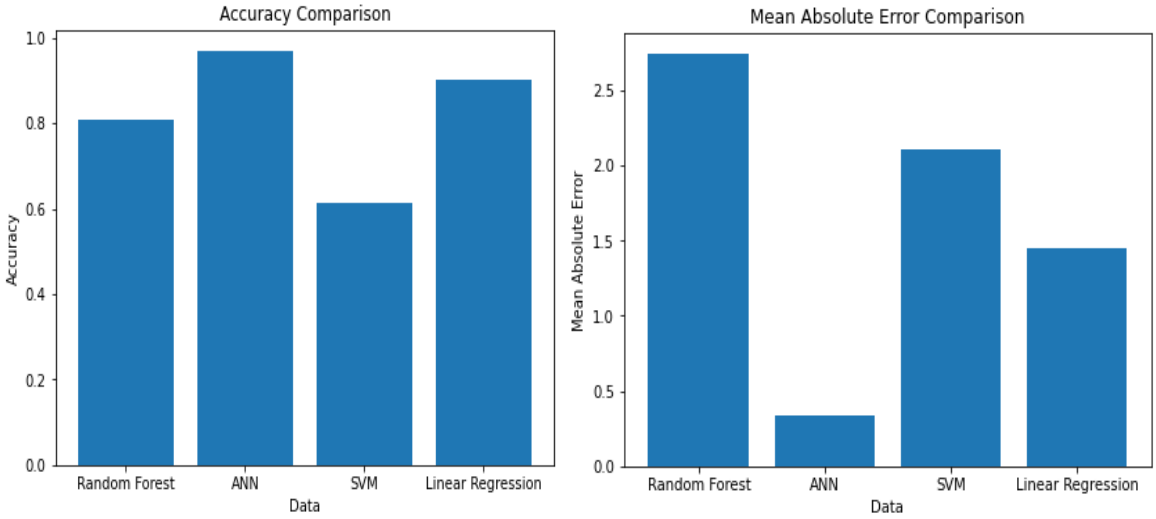


Figure 6. Comparative Analysis of the performance of all the regression models

Chapter 7

OBSERVATIONS AND RESULTS

The project uses several frontend libraries and packages such as chartJS, MaterialiseJS, Bootstrap and JQuery to design a web based application to ease out the access to the users. The figure below shows the landing page of the project. The project is named as Farm Smart.

Figure 7. Landing Page of the project

The landing page has many navigation points from which the user can easily navigate to different modules without the break of flow. One such navigation element is the sticky Navigation bar with 3 links to respective 3 module. However the main focus on navigation is the card elements used to navigate to different module as shown below.

As it is clearly seen that the landing page provides access to three main services of focus. Yield Predictor is the base module whereas the fertilizer recommender and Whole Price Index Analysis are the add on modules. The buttons on the cards navigates a user to the respective page of the module wherein the user can provide there inputs and obtain the required results.

Bootstrap and JQuery is used to design the page and its elements like the cards and navigation bar.

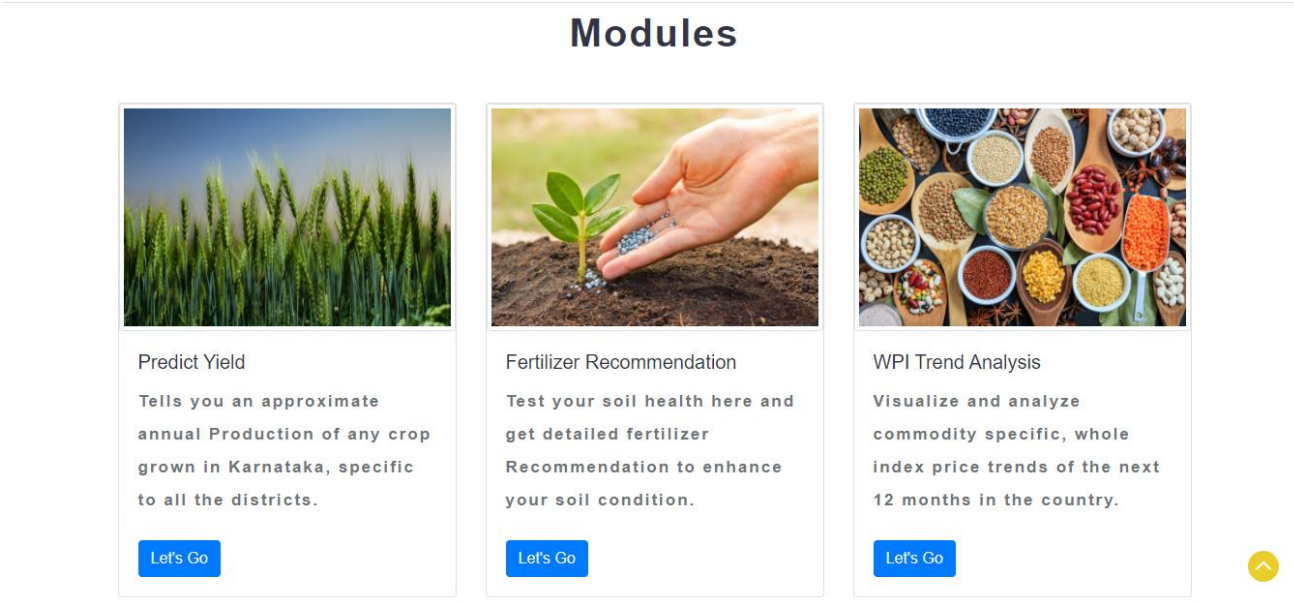


Figure 8. Module navigation links

Each of the link addresses to each of the module. The figure below shows the page for the yield predictor. It has a form that takes inputs required by the neural network to predict a yield. All the form inputs must be filled in order to obtain a result.

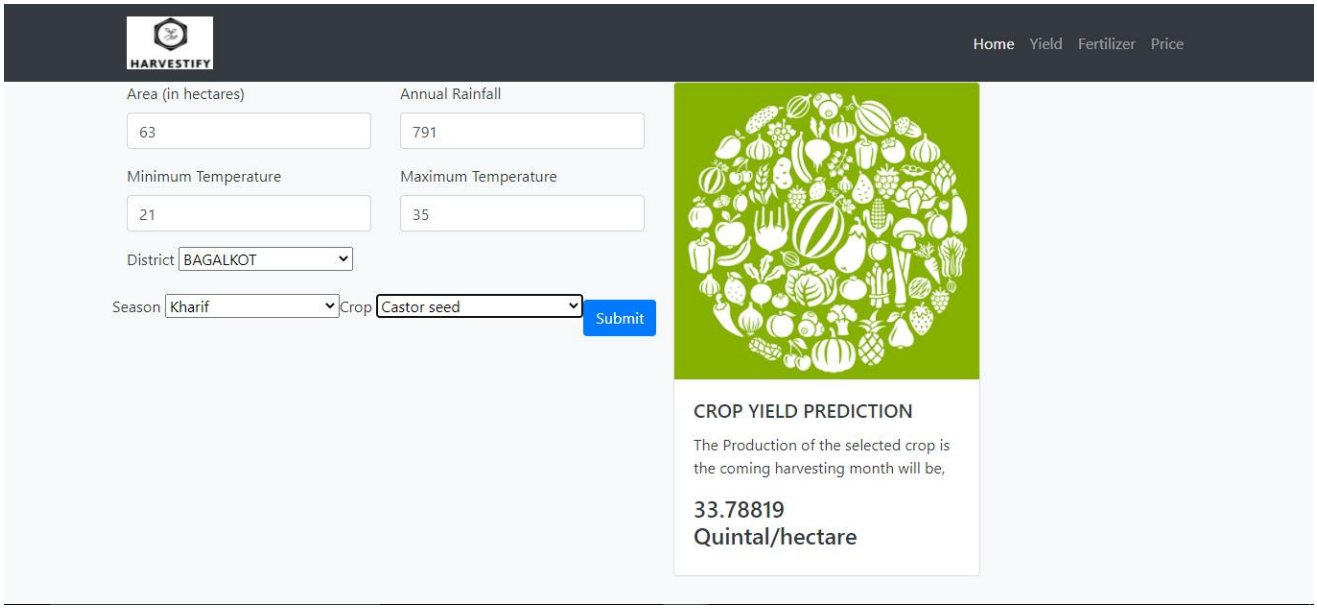


Figure 9. Yield Predictor Page

The next three figures shows the Whole Price index analysis. The whole price trend analysis welcomes the user by first showing the top gainers and bottom most gainers on the next 6 months, and provides a slide show card that rotates to show the rise and fall of the WPI in the next 6 months.

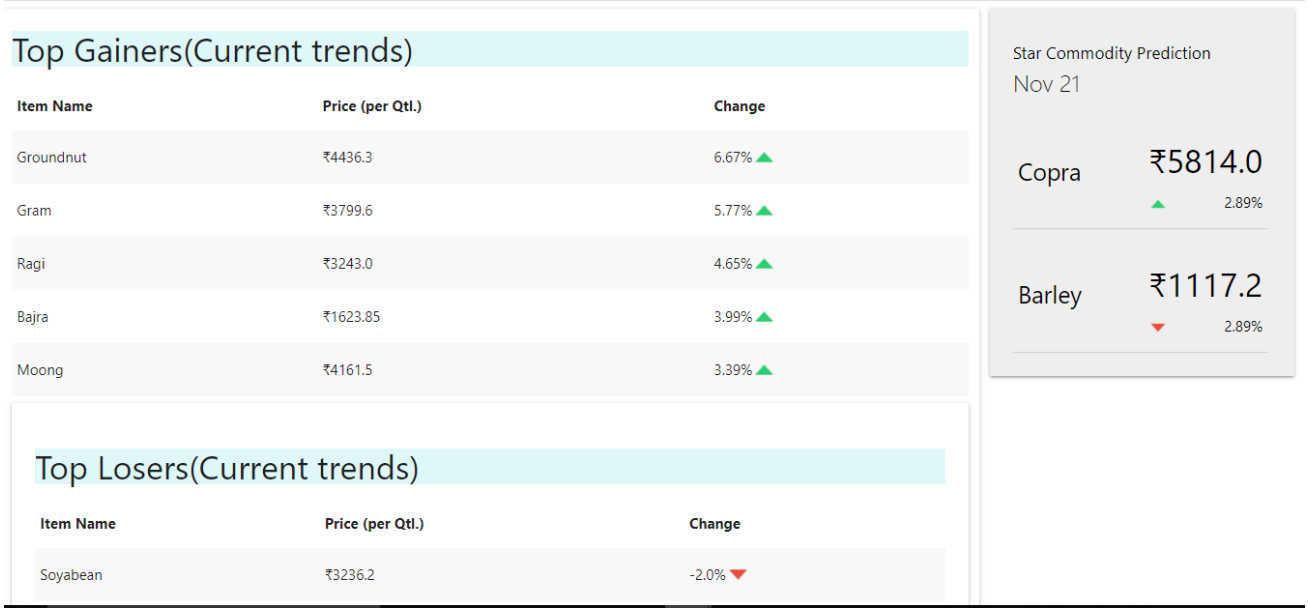


Figure 10. Whole price index Analysis landing page.

The user can also view the rise or fall of the WPI for each commodity. The top commodities or crops grown in Karnataka are listed in the following sub-section. Each of the link takes the user to a detailed information of the profit analysis of a crop.

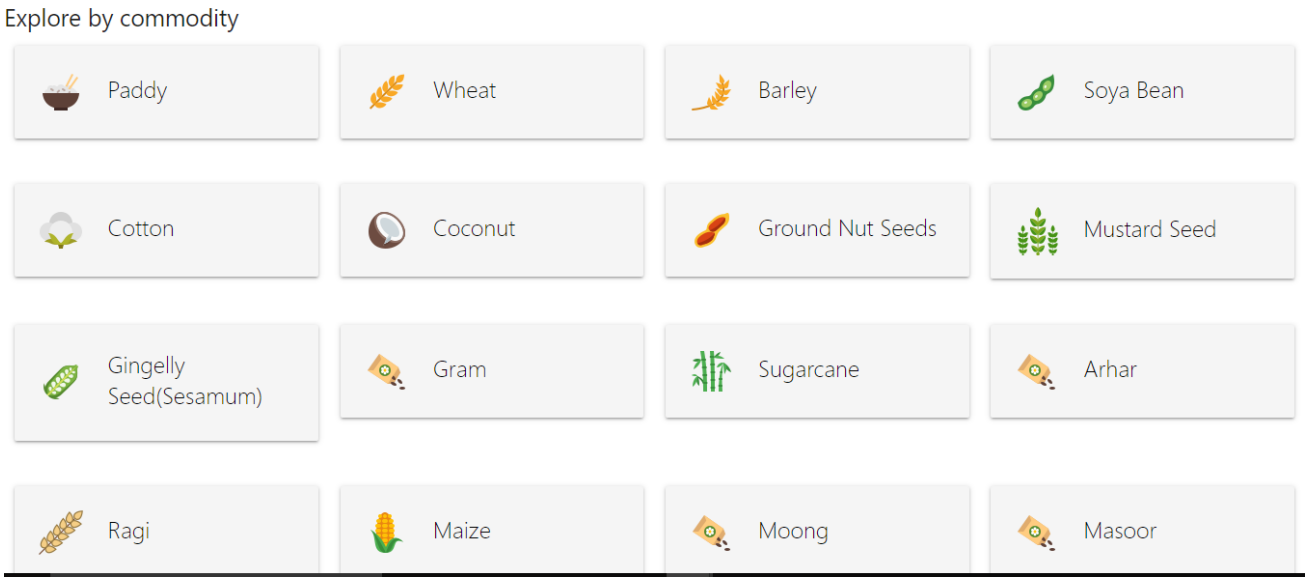


Figure 11. Commodity wise Navigation of WPI Analysis

As said the user gets a detailed information on any crop via the “explore by commodity” tab. The detailed page shows the import and export details of a crop and what was there price is the previous years, as shown below.

wheat

Current Price	₹ 1495.8 / ql
Prime Location	BAGALKOT., BELGAUM, , DHARWAD, , GULBARGA
Crop Type	rabi
Export	Sri Lanka, United Arab Emirates, Taiwan

Brief Forecast

Min. crop price time	Oct 21	₹1476.9
Max. crop price time	Dec 21	₹1672.65

Figure 12. Detailed Crop WPI analysis section

The final page shows the WPI Analysis graphically using ChartJS. It iterates over the decision tree algorithm to provide a 12 month price prediction of a graph using there base year price and the previous year trends.

Forecast Trends

Month	Price (per Qtl.)	Change
Sep 21	₹1495.8	0.0% ▲
Oct 21	₹1476.9	-1.26% ▼
Nov 21	₹1539.0	2.89% ▲
Dec 21	₹1672.65	11.82% ▲
Jan 22	₹1551.15	3.7% ▲
Feb 22	₹1572.75	5.14% ▲
Mar 22	₹1556.55	4.06% ▲
Apr 22	₹1574.1	5.23% ▲
May 22	₹1564.65	4.6% ▲

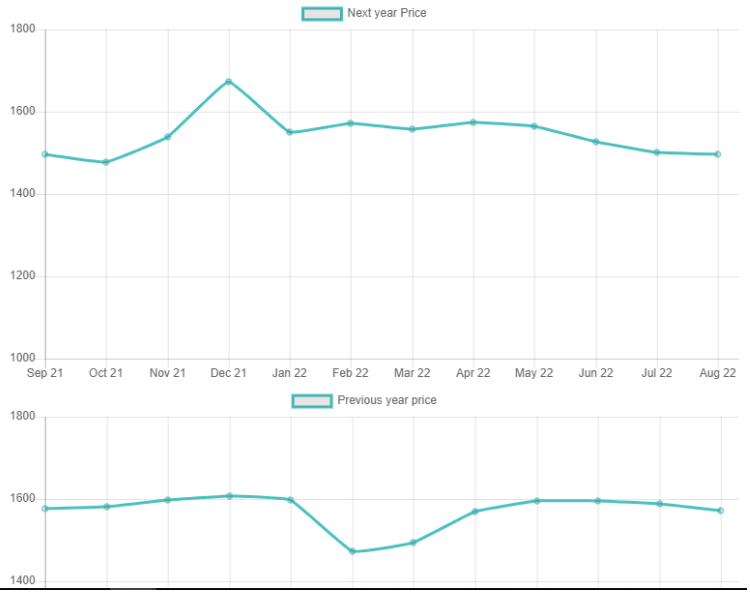


Figure 13. Graphical View of the WPI Analysis.

Chapter 8

CONCLUSION AND FUTURE SCOPE

In the current study, we collected multiple datasets and performed appropriate feature engineering to build a single source of data that accounts for all of the essential features to help model correctness.
To provide a comparison study of our neural network model's performance, we utilise the same dataset to train three additional regression models, namely, the Multinomial Regression model, the Random Forest regression model, and the support vector regression model. All three models' performance was evaluated using the same two measures described above: mean absolute error and R-Squared score.

It was clearly seen in the graphical evaluation that Neural Network not just outperformed other classical Machine Learning algorithms in terms of Accuracy which was found to be 96.24%, but also was able predict the result with minimum Error.

A non-linear way of interpreting the connection is necessary to demonstrate the interactions between the factors impacting crop production. Due to the complexity of the factors impacting crop production, a linear technique such as linear regression was deemed insufficient to illustrate the interactions between the components and crop yield. For forecasting agricultural yield, ANN was thought to be a viable alternative to standard regression methods. A neural network not only predicts non-linear correlation successfully, but it can also recognize complicated patterns in data and train appropriately, something most traditional approaches fail to do.

The following work can also be extended by using appropriate hardware to dynamically fetch the data from the attributes that affect the crop yield such as soil, temperature, precipitation and rainfall.
Forecasting using Remote Sensing data can also be improved in order to eradicate the hassle of handling the static data. However, these satellite images can be used in associations with the in-land data to provide a new dimension to the following work.

Chapter 9

BIBLIOGRAPHY

[1]. Manish Mishra, Monika Srivastava, "A View of Artificial Neural Network", IEEE International Conference on Advances in Engineering & Technology Research (ICAETR - 2014), August 01-02, 2014

[2]. R. J. Brooks, et al., "Simplifying Sirius: sensitivity analysis and development of a meta-model for wheat yield prediction," European Journal of Agronomy, vol. 14, pp. 43-60, 2001.

[3]. Gabhane Srushti, Shaikh Naushinnaaz, Sadavarte Shivani, Khan Huda, A.I. Waghmare, "Crop Yield Prediction to maximize profit using Machine Learning", International Research Journal of Modernization in Engineering Technology and Science Volume:02/Issue:06/June-2020

[4]. "Correlation Of Climatic Factors With Cereal Crops Yield: A Study From Historical Data Of Morang District, Nepal", Badri Khanal, *The Journal of Agriculture and Environment Vol: 16, June 2015*

[5]. Agro based crop and fertilizer recommendation system using machine learning, Preethi G, Rathi Priya V, Sanjula S M, Lalitha S D, Vijaya Bindhu B, European Journal of Molecular & Clinical Medicine ISSN 2515-8260

[6]. *Rice Crop Yield Prediction Using* Artificial Neural Networks, Nikita Gandhi, Owais Petkar, Leisa J. Armstrong, 2016 IEEE International Conference on Technological Innovations in ICT For Agriculture and Rural Development (TIAR 2016)

[7]. Rice Crop Yield Prediction Using Artificial Neural Networks, Nikita Gandhi, Owais Petkar, Leisa J. Armstrong, Amiya Kumar Tripathi, 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)

[8]. Wheat Yield Prediction: Artificial Neural Network based Approach, *Muhd Khairulzaman Abdul Kadir, Mohd Zaki Ayob, Nadaraj Miniappan, 2014 4th International Conference on Engineering Technology and Technopreneuship (ICE2T)*

[9]. Understanding Satellite-Imagery-Based Crop Yield Predictions, Mark Sabini, Gili Rusak and Brad Ross

[10]. Paul C. Doraiswamy, Sophie Moulin, Paul W. Cook, and Alan Stern, "Crop Yield Assessment from Remote Sensing", Photogrammetric Engineering & Remote Sensing Vol. 69, No. 6, June 2003, pp. 665–674.

[11]. Hordri N. F., Samar, A., Yuhaniz S. S., Shamsuddin S. M., "A Systematic Literature Review on Features of Deep Learning in Big Data Analytics", Int. J. Advance Soft Compu. Appl, Vol. 9, No. 1, March 2017, ISSN: 2074-8523

[12]. Teresa Priyanka, Pratishtha Soni, C. Malathy , "Agricultural Crop Yield Prediction Using Artificial Intelligence and Satellite Imagery", Eurasian Journal of Analytical Chemistry, 2018 13 (SP): 6-12, ISSN: 1306-3057

[13]. Raorane A.A., Kulkarni R.V., "Review- Role of Data Mining in Agriculture", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (2) , 2013, 270 - 272, ISSN: 0975-964

[14]. Jun Qi, Jun Du, Sabato Marco Siniscalchi, Xiaoli Ma, Chin-Hui Lee, "On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression", IEEE SIGNAL PROCESSING LETTERS, VOL. 27, 2020