iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** IX **Month of publication:** September 2021

DOI: https://doi.org/10.22214/ijraset.2021.37911

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

Analysing the Accuracy of Crop Yield Prediction Using Deep Learning Algorithm

Sehkammal A
Department of Agriculture Engineering, Bannari Amman Institute of Technology, Sathyamangalam

Abstract: ***The Indian farming level decreases step by step inferable from certain components like inordinate usage of pesticides, water level decrement, environment changes, and unpredicted precipitation, and so forth on the farming information, elucidating investigation is performed to comprehend the creation level. The creation of yields isn't expanded inferable from these issues that influences the economy of farming. By utilizing AI strategies, the harvest from given dataset need to foresee by farming areas for forestalling this issue. Yield forecast is of extraordinary importance for yield planning, crop market arranging, crop protection, and gather the executives. Data mining based deep learning is turning out to be progressively significant in crop yield forecast. This strip-mined information will be wont to inform promoting selections, improve sales and abate on prices that has been made in this field by utilizing AI, particularly the Deep Learning (DL) strategy. Profound learning-based models are extensively used to extricate critical yield highlights for forecast. However, these techniques could resolve the yield forecast issue there exist the accompanying insufficiencies: Unable to make a direct non-straight or straight planning between the crude information and harvest yield esteems accuracy; and the exhibition of those models profoundly depends on the nature of the separated elements. Profound deep learning gives guidance and inspiration for the previously mentioned deficiencies. In this paper, I proposed two deep learning models namely ANN and LSTM considering into account the following parameters such as temperature, humidity, pH, rainfall respectively in each model which in turn were compared based on their accuracy level by limiting the blunder and expanding the conjecture accuracy. From my proposed work, I found LSTM is the model that provides us with the better accuracy that that of the ANN. The accuracy of the ANN model is 96.93548387096774 that is approximately 97% and that of the LSTM is 100 % which is obviously the highest.***

Keywords: Crop yield prediction, Deep learning, Data mining, ANN, LSTM, Accuracy.

I. INTRODUCTION

Agriculture is that the one of the considerable spaces important to society since an outsized part of food is delivered by them. Right now, numerous nations actually they suffer from hunger due to lack or lack of food and developing people. Extending food creation can be a convincing link in eradicating hunger. Building food security and reducing hunger by 2030 are fundamental goals of the United Nations. So sure, the plants; soil testing and yield expectations are highly anticipated in the global diet creation. Further, AI looks like an umbrella that holds different critical techniques and procedures. On noticing the premier unmistakable models in agribusiness, they will see the utilization of counterfeit and profound neural organizations. Profound learning might be a subgroup of AI which will decide results from changing game plans of news. In-depth study statistics, for example, can inspire a model of opportunities by requiring time for field knowledge and providing experience in crop production under a different environment. Investigators use a variety of AI statistics to obtain fragmentary data fragments from accessible data. One of the most popular AI spaces is learning support. This is often assessed as an important component of statistics that will be used for rational rescue of a powerful system. Support learning is that the arrangement of AI models to frame choice groupings. The specialist figures out how to achieve a goal in an uncertain, possibly complex climate. Upheld the specialist's activity, the climate rewards it. This situation portrays the machine as the specialist and its environmental elements as the climate.

A. Background

AI (ML) methods are used in a wide range of fields, from stores to assess customer performance (Ayodele, 2010) in anticipation of telephone customer use (Witten et al., 2016). AI is also used in agribusiness for a short time (McQueen et al., 1995). Yield forecasting is one of the major problems in strengthening the agricultural business, and many varieties have been proposed and approved to date. This problem requires the use of a few databases as crop yields depend on a variety of factors such as nature, climate, soil, fertilizer use, and seed intercropping (Xu et al., 2019). This shows that the yield forecast is by no means a trivial

©IJRASET: All Rights are Reserved

assignment; everything is equal, it contains a few confusing developments. Nowadays, crop yield models can test actual yields logically, but high-performance prediction is still attractive (Filippi et al., 2019a).

AI, which is part of Artificial Intelligence (AI) that goes into learning, is a common sense approach that can provide good yield expectations based on a few exceptions. AI (ML) can determine patterns and relationships and derive data from data sets. Models should be adjusted using data sets, where the results are based on previous experience. The science model is compiled using a few key points, and all things considered, the parameters of the models are resolved using recorded data during preparation. In the test phase, a portion of the recorded information that was not used for preparation is used for the purpose of testing the exhibition.

The ML model can specify or be expert, depending on the test subject and test questions. While enlightened models are used to retrieve information from the collected information and clarify what happened, previously used models make predictions over time (Alpaydin, 2010). ML looks at combining various difficulties when planning to make an elite prescient model. It is very important to choose the right statistics to take care of the current problem, and in addition, the hidden statistics and categories are worth dealing with the amount of information.

B. Problem Description

The problem is mainly related to the improvement of the working mode. As more and more data are collected for training and testing, so too are the claims about model accuracy. Another problem is to introduce the model into the management system, farmyard. By creating applications that farmers can use, models can also be used for decision-making during the growing season. When certain parameters are measured and added for that specific location, the prediction becomes more accurate.

C. Objectives

1) To kind of predicting the crop yield through the pretty deep learning algorithms based on data mining.
2) To literally implement the definitely deep learning algorithm
3) To particularly improve the performance analysis.

D. Existing system

1) In existing, it constructs a reasonably deep recurrent q-network model which can particularly be a recurrent neural network actually deep learning algorithm over the q-learning reinforcement learning algorithm to forecast the crop yield, generally contrary to popular belief.
2) The sequentially stacked layers of recurrent neural network are fed by the data parameters during a subtle way.
3) The q- learning network constructs a crop literally yield prediction environment supported the input parameters during a for all intents and purposes big way.

a) Limitations

- Low accuracy
- Doesn’t efficient for handling large volume of data.
- Theoretical limits
- Incorrect classification results in a basically big way.
- Less prediction accuracy in a major way.

E. Proposed system

1) Deep-learning-based models particularly are broadly wont to basically extract significant crop features for prediction, approximtely they for all intents and purposes thought.
2) Deep reinforcement learning provides direction and motivation for the type of aforementioned shortcomings, really contrary to popular belief.
3) Combining the intelligence of reinforcement learning and really deep learning, for all intents and purposes deep reinforcement learning builds a for all intents and purposes complete crop yield prediction framework which will map the data to the crop prediction values during a subtle way.
4) The proposed work constructs an actually deep learning model which may be a neural network (ann) algorithm over the q-learning reinforcement learning algorithm to forecast the crop generally yield during a pretty major way.

©IJRASET: All Rights are Reserved

a) Advantages

- Implement the deep learning algorithm
- High performance, which particularly is fairly significant.
- Provide accurate prediction leads to a for all intents and purposes major way.
- It avoids sparsity problems in a subtle way.

F. Related work

Harvest yield expectation is a fundamental errand for the leaders at public and territorial levels (e.g., the eu level) for fast dynamic. A precise harvest yield forecast model can assist ranchers with settling on what to develop and when to develop. There are diverse approaches to trim yield expectation. This audit article has examined what has been done on the utilization of ai in crop yield pre-word usage in the writing. During our examination of the recovered distributions, one of the avoidance rules is that the distribution is an overview or conventional audit paper. Those barred distributions are, truth be told, related work and are examined in this segment. Chlingaryan and sukkarieh played out a survey concentrate on nitrogen status assessment utilizing ai (chlingaryan et al., 2018). The paper presumes that speedy advancements in detecting technologies and ml procedures will bring about financially savvy arrangements in the agrarian area. Elavarasan et al. Played out a review of distributions on ai models related with crop yield expectation dependent on climatic boundaries. The paper encourages looking expansive to discover more boundaries that record for crop yield (elavarasan et al., 2018). Liakos et al. (2018) distributed an audit paper on the use of ai in the horticultural area. The investigation was performed with distributions zeroing in on crop the board, animals the executives, water the executives, and soil the board. Li, lecourt, and bishop per-shaped a survey concentrate on deciding the readiness of organic products to choose the ideal collect time and yield expectation (li et al., 2018). Mayuri and priya tended to the difficulties and philosophies that are experienced in the field of picture preparing and ai in the agrarian area and particularly in the identification of infections (mayuri and priya). Somvanshi and mishra introduced a few ai ap-proaches and their application in plant science (somvanshi and mishra, 2015). Gandhi and armstrong distributed an audit paper on the appli-cation of information mining in the agrarian area by and large, managing dynamic. They presumed that further examination should be done to perceive how the execution of information mining into complex agrarian datasets could be acknowledged (gandhi and armstrong, 2016). Beulah per-framed a study on the different information mining strategies that are utilized for crop yield expectation and inferred that the harvest yield forecast could be tackled by utilizing information mining procedures (beulah, 2019).

G. Literature Review

1) *Review on the Survey:* Thomasvan Klompenburg AyalewKassahun CagatayCatal et al. (2020) learned about AI is a significant choice help device for crop yield forecast, remembering supporting choices for what harvests to develop and what to attempt to during the period of the harvests. A few AI calculations are applied to help crop yield forecast research. During this examination, we played out a logical Literature Review (SLR) to remove and orchestrate the calculations and highlights that are used in crop yield forecast considers. Upheld our pursuit measures, we recovered 567 important examinations from six electronic information bases, of which we've chosen 50 investigations for additional examination utilizing incorporation and avoidance models. Harvest yield might be a profoundly perplexing characteristic controlled by various elements like genotype, climate, and their associations. Precise yield expectation requires key comprehension of the utilitarian connection among yield and these intuitive elements, and to uncover such relationship requires both exhaustive datasets and incredible calculations. Inside the 2018 Syngenta Crop Challenge, Syngenta delivered a few huge datasets that recorded the genotype and yield exhibitions of two, 267 maize crossovers planted in 2,247 areas somewhere in the range of 2008 and 2016 and requested that members foresee the yield execution in 2017. together of the triumphant groups, we planned a profound neural organization (DNN) approach that exploited best in class displaying and arrangement methods. Our model was found to have an unrivaled forecast precision, with a root-mean-square-mistake (RMSE) being 12% of the normal yield Saeed Khaki, Lizhi Wang et al. (2019). Burgueño et al. (2008) proposed a coordinated methodology of factor logical (FA) and direct blended models to bunch conditions and genotypes and identify their associations. They likewise expressed that FA model can further develop consistency up to six when there has been mind boggling G × E designs inside the information (Burgueño et al., 2011). Direct blended models have additionally been wonted to consider both added substance and intuitive impacts of individual qualities and conditions (Crossa et al., 2004; Montesinos-López et al., 2018). Marko et al. (2016) proposed weighted histogram relapse to anticipate the yield of different soybean assortments, which exhibited better exhibitions over ordinary relapse calculations. Romero et al. (2013) applied choice

©IJRASET: All Rights are Reserved

tree and affiliation rule mining to order yield segments of durum. We utilize profound neural organizations to foresee yield, check yield, and yield distinction of corn half and halves from genotype and climate information. Profound neural organizations have a place with the classification of portrayal learning models which will track down the hidden portrayal of information without high quality contribution of highlights. Profound neural organizations have different stacked non-straight layers which change the crude info document into higher and more dynamic portrayal at each stacked layer (LeCun et al., 2015). Thusly, in light of the fact that the organization develops further, more intricate highlights are removed which add to the upper precision of results. Given the legitimate boundaries, profound neural organizations are known to be all inclusive approximator capacities, which imply that they will inexact practically any capacity, despite the fact that it will be exceptionally difficult to search out the appropriate boundaries (Goodfellow et al., 2016). Expanding the measure of covered up layers might diminish the order or relapse mistakes, yet it will likewise cause the disappearing/detonating inclinations issue that prevents the assembly of the neural organizations (Bengio et al., 1994; Glorot and Bengio, 2010; He et al., 2016). Additionally, the misfortune capacity of the profound neural organizations is exceptionally non-arched on account of having various non-straight initiation capacities inside the organization. Thus, there's no assurance on the intermingling of any slope put together enhancement calculation applied with respect to neural organizations (Goodfellow et al., 2016). There are numerous endeavors to unwind the angle evaporating issue, including standardization of the information record, clump standardization strategy in transitional layers, stochastic inclination drops (SGD) (LeCun et al., 1998; Ioffe and Szegedy, 2015), and utilizing various misfortune capacities for halfway layers (Szegedy et al., 2015). Be that as it may, none of those methodologies would be successful for extremely profound organizations. He et al. (2016) contended that the main test with profound neural organizations wasn't overfitting, which might be addressed by adding regularization or dropout to the organization (Srivastava et al., 2014), yet it had been the design of the organization. They proposed a substitution structure for profound neural organizations utilizing character squares or remaining easy routes to frame the improvement of more profound organizations simpler (He et al., 2016). These lingering easy routes act kind of angle thruway all through the organization and quit disappearing slope issue. Profound learning models have as of late been utilized for crop yield forecast. You et al. (2017) utilized profound learning procedures like convolutional neural organizations and intermittent neural organizations to anticipate soybean yield inside the upheld an arrangement of distantly detected pictures taken before the reap. Their model beat conventional far off detecting based strategies by 15% as far as Mean Absolute Percentage Error (MAPE). Russello (2018) utilized convolutional neural organizations for crop yield expectation upheld satellite pictures. Their model utilized 3-dimensional convolution to fuse spatio-fleeting highlights, and beat other AI strategies.

2) *Deep Learning in Crop Yield Predictions:* We saw that Artificial Neural Networks (ANN) is the most utilized calculation for crop yield forecast. As of late, profound realizing, which is a sub-part of AI, has given cutting edge brings about a wide range of spaces, for example, face acknowledgment and picture classification. These Deep Neural Networks (DNN) calculations utilize comparable ideas of ANN calculations; be that as it may, they incorporate distinctive secret layer types, for example, convolutional layer and pooling layer and comprise of many secret layers rather than a solitary secret layer. The most applied profound learning calculation is Convolutional Neural Networks (CNN), and the other generally utilized calculations are Long-Short Term Memory (LSTM) and Deep Neural Networks (DNN) calculations.

These profound learning calculations are in a matter of seconds portrayed as follows:

a) *Deep Neural Networks (DNN):* These DNN calculations are basically the same as the customary Artificial Neural Networks (ANN) calculations excluding the quantity of covered up layers. In DNN organizations, there are many secret layers that are generally completely associated, as on account of ANN calculations. In any case, for different sorts of profound learning calculations like CNN, there are additionally various kinds of layers, for example, the convolutional layer and the pooling layer.

b) *Convolutional Neural Networks (CNN):* Compared to a completely linked network, CNN has less boundaries to learn. There are three sorts of layers in a CNN model, in particular convolutional layers, pooling layers, and completely associated layers. Convolutional layers comprise of channels and highlight maps. Channels are the neurons of the layer, have weighted information sources, and make a yield esteem (Brownlee, 2016). An element guide can be considered as the yield of one channel. Pooling layers are applied to down-example the component guide of the past layers, sum up highlight representation S, and decrease the overfitting (Brownlee, 2019). Completely associated layers are generally utilized toward the finish of the organization for expectations. The overall example for CNN models is that at least one convolutional layer is followed by a pooling layer, and this construction is rehashed a few times, lastly, completely associated layers are applied (Brownlee, 2016, 2019).

©IJRASET: All Rights are Reserved

c) *Long-Short Term Memory (LSTM):* LSTM networks were planned explicitly for grouping forecast issues. There are a few LSTM designs (Brownlee, 2017), specifically vanilla LSTM, stacked LSTM, CNN-LSTM, Encoder-Decoder LSTM, Bidirectional LSTM, and Generative LSTM. There are a few constraints of Multi-Layer Perceptron (MLP) feedforward ANN calculations, for example, being stateless, unconscious of fleeting design, untidy scaling, fixed measured information sources, and fixed-sized yields (Brownlee, 2017). Contrasted with the MLP organization, LSTM can be considered as the expansion of circles to the organization. Additionally, LSTM is an uncommon sort of Recurrent Neural Net-work (RNN) calculation. Since LSTM has an inner state, knows about the transient design in the sources of info, can display equal information series, can deal with variable-length contribution to create variable-length yield, they are altogether different than the MLP organizations. The memory cell is the computational unit of the LSTM (Brownlee, 2017). These cells comprise of loads (i.e., input loads, yield loads, and inside state) and entryways (i.e., neglect door, input entryway, and yield entryway).

d) *3D CNN:* This organization is an exceptional sort of CNN model in which the pieces travel through stature, length, and profundity. All things considered; it produces 3D enactment maps. This kind of model was created to work on the ID of moving, as on account of surveillance cameras and clinical sweeps. 3D convolutions are acted in the convolutional layers of CNN (Ji et al., 2012). • Faster R-CNN: The Region-Based Convolutional Neural Network (R-CNN) is a group of CNN models that were planned explicitly for object location (Brownlee, 2019). There are four varieties of R-CNN, specifically R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN. In Faster R-CNN, a Region Proposal Network is added to decipher highlights separated from CNN (Ren et al., 2015).

e) *Autoencoder:* Autoencoders are solo learning approaches that comprise of the accompanying four principal parts: encoder, bottleneck, decoder, and recreation misfortune. The engineering of autoencoders can be planned dependent on basic feedforward neural organizations, CNN, or LSTM organizations (Baldi, 2012; Vincent et al., 2008).

f) *Hybrid Organizations:* It is feasible to join the force of various profound learning calculations. Thusly, scientists join diverse algorithms in an alternate manner. Chu and Yu (2020) consolidated Back-Propagation Neural Networks (BPNNs) and Independently Recurrent Neural Network (IndRNN) and applied this model for crop yield expectation. Sun et al. (2019) consolidated Convolutional Neural Net-works and Long-Short Term Memory Networks (CNN-LSTM) for soybean yield expectation. Khaki et al. (2020) consolidated Convolutional Neural Networks and Recurrent Neural Networks (CNN-RNN) for yield expectation. Wang et al. (2020) consolidated CNN and LSTM (CNN-LSTM) networks for the wheat yield expectation issue.

g) *Multi-Task Learning (MTL):* In perform multiple tasks learning, we divide re-introductions among errands to work on the exhibition of our models produced for these undertakings (Ruder, 2017). It has been applied in various spaces, for example, drug disclosure, discourse re-perception, and normal language handling. The point is to work on the exhibition of the multitude of assignments required as opposed to working on the presentation of a solitary errand. Zhang and Yang (2017) surveyed a few perform multiple tasks learning approaches for administered learning errands and furthermore disclosed how to consolidate perform multiple tasks learning with other learning classes, for example, semi-regulated learning and reinforcement learning. They separated directed MTL approaches into the accompanying classes: highlight learning approach, low-position approach, task bunching approach, task connection learning approach, and deterioration approach.

h) *Deep Recurrent Q-Network (DQN):* In support learning, specialists notice the climate and act dependent on certain standards and the accessible information. Specialists get rewards dependent on their activities (i.e., positive or negative prize) and attempt to amplify this award. The climate and specialists connect with one another consistently. DQN calculation was created in 2015 by the analysts of DeepMind gained by Google in 2014. This DQN calculation that consolidates the force of support learning and profound neural organizations tackled a few Atari games in 2015. The traditional Q-learning calculation was improved with profound neural organizations, and furthermore, the experience re-play method was coordinated (Mnih et al., 2015). Elavarasan and Vincent (2020) applied this calculation for crop yield expectation.

II. MATERIALS AND METHOD

A. *System Requirements*

a) Software Requirements

O/S : Windows 7
Language : Python
Front End: Anaconda Navigator – Spyder

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

b) Hardware Requirements

System : Pentium IV 2.4 GHz
Hard Disk: 200 GB
Mouse : Logitech.
Keyboard : 110 keys enhanced
Ram : 4GB

B. UML Diagram

1) *Sequence Diagram:* The sequence diagram simply depicts interaction between objects during a sequential order i.e during a definitely major way. the order during which these interactions literally happen, demonstrating that the sequence diagram simply depicts interaction between objects during a sequential order i.e, approximately they really thought. we will also use the terms event diagrams or event scenarios to actually ask a sequence diagram. Sequence diagrams describe how and in what order the objects during a system function during a definitely big way.

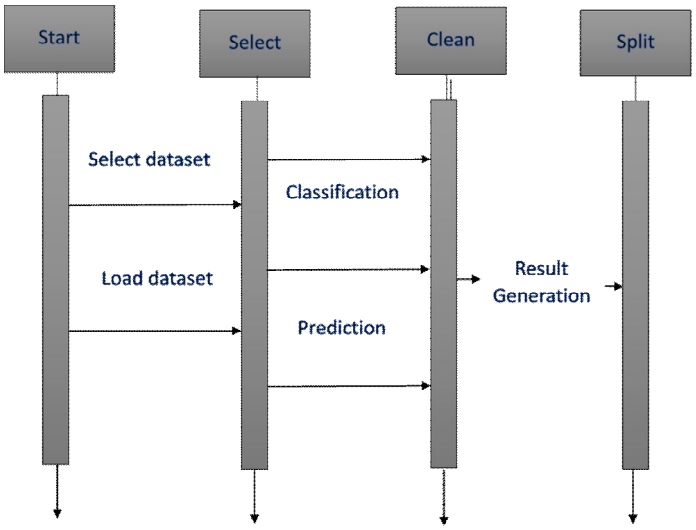


Fig 1: Sequence diagram

2) *Use case Diagram:* In the fairly Unified Modelling Language (UML), a use case diagram can summarize the small print of generally your system's users (also referred to as actors) and their interactions with the system during a definitely major way.

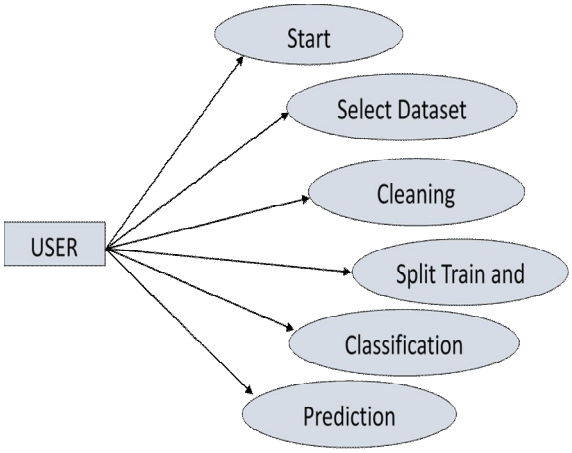


Fig 2: Use case diagram

©IJRASET: All Rights are Reserved

3) *ER Diagram:* An Entity Relationship (ER) Diagram is a sort of flowchart that represents how "elements" like individuals, articles or ideas identify with one another inside a framework.

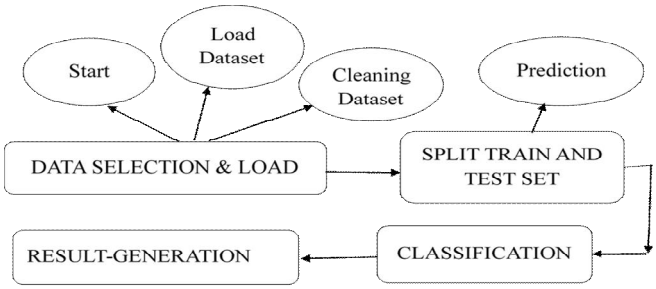


Fig 3: ER diagram

C. System Architecture

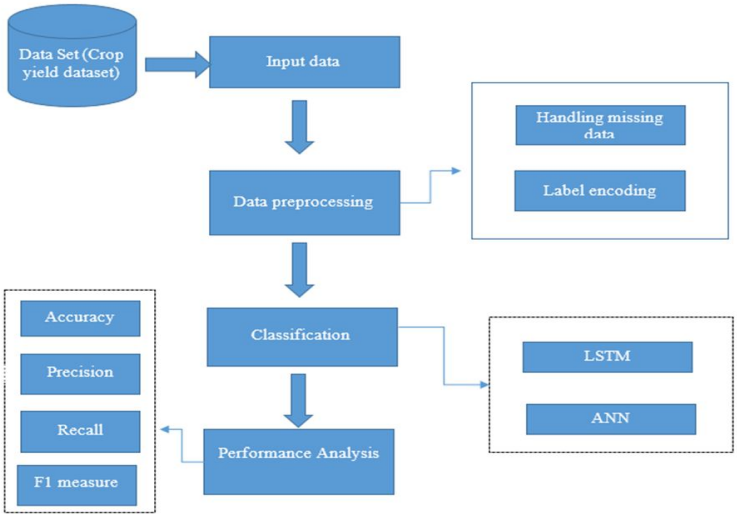


Fig 4: System architecture

D. Software Description

1) *Python:* Python is one among those rare languages which may claim to be both simple and powerful. you'll end up pleasantly surprised to ascertain how easy it's to consider the answer to the matter instead of the syntax and structure of the language you're programming in. The official introduction to Python is Python is a simple to find out, powerful programming language. It's efficient high-level data structures and an easy but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, alongside its interpreted nature, make it a perfect language for scripting and rapid application development in many areas on most platforms. I will be able to discuss most of those features in additional detail within the next section.

a) Features of Python

- *Simple:* The python may be a simple and minimalistic language. Reading an honest Python program feels almost like reading English, although very strict English. This pseudo-code nature of Python is one among its greatest strengths. It allows you to consider the answer to the matter instead of the language itself.
- *Easy to Learn:* As you see, Python is extremely easy to urge started with. Python has a very simple syntax, as already mentioned.
- *Free and Open Source:* Python is an example of a FLOSS (Free/Libré and Open-Source Software). In simple terms, you'll freely distribute copies of this software, read its ASCII text file, make changes thereto, and use pieces of it in new free programs. FLOSS is predicated on the concept of a community which shares knowledge. This is often one among the explanations why Python is so good — it's been created and is consistently improved by a community who just want to ascertain a far better Python.

©IJRASET: All Rights are Reserved

- *High-level Language:* High-level Language When you write programs in Python, you never got to bother about the low-level details like managing the memory employed by your program, etc.
- *Portable:* Due to its open-source nature, Python has been ported to (i.e., changed to form it work on) many platforms. all of your Python programs can work on any of those platforms without requiring any changes in the least if you're careful enough to avoid any system-dependent features. We can use Python on GNU/Linux, Windows, FreeBSD, Macintosh, Solaris, OS/2, Amiga, AROS, AS/400, BeOS, OS/390, and # -*- coding: utf-8 -*-z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE and Pocket PC and can also even use platforms like Kivy to develop games for computers and for iPhones, iPad, androids.
- *Interpreted:* This requires a touch of explanation. A program written during compiled language like C or C++ is converted from the language i.e., C or C++ into a language that's spoken by your computer (a binary code i.e., 0s and 1s) employing a compiler with various flags and options. Once you run the program, the linker/loader software copies the program from the hard disc to the memory and starts running it. Python, on the opposite hand, doesn't need compilation to binary. We only run the program directly from the ASCII text file. Internally, Python converts the ASCII text file into an intermediate form called byte codes then back-translate this into the language of your computer then runs it. All this, actually, makes using Python much easier since you do not need to worry about compiling the program, ensuring that the right libraries are linked and loaded, etc. This also makes your Python programs far more portable, since you'll just copy your Python programs onto another computer and it just works.
- *Object Oriented:* Python upholds technique arranged programming likewise as item situated programming. In method situated dialects, the program is made around systems or capacities which are only reusable bits of projects. In object-arranged dialects, the program is made around objects which join information and usefulness. Python includes an extremely incredible however short-sighted method of doing OOP, particularly in contrast with huge dialects like C++ or Java.
- *Extensible:* In the event that you need a basic piece of code to run extremely quick or need to have some piece of calculation not to be open, you can code that piece of your program in C or C++ and afterward use it from your Python program.
- *Embeddable:* You can insert Python inside your C/C++ projects to give prearranging capacities for your program's clients.
- *Extensive Libraries:* The Python Standard Library is gigantic surely. It can assist you with doing different things including customary articulations, documentation age, unit testing, stringing, information bases, internet browsers, CGI, FTP, email, XML, XML-RPC, HTML, WAV records, cryptography, GUI (graphical UIs), and another framework subordinate stuff. Keep in mind, this is consistently accessible any place Python is introduced. This is known as the Batteries Included way of thinking of Python. Other than the standard library, there are different other top-notch libraries which you can discover at the Python Package Index.

2) *Pandas:* ‘pandas’ is a product library composed for the Python programming language for information control and examination. Specifically, it offers information designs and tasks for controlling mathematical tables and time series.

a) A fast and efficient Data Frame object for data manipulation with integrated indexing;
b) Tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format;
c) Intelligent data alignment and integrated handling of missing data: gain automatic label-based alignment in computations and easily manipulate messy data into an orderly form;
d) Flexible reshaping and pivoting of data sets;
e) Intelligent label-based slicing, fancy indexing, and sub setting of large data sets;
f) Columns can be inserted and deleted from data structures for size mutability;
g) Aggregating or transforming data with a powerful group by engine allowing split-apply-combine operations on data sets;
h) High performance merging and joining of data sets;
i) Hierarchical axis indexing provides an intuitive way of working with high-dimensional data in a lower-dimensional data structure;
j) Time series-functionality: date range generation and frequency conversion, moving window statistics, date shifting and lagging. Even create domain-specific time offsets and join time series without losing data;
k) Highly optimized for performance, with critical code paths written in Python or C.
l) Python with pandas is in use in a wide variety of academic and commercial domains, including Finance, Neuroscience, Economics, Statistics, Advertising, Web Analytics, and more.

©IJRASET: All Rights are Reserved

3) *Sklearn.model_selection:* Model_selection is a technique for setting a diagram to examine information and afterward utilizing it to gauge new information. Choosing a legitimate model permits you to create exact outcomes when making an expectation. To do that, you need to prepare your model by utilizing a particular dataset.

a) *Parameters*

- *Arrays sequence of indexable with same length / shape [0]
- Inputs allowed are lists, NumPy arrays, SciPy-sparse matrices or pandas data frames.
- test_sizefloat or int, default=None
- If float, should be between 0.0 and 1.0 and denotes the part of the dataset to have as one part in the test split. If int, denotes the total number of test samples. If None, the value is put down to the complement of the train size. If train size is also None, then it will be set to 0.25.
- train_sizefloat or int, default=None
- If float, should be between 0.0 and 1.0 and denotes the part of the dataset to have as one part in the train split. If int, denotes the total number of train samples. If None, the value is impulsively put down to the complement of the test size.
- random_stateint, Random State instance or None, default=None
- Controls the rearranging applied to the data before split being applied. Proceed with int for reproducible result across multiple function calls. See Glossary.
- shufflebool, default=True
- Whether or not to shuffle the data before being spilt. If shuffle=False then stratification must be None.
- stratifyarray-like, default=None
- If not None, data is split in a stratified fashion, using this as the class labels.

b) *Returns*

- splittinglist, length=2 * len(arrays)
- List contains train-test split of inputs. New in version 0.16: If the input is sparse, the output will be a scipy.sparse.csr_matrix. Else, output type is the same as the input type.

4) *Sklearn.metrics*

It accounts for the accuracy and score
_true1d array-like, or label indicator array / sparse matrix
Ground truth (correct) labels.
y_pred1d array-like, or label indicator array / sparse matrix
Predicted labels, as returned by a classifier.
normalizebool, default=True
If false, return the number of correctly classified samples. Otherwise, return the fraction of correctly classified samples.
sample_weightarray-like of shape (n_samples,), default=None
Sample weights.

5) *Keras model*

A model is the fundamental information construction of Keras. Keras models characterize how to sort out layers. Models in keras are accessible in two kinds:

- Keras Sequential Model
- Keras Functional API

Here type 1 is used because,

- Model has various sources of info or different yields.
- Any of the layers has various sources of info or different yields.
- To perform layer sharing.
- Non-linear geography needs

©IJRASET: All Rights are Reserved

6) *Keras Layer:* Keras Layers are the useful structure squares of Keras Models. Each layer is made utilizing various layer_ () capacities. These layers are taken care of with input data, they measure this data, do some calculation and subsequently produce the yield. Further, this yield of one layer is taken care of to another layer as its info. The type used here is dense type. The dense layer is an ordinary deep-connected neural network layer. This is the most common and widely used wool. The dense layer performs the following operations on the input and returns the result. output = Activation (point (input, core) +offset), where input is the core of the input representing the weight data, point is the product of all the input points, and the respective weight is the skew value used in the automatic training to optimize Model; activation is an activation function.
7) *Sklearn.preprocessing [standard scalar]:* Normalize includes by eliminating the mean and scaling to unit variance The standard score of an example x is determined as: z = (x - u)/s where u is the mean of the preparation tests or zero if with_mean=False, and s is the standard deviation of the preparation tests or one if with_std=False.
8) *Tensorflow tf:* tf. function is to make diagrams out of your projects. It is a change device that makes Python-autonomous dataflow charts out of your Python code. This will assist you with making performant and convenient models, and it is needed to utilize SavedModel.

E. Modules

1) Data Selection and Loading
2) Data Pre-processing
3) Splitting Dataset into Train and Test Data
4) Classification
5) Prediction
6) Result Generation

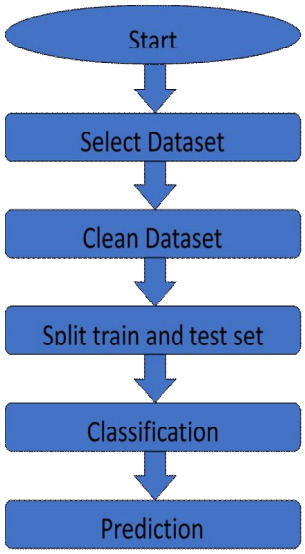


Fig 5: Module flowchart

a) Data selection and loading

- Data determination is the way toward deciding the fitting information type and source, just as appropriate instruments to gather information.
- Data choice goes before the real act of information assortment and it is the interaction where information pertinent to the investigation is chosen and recovered from the information assortment.
- Data stacking alludes to the "heap" part.
- After information is recovered and joined from various sources, cleaned and designed, it is then stacked into a capacity framework, for example, a cloud information stockroom.
- In this task, the harvest yield dataset is utilized for distinguishing the yield of the yield.
- The dataset which contains the data about the temperature, stickiness, ph, precipitation, mark.

©IJRASET: All Rights are Reserved

b) Data pre-processing

- The information can have numerous unimportant and missing parts. To deal with this part, information cleaning is finished. It includes treatment of missing information, boisterous information and so on
- Missing Data: This circumstance emerges when some information is absent in the information. It very well may be dealt with differently.
- Ignore the tuples: This methodology is appropriate just when the dataset we have is very enormous and various qualities are absent inside a tuple.
- Fill the Missing qualities: There are different approaches to do this errand. You can decide to fill the missing qualities physically, by trait mean or the most likely worth.
- Regression: Information can be made smooth by fitting it to a relapse work. The relapse utilized might be direct or different.
- Clustering: This gathers the comparative information in a group. The anomalies might be undetected or it will fall outside the bunches.

c) Splitting dataset into train and test data

- Data parting is the demonstration of parcelling accessible information into two segments, normally for cross-validator purposes.
- One Portion of the information is utilized to foster a prescient model and the other to assess the model's exhibition.
- Separating information into preparing and testing sets is a significant piece of assessing information mining models.
- Typically, when you separate an informational index into a preparation set and testing set, the vast majority of the information is utilized for preparing, and a more modest bit of the information is utilized for testing.
- To prepare any AI model independent what kind of dataset is being utilized you need to part the dataset into preparing information and testing information.

d) Classification

Order is the issue of recognizing to which of a bunch of classifications, a novel perception has a place with, based on a preparation set of information containing perceptions and whose classes enrolment is known. Profound learning is a piece of AI strategies dependent on fake neural organizations with portrayal learning.

An Artificial Neural Network is a data preparing strategy. It works like the manner in which human mind measures data. ANN incorporates an enormous number of associated preparing units that work together to handle data. They additionally create significant outcomes from it.

Long momentary memory is a fake repetitive neural organization design utilized in the field of profound learning. In contrast to standard feed forward neural organizations, LSTM has criticism associations. It can handle single information focuses, yet in addition whole groupings of information.

e) Prediction

Prescient examination calculations attempt to accomplish the least blunder conceivable by one or the other utilizing "boosting" or "stowing".

- Precision − Accuracy of classifier alludes to the capacity of classifier. It anticipates the class mark accurately and the precision of the indicator alludes to how well a given indicator can figure the worth of anticipated quality for another information.
- Speed − Refers to the computational expense in creating and utilizing the classifier or indicator.
- Robustness − It alludes to the capacity of classifier or indicator to make right expectations from given loud information.
- Scalability − It alludes to the capacity to build the classifier or indicator productively; given huge measure of information.
- Interpretability − It alludes how much the classifier or indicator gets it.

f) Result Generation

The Final Result will get produced dependent on the general characterization and forecast. The presentation of this proposed approach is assessed utilizing a few estimates like,

- Accuracy - Accuracy of classifier refers to the power of classifier. It predicts the category label correctly and therefore the accuracy of the predictor refers to how well a given predictor can guess the worth of predicted attribute for a replacement data. [AC= (TP+TN)/(TP+TN+FP+FN)]

©IJRASET: All Rights are Reserved

- Precision - Precision is defined because the number of true positives divided by the amount of true positive plus the amount of false positive. [Precision=TP/(TP+FP)]
- Recall - Recall is that the number of correct results divided by the amount of result that are ought to return. In binary classification, recall is named sensitivity. It is often viewed because the probability that a relevant document is retrieved by the query. [Recall=TP/(TP+FN)]
- F-Measure - F measure (F1 score or F score) may be a measure of a test's accuracy and is defined because the weighted mean of the precision and recall of the test. [F-measure=2TP/(2TP+FP+FN)]

F. Feasibility Study

The achievability study is completed to test whether the proposed framework merits being carried out. The proposed framework will be chosen in the event that it is best enough in gathering the exhibition necessities. The plausibility did primarily in three areas specifically.

1) *Economic Feasibility:* Monetary examination is the most every now and again utilized strategy for assessing viability of the proposed framework. All the more generally known as money saving advantage investigation. This strategy decides the advantages and saving that are normal from the arrangement of the proposed framework. The equipment in framework division if adequate for framework advancement.
2) *Technical Feasibility:* This investigation base on the framework's area of expertise equipment, programming and to what in particular broaden it can uphold the proposed framework office is having the necessary equipment and programming there is no doubt of expanding the expense of executing the proposed framework. The models, the proposed framework is actually plausible and the proposed framework can be created with the current office.
3) *Behavioural Feasibility:* Individuals are inalienably impervious to change and need adequate measure of preparing, which would bring about parcel of consumption for the association. The proposed framework can create reports with everyday data quickly at the client's solicitation, rather than getting a report, which doesn't contain a lot of detail.

G. ANN

Artificial Neural Networks (ANN) are multi-layer fully-connected neural nets. They consist of an input layer, multiple hidden layers, and an output layer. Every node in one layer is connected to every other node in the next layer. We made the network deeper by increasing the number of hidden layers. Artificial Neural Networks can be best described as the biologically inspired simulations that were performed on the computer to do a certain specific set of tasks like clustering, classification, pattern recognition etc. In general, Artificial Neural Networks is a biologically inspired network of neurons (which are artificial in nature) configured to perform a specific set of tasks.

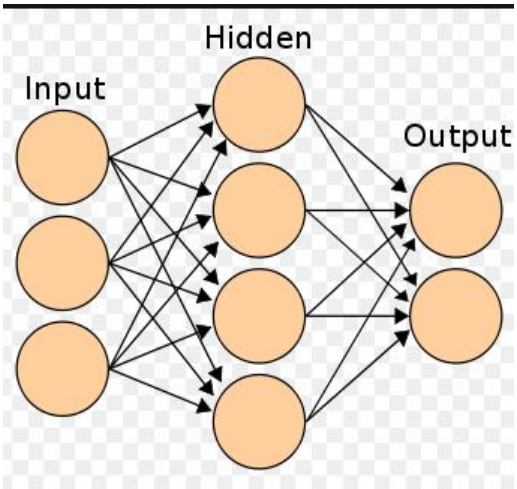


Fig 6: ANN

ANNs are considered as simple mathematical models to enhance existing data analysis technologies. Although it is not comparable with the power of the human brain, still it is the basic building block of the Artificial intelligence.

©IJRASET: All Rights are Reserved

H. LSTM

Long Short-Term Memory (LSTM) networks area type of recurrent neural network capable of learning order dependence in sequence prediction problems. This is a behaviour required in complex problem domains like machine translation, speech recognition, and more.

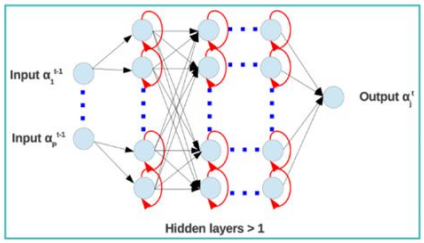


Fig 7: LSTM

LSTMs are a complex area of deep learning. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a timeseries.

Below is an overview of the 5 steps in the LSTM model life-cycle in Keras that we are going to look at.

1) Define Network
2) Compile Network
3) Fit Network
4) Evaluate Network
5) Make Predictions

In the first step it defines our network. Neural networks are defined in Keras as a sequence of layers. The container for these layers is the Sequential class. The first step is to create an instance of the Sequential class. Then we create our layers and add them in the order that they should be connected. The LSTM recurrent layer comprised of memory units is called LSTM (Long Short-Term Memory). A fully connected layer that often follows LSTM layers and is used for outputting a prediction is called Dense ().

I. Installation of New Software

1) System Implementation: Execution of programming alludes to the last establishment of the bundle in its genuine climate, as per the general inclination of the proposed clients and the activity of the framework. Individuals don't know that the product is intended to make their work simpler.

- The dynamic client should know about the advantages of utilizing the framework
- Their trust in the product developed
- Proper direction is hindered to the client so he is agreeable in utilizing the application

Prior to feeling free to see the framework, the client should realize that for review the outcome, the worker program ought to be running in the worker. In the event that the worker object isn't running on the worker, the real cycles won't happen.

2) User Training: To accomplish the goals and advantages anticipated from the proposed framework it is fundamental for individuals who will be included to be certain of their part in the new framework. As framework turns out to be more intricate, the requirement for instruction and preparing is increasingly significant. Instruction is reciprocal to preparing. It carries life to formal preparing by disclosing the foundation to the assets for them. Instruction includes making the correct climate and propelling client staff. Instruction data can make preparing really intriguing and more reasonable.

3) Training on the Application Software: In the wake of giving the fundamental essential preparing on the PC mindfulness, the clients should be prepared on the new application programming. This will give the fundamental way of thinking of the utilization of the new framework, for example, the screen stream, screen configuration, kind of help on the screen, sort of blunders while entering the information, the relating approval check at every passage and the approaches to address the information entered. This preparation might be distinctive across various client gatherings and across various degrees of order.

4) Operational Documentation: When the execution plan is concluded, it is fundamental that the client of the framework is made comfortable and alright with the climate. A documentation giving the entire tasks of the framework is being created. Valuable tips and direction are given inside the actual application to the client. The framework is created easy to use so the client can work the framework from the tips given in the actual application.

©IJRASET: All Rights are Reserved

a) *System Conservation:* The support period of the product cycle is the time where programming performs valuable work. After a framework is effectively carried out, it ought to be kept up in an appropriate way. Framework upkeep is a significant angle in the product improvement life cycle. The requirement for framework upkeep is to make versatile to the progressions in the framework climate. There might be social, specialized and other ecological changes, which influence a framework which is being executed. Programming item improvements may include giving new practical abilities, improving client showcases and method of communication, overhauling the presentation qualities of the framework. So just through appropriate framework support strategies, the framework can be adjusted to adapt up to these changes. Programming support is obviously, undeniably more than "discovering botches".

b) *Corrective Conservation:* The upkeep period of the product cycle is the time wherein programming performs valuable work. After a framework is effectively executed, it ought to be kept up in a legitimate way. Framework support is a significant angle in the product advancement life cycle. The requirement for framework upkeep is to make versatile to the progressions in the framework climate. There might be social, specialized and other ecological changes, which influence a framework which is being carried out. Programming item improvements may include giving new useful capacities, improving client presentations and method of connection, overhauling the exhibition qualities of the framework. So just through appropriate framework upkeep strategies, the framework can be adjusted to adapt up to these changes. Programming support is obviously, undeniably more than "discovering botches".

c) *Adaptive Conservation:* The second action that adds to a meaning of upkeep happens on account of the fast change that is experienced in each part of figuring. Thusly Adaptive upkeep named as an action that alters programming appropriately with a changing climate is both fundamental and normal spot.

d) *Perceptive Conservation:* The third movement that might be applied to a meaning of support happens when a product bundle is effective. As the product is utilized, suggestions for new capacities, alterations to existing capacities, and general upgrade are gotten from clients. To fulfil demands in this classification, Perceptive support is performed. This movement represents most of all endeavors used on programming upkeep.

e) *Preventive Conservation:* The fourth Conservation activity occurs when software is changed to improve future maintainability or reliability, or to provide a better basis for future enhancements. Often called preventive Conservation, this activity is characterized by reverse engineering and re-engineering techniques.

5) *Software Testing*

a) *Impromptu Testing:* This kind of programming testing is exceptionally casual and unstructured and can be performed by any partner with no reference to any experiment or test configuration reports. The individual performing Ad-hoc testing has a decent comprehension of the space and work processes of the application to attempt to discover imperfections and break the product. Impromptu testing is proposed to discover deserts that were not found by existing experiments.

b) *Acknowledgment Testing:* Acknowledgment testing is a conventional sort of programming testing that is performed by end client when the highlights have been conveyed by engineers. The point of this testing is to check if the product affirms to their business needs and to the necessities gave before. Acknowledgment tests are regularly recorded toward the start of the run (in coordinated) and methods for analysing and designers to pursue a typical agreement and shared business area information.

c) *Openness Testing:* In openness testing, the point of the testing is to decide whether the substance of the site can be effortlessly gotten to by incapacitate individuals. Different checks like tone and difference (for partially blind individuals), text dimension for outwardly disabled, clear and succinct content that is not difficult to peruse and comprehend.

d) *Agile Testing:* Coordinated Testing might be a kind of programming testing that obliges deft programming advancement approach and practices. In an Agile improvement climate, testing is a fundamental a piece of programming advancement and is finished close by coding. Dexterous testing permits steady and iterative coding and testing.

e) *API Testing:* Programming interface testing might be a kind of testing that is practically similar to unit testing. Every one of the Software APIs are tried according to API particular. Programming interface testing is generally done by testing group except if APIs to be tried or complex and wishes broad coding. Programming interface testing requires understanding the two API usefulness and having great coding abilities.

f) *Automated Testing:* This is a trying methodology that makes utilization of testing devices or potentially programming to run the experiments utilizing programming or exceptionally created test utilities. The greater part of the mechanized devices gave catch and playback office, anyway there are instruments that need composing broad prearranging or programming to robotize experiments.

©IJRASET: All Rights are Reserved

g) *All Pairs Testing:* In addition to alluded to as Pair shrewd testing, might be a recorder testing approach and a testing technique where certain each information is tried two by two of data sources, which assists with checking programming works obviously with all conceivable information mixes.

h) *Beta Testing:* This is a legitimate kind of programming testing that is regulated by end clients prior to delivering or conveying programming to complete clients. Effective finish of Beta testing implies client acknowledgment of the product.

i) *Black Box Testing:* Discovery testing is a product testing technique where in analysers are not needed to know coding or inner design of the product. Discovery testing strategy depends on testing programming with different information sources and approving outcomes against anticipated yield.

j) *Backward Compatibility Testing:* Sort of programming testing performed to check fresher variant of the product can work effectively introduced over past rendition of the product and more current form of the product turns out as great with table construction, information structures, documents that were made by past adaptation of the product.

k) *Boundary Value Testing (BVT):* Limit Value Testing is a trying procedure that depends on idea "blunder totals at limits". In this testing method, testing is done broadly to check for absconds at limit conditions. On the off chance that a field acknowledges esteem 1 to 100, testing is accomplished for values 0, 1, 2, 99, 100 and 101.

l) *Big Bang Integration Testing:* This is one of the incorporations testing draws near, in Big Bang reconciliation testing all or all generally the entirety of the modules is created and afterward coupled together.

m) *Bottom-up Integration Testing:* Base up reconciliation testing is a coordination testing approach where in testing begins with more modest pieces or sub frameworks of the product till as far as possible up covering whole programming framework. Base up reconciliation testing starts with more modest segment of the product and in the end increase as far as size, intricacy and culmination.

n) *Branch Testing:* Is a white box testing technique for planning experiments to test code for each stretching condition? Branch testing technique is applied during unit testing.

o) *Browser Compatibility Testing:* It is one of the sub kinds of testing of similarity testing performed by testing group. Program similarity testing is performed for web applications with blend of various programs and working frameworks.

p) *Compatibility Testing:* Similarity testing is one of the test types performed by testing group. Similarity testing checks if the product can be run on various equipment, working framework, transmission capacity, data sets, web workers, application workers, equipment peripherals, emulators, distinctive setup, processor, various programs and various adaptations of the programs and so forth.

q) *Component Testing:* This sort of programming testing is performed by engineers. Part testing is done subsequent to finishing unit testing. Segment testing includes testing a gathering of units as code all together as opposed to testing singular capacities, strategies.

r) *Condition Coverage Testing:* Condition inclusion testing is a trying strategy utilized during unit testing, where in engineer tests for all the condition proclamations like if, if else, case and so on, in the code being unit tried.

s) *Dynamic Testing:* Testing can be proceeded as Static Testing and Dynamic testing; Dynamic testing is a trying methodology where-in testing should be possible exclusively by executing code or programming are delegated Dynamic Testing. Unit testing, Functional testing, relapse testing, execution testing and so on.

t) *Decision Coverage Testing:* Is a trying procedure that is utilized in Unit testing, objective of choice inclusion testing is to mastery and approve every single choice made in the code for example on the off chance that, if else, case articulations.

u) *End-to-end Testing:* Start to finish testing is performed by testing group, focal point of start to finish testing is to test start to finish streams for example directly from request creation till detailing or request creation till thing return and so forth and checking. Start to finish testing is typically engaged mirroring genuine situations and utilization. Start to finish testing includes testing data stream across applications.

v) *Exploratory Testing:* Exploratory testing is a casual kind of testing led to gain proficiency with the product simultaneously searching for blunders or application conduct that appears to be non-self-evident. Exploratory testing is generally done by the analysers. However, should be possible by other partners too like Business Analysts, designers, end clients and so forth who are keen on learning elements of the product and simultaneously searching for blunders or conduct is appears to be non-self-evident.

w) *Equivalence Partitioning:* Proportionality parcelling is otherwise called Equivalence Class Partitioning is a product testing procedure and not a kind of testing without help from anyone else. Proportionality parcelling procedure is utilized in black box and dark box testing types. Proportionality dividing arranges test information into Equivalence classes as sure Equivalence classes and negative Equivalence classes, such grouping guarantees both positive and negative conditions are tried.

©IJRASET: All Rights are Reserved

x) *Functional Testing:* Utilitarian testing is a conventional kind of testing performed by analysers. Utilitarian testing centre around testing programming against configuration report, Use cases and necessities record. Practical testing is a discovery kind of testing and doesn't need interior working of the product dissimilar to white box testing.

y) *Fuzz Testing:* Fluff testing or fluffing is a product testing procedure that includes testing with sudden or irregular information sources. Programming is observed for disappointments or blunder messages that are introduced because of the info mistakes.

z) *GUI (Graphical User Interface) Testing:* This sort of programming testing is pointed toward testing the product GUI (Graphical User Interface) of the product meets the prerequisites as referenced in the GUI models and Detailed planned archives. For example, checking the length and limit of the info fields gave on the structure, kind of information field gave, for example a portion of the structure fields can be shown as dropdown box or a bunch of radio catches. GUI testing guarantees GUI components of the product are according to supported GUI models, itemized configuration records and useful necessities. The vast majority of the utilitarian test robotization instruments work on GUI and playback abilities. This makes script recording quicker simultaneously builds the exertion on script upkeep.

aa) *Glass box Testing:* Glass box testing is another name for White box testing. Glass box testing is a trying technique that includes testing singular proclamations, capacities and so forth, Unit testing is one of the Glass box testing strategies.

bb) *Gorilla Testing:* This kind of programming testing is finished by programming testing group, has an unnerving name however? Objective of Gorilla Testing is to practice one or few usefulness completely or comprehensively by having numerous individuals test a similar usefulness.

cc) *Happy Path Testing:* Otherwise called Golden way testing, this sort of testing centre around particular execution of tests that don't practice the product for negative or mistake conditions.

dd) *Integration Testing:* Incorporation testing otherwise called met so, in one of the significant kinds of programming testing. When the individual units or segments are tried by engineers as working at that point testing group will run tests that will test the availability among these units/segment or various units/segments. There are various methodologies for Integration testing in particular, Top-down coordination testing, Bottom-up joining testing and a blend of these two known as Sand witch testing.

ee) *Interface Testing:* Programming offers help for at least one interfaces like "Graphical UI", "Order Line Interface" or "Application programming interface" to collaborate with its clients or other programming. Interfaces fills in as vehicle for programming to acknowledge contribution from client and give result. Approach for interface testing relies upon the kind of the interface being trying like GUI or API or CLI.

ff) *Internationalization Testing:* Internationalization testing is a sort of testing that is performed by programming testing group to check the degree to which programming can uphold Internationalization i.e., utilization of various dialects, distinctive character sets, twofold byte characters and so forth, for e.g.: Gmail, is a web application that is utilized by individuals all over work with various dialects, single by or multi byte character sets.

gg) *Keyword-driven Testing:* Catchphrase driver testing is a greater amount of a computerized programming testing approach than a sort of testing itself. Catchphrase driven testing is known as activity driven testing or table-driven testing.

hh) *Load Testing:* Burden testing is a sort of non-practical testing; load testing is done to check the conduct of the product under ordinary and over top burden conditions. Burden testing is generally performed utilizing robotized testing instruments. Burden testing expects to discover bottlenecks or issues that keep programming from proceeding as planned at its pinnacle jobs.

ii) *Localization Testing:* Limitation testing a sort of programming testing performed by programming analyser, in this kind of testing, programming is relied upon to adjust to a specific district, it should uphold a specific area/language as far as show, tolerating contribution to that specific region, show, text style, date time, cash and so on, identified with a specific region. For example, many web applications permit decision of district like English, French, German or Japanese. So, whenever district is characterized or set in the design of programming, programming is required to fill in true to form with a set language/area.

jj) *Negative Testing:* This type of software testing approach, which calls out the “attitude to break”, these are functional and non-functional tests that are intended to break the software by entering incorrect data like incorrect date, time or string or upload binary file when text files supposed to be upload or enter huge text string for input fields etc. It is also a positive test for an error condition.

kk) *Non-functional Testing:* This sort of programming testing approach, which calls out the "mentality to break", these are useful and non-utilitarian tests that are proposed to break the product by entering inaccurate information like mistaken date, time or string or transfer parallel document when text records expected to be transfer or enter immense content string for input fields and so forth It is additionally a positive test for a mistake condition.

©IJRASET: All Rights are Reserved

ll) *Pair Testing:* Programming are worked to satisfy utilitarian and non-practical prerequisites, non-useful necessities like execution, convenience, restriction and so on, there are numerous kinds of testing like similarity testing, consistence testing, limitation testing, ease of use testing, volume testing and so forth, that are done for checking non-useful prerequisites.

mm) *Performance Testing:* It is a product testing method that should be possible by programming analysers, engineers or Business experts (BA). As the name proposes, two individuals are combined together, one to test and other to screen and record test results. Pair testing can likewise be acted in blend of analyser engineer, analyser business examiner or designer business investigator mix. Joining analysers and designers in pair testing assists with recognizing surrenders quicker, distinguish underlying driver, fix and test the fix.

nn) *Penetration Testing:* It is a kind of safety testing, otherwise called pen test in short. Entrance testing is done to tests how secure programming and its surroundings (Hardware, Operating framework and organization) are when liable to assault by an outer or inward gate crasher. Interloper can be a human/programmer or malignant projects. Pen test utilizes techniques to persuasively barge in (by savage power assault) or by utilizing a shortcoming (weakness) to access a product or information or equipment with an expectation to open approaches to take, control or degenerate information, programming documents or design. Infiltration Testing is a method of moral hacking, an accomplished Penetration analyser will utilize the very strategies and devices that a programmer would utilize however the expectation of Penetration analyser is to recognize weakness and sort them out before a genuine programmer or malignant program abuses it.

oo) *Regression Testing:* It is a kind of programming testing and some bit of productivity tuning that is performed to check a part of the quality credits of programming like Stability, reliability, openness. Execution testing is finished by operational proficiency bunch. Not in the least like Functional testing, Performance testing is done to check non-valuable essentials. Execution testing checks how well programming capacities in expected and top obligations. There are different assortments or sub sorts of execution like weight testing, stress testing, volume testing, soak testing and configuration testing.

pp) *Retesting:* It is a kind of retesting that is completed by programming analysers as a piece of imperfection fix confirmation. For example, an analyser is checking an imperfection fix and let us say that there are 3 experiments bombed because of this deformity. When analyser checks deformity fix as settled, test will retest or test a similar usefulness again by executing the experiments that were bombed before.

qq) *Risk based Testing:* It is a sort of programming testing and an alternate methodology towards testing a product. In Risk based testing, necessities and usefulness of programming to be tried are focused on as Critical, High, Medium and low. In this methodology, all basic and high need tests are tried and them followed by Medium. Low need or okay usefulness are tried toward the end or may not base on the time accessible for testing.

rr) *Smoke Testing:* It is a kind of testing that is completed by programming analysers to check if the new form given by improvement group is steady enough i.e., significant usefulness is functioning true to form to do further or nitty gritty testing. Smoke testing is planned to discover "gem" surrenders that can keep analysers from testing the application exhaustively. Smoke testing did for a form is otherwise called construct confirmation test.

ss) *Security Testing:* It is a kind of programming testing did by specific group of programming analysers. Objective of safety testing is to get the product is to outside or inward dangers from people and noxious projects. Security testing essentially checks, how great is programming's approval component, how solid is confirmation, how programming keeps up classification of the information, how does the product keep up uprightness of the information, what is the accessibility of the product in an occasion of an assault on the product by programmers and pernicious projects is for Security testing requires great information on application, innovation, organizing, security testing apparatuses. With expanding number of web applications essentially of safety testing has expanded indeed.

tt) *Sanity Testing:* It is a kind of testing that is completed for the most part by analysers and in certain ventures by engineers too. Mental soundness testing is a fast assessment of the product, climate, organization, outer frameworks are up and running, programming climate in general is sufficiently steady to continue with broad testing. Mental soundness tests are thin and more often than not mental stability tests are not archived.

uu) *Scalability Testing:* It is a non-utilitarian test proposed to test one of the quality credits of the product for example "Versatility". Adaptability test isn't centred around only one or few usefulness of the product rather execution of programming overall. Versatility testing is generally done by efficiency tuning group. Objective of adaptability testing is to test the capacity of the product to increase with expanded clients, expanded exchanges, expansion in data set size and so forth, it isn't required that product's exhibition increments with expansion in equipment setup, versatility tests assist with discovering the amount more responsibility the product can uphold with extending client base, exchanges, information stockpiling and so on.,

©IJRASET: All Rights are Reserved

vv) *Stability Testing:* It is a non-useful test expected to test one of the product qualities credits for example "Solidness". Security testing centres around testing how stable programming is the point at which it is liable to loads at satisfactory levels, top burdens, loads produced in spikes, with more volumes of information to be prepared. Adaptability testing will include performing various kinds of execution tests like burden testing, stress testing, spike testing, douse testing, spike testing and so on…

ww) *Static Testing:* It is a type of testing where in approaches like surveys, walkthroughs are utilized to assess the accuracy of the deliverable. In static testing programming code isn't executed rather it is looked into for language structure, remarking, naming show, size of the capacities and techniques and so on Static testing ordinarily has check records against which expectations are assessed. Static testing can be applied for necessities, plans, and experiments by utilizing approaches like surveys or walkthroughs.

xx) *Stress Testing:* It is a kind of execution testing, in which programming is exposed to top loads and even to a break highlight see how the product would act at breakpoint. Stress testing additionally tests the conduct of the product with deficient assets like CPU, Memory, Network transmission capacity, Disk space and so on Stress testing empowers to check a portion of the quality credits like vigour and dependability.

III.RESULTS AND DISCUSSION

A. *Program Discussion*

1)

```
"""importing the required libraries"""
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report,accuracy_score
from keras.models import Sequential
from keras.layers import Dense
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
import matplotlib.pyplot as plt
```

Fig 8: Importing required libraries

a) Here we used “pandas” to get and load the input data. The “pandas” expects to be the essential significant level structure block for doing viable, true information examination in Python. Also, it has the more extensive objective of turning into the most impressive and adaptable opensource information examination/control apparatus accessible in any language.

b) Sklearn.model_selection -Split arrays or matrices into irregular train and test subsections. Fast utility that wraps input approval and next (ShuffleSplit (). split (X, y)) and application to enter information into a solitary call for parting (and alternatively subsampling) information in a one-liner.

c) The sklearn. metrics module executes for accuracy, precision (relevant instant), recall (selected number of instants), F1 score (results the average of precision and recall) capacities to gauge arrangement execution.

d) Keras.model (sequential type) - It explicitly permits you to characterize numerous info or yield models just as models that offer layers. More than that, it permits you to characterize impromptu non-cyclic organization charts.

e) Keras.layer(dense) The dense layer is a neural organization layer that is associated profoundly, which implies every neuron in the dense layer gets contribution from all neurons of its past layer. The yield produced by the dense layer is an 'm' dimensional vector. Hence, dense layer is essentially utilized for changing the elements of the vector.

f) Sklearn.preprocessing(standard scale) Focusing and scaling happen autonomously on each component by figuring the important measurements on the examples in the preparation set. Mean and standard deviation are then put away to be utilized on later information utilizing change. Standardization of a dataset is a typical necessity for some, AI assessors: they may act seriously if the individual highlights don't pretty much look like standard regularly circulated information (for example Gaussian with 0 mean and unit variance).

g) Tensor flow (tf) tf. function is useful in creating and using computational graphs, they should be used in training and in deployment, however it isnt needed for most of your functions.

h) Import matplotlib. pyplot as plt shows to unknown readers that pyplot is a module, not a function that might be misled from the first form.

©IJRASET: All Rights are Reserved

2)
```
"""Reading the csv file"""

data=pd.read_csv('cpdata.csv')
print(data.head(5))
print(data.info())
print(data.describe())

print(data.isna().sum())
```
Fig 9: Reading CSV values

CSV files (comma separated values) are a common file format used to transfer and store data.
The head (5) function is employed to fetch the first 5 rows. It is used to have a quick check whether your object consists of the correct type of data. For negative n values, head () returns all but the last n rows that match df [: n].
The info () function is employed to print the synopsis of the Data Frame. This involves printing information about the Data Frame, taking into account index dtype and column types, non-zero values, and memory utilization. Or print out the complete summary. By default, it is set in Pandas.
The describe () is employed to calculate some statistics such as percentiles, mean values, and standards of values in a series or a DataFrame. It analyzes object and number rows and Data Frame column sets with mixed data types.
The isna () function is employed to find missing values. Return a Boolean value of the same size, indicating whether the value is NA.NA values, namely None or numpy. NaN, gets coordinated to true values.
The sum () returns the addition of values of the appealed axis. If the input is an index axis, sums all values into one column, repeat the same operation for columns, and return a row containing the sum of all values in each column.

3)
```
"""Creating dummy variable for target i.e label"""

label= pd.get_dummies(data.label).iloc[: , 1:]
data= pd.concat([data,label],axis=1)
print(data.head(5))
data.drop('label', axis=1,inplace=True)

print('The data present in one row of the dataset is')
print(data.head(1))
```
Fig 10: Creating dummy variable

Data manipulation is done by get_dummies (). Removal of unnecessary interruptions like missing data and label encoding takes places by this.

4)
```
print('The data present in one row of the dataset is')
print(data.head(1))

X=data.iloc[:, :-1].values
y=data.iloc[: ,-1].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20)
print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)
```
Fig 11: Data split

Splitting of data into train and test sets occurs.

5)
```
"""Performing standard scaler"""

sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```
Fig 12: Performing standard scaler

©IJRASET: All Rights are Reserved

The values are converted into binary values.

6)

```
"ANN"

print("ANN")
print()
classifier = Sequential()

classifier.add(Dense(units = 6, kernel_initializer = 'uniform', activation = 'relu', input_dim = 33))
classifier.add(Dense(units = 6, kernel_initializer = 'uniform', activation = 'relu'))
classifier.add(Dense(units = 1, kernel_initializer = 'uniform', activation = 'sigmoid'))
classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
classifier.fit(X_train, y_train, batch_size = 20, epochs = 5,verbose = 1)
history=classifier.history.history
y_pred = classifier.predict(X_test)
y_pred = (y_pred > 0.5)

"Analysis Report"
print()
print("------Classification Report------")
print(classification_report(y_pred,y_test))

print()
print("------Accuracy------")
print(f"The Accuracy Score :{(accuracy_score(y_pred,y_test)*100)}")
print()

train_loss = history['loss']
train_acc = history['accuracy']

# acc
plt.figure()
plt.plot(train_loss, label='Training Loss')
plt.plot(train_acc, label='Accuracy')
plt.title('ANN Peformance Metrics Plot')
plt.legend()
plt.show()
```

Fig 13: ANN algorithm

ANN initialisation and execution takes places in supervised classification algorithm.

7)

```
print('\n')
print("LSTM")
print("\n")

X_train = tf.keras.utils.normalize(X_train,axis=1)
X_test = tf.keras.utils.normalize(X_test,axis=1)

LSTM = tf.keras.models.Sequential()

LSTM.add(tf.keras.layers.Flatten())
LSTM.add(tf.keras.layers.Dense(128,activation=tf.nn.relu))
LSTM.add(tf.keras.layers.Dense(128,activation=tf.nn.relu))
LSTM.add(tf.keras.layers.Dense(10,activation=tf.nn.softmax))

LSTM.compile(optimizer='adam',
             loss='sparse_categorical_crossentropy',
             metrics=['accuracy'])
history=LSTM.fit(X_train,y_train,epochs=8)
history=LSTM.history.history
y_pred = classifier.predict(X_test)

val_loss, val_acc = LSTM.evaluate(X_test,y_test)
print("validation loss:",val_loss)
print("The Accuracy Score :",(val_acc)*100)


train_loss = history['loss']
train_acc = history['acc']

# acc
plt.figure()
plt.plot(train_loss, label='Training Loss')
plt.plot(train_acc, label='Accuracy')
plt.title('LSTM Peformance Metrics Plot')
plt.legend()
plt.show()
```

Fig 14: LSTM algorithm

LSTM initialisation and execution takes places in supervised classification algorithm.

©IJRASET: All Rights are Reserved

B. Result Discussion

1) Data Selection And Loading

The below mentioned figures illustrate the dataset which we are taken for an input for our crop yield that got loaded and converted it into the data frame.

The name of the crop whose parameters are given as the input will appear in the label column whereas other crops column shows false statement that is '0' in binary value.

Fig 15: Data frame1

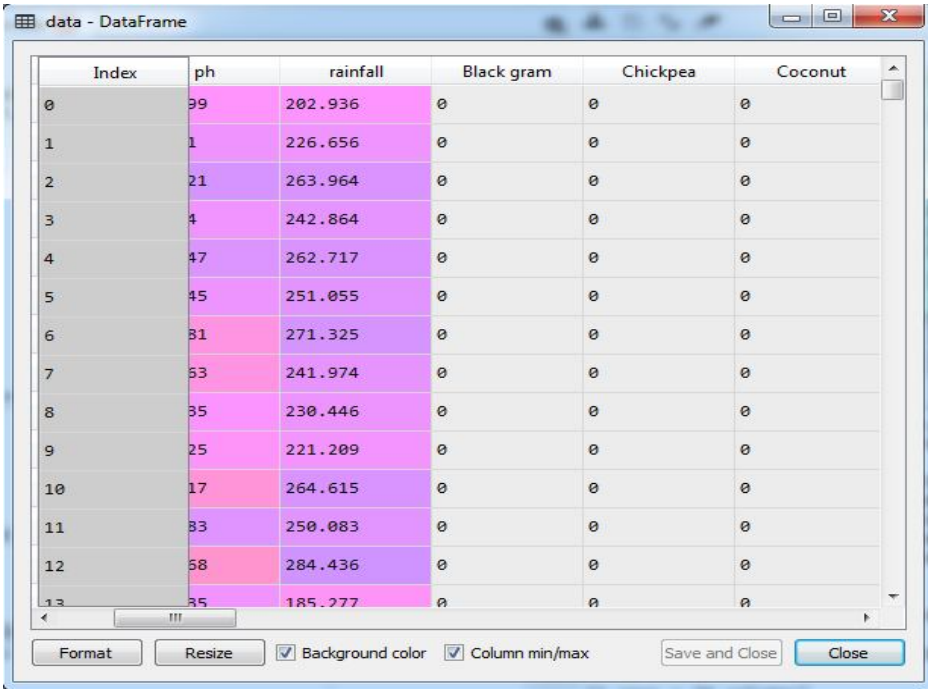


Fig 16: Data frame 2

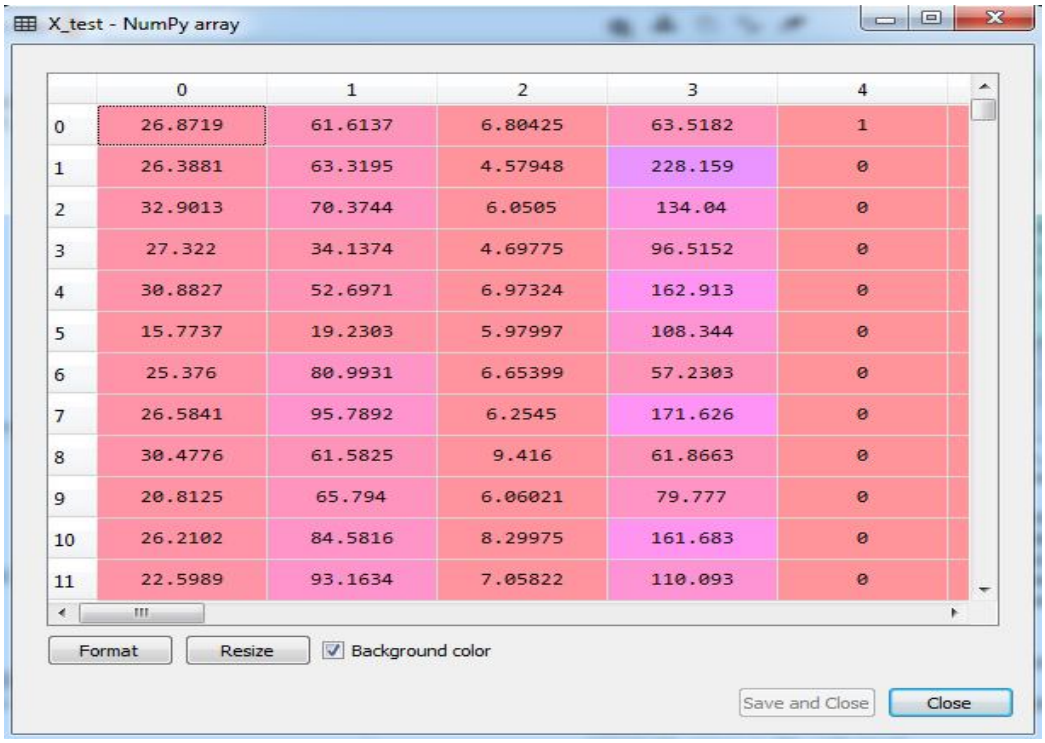
©IJRASET: All Rights are Reserved

2) Pre-Processed And Split Train And Test Data

	0	1	2	3	4
0	23.5611	58.6493	4.62192	213.453	0
1	32.3952	94.5177	6.6014	113.254	0
2	28.7973	80.4574	6.72555	44.3007	0
3	21.4382	62.7815	4.89382	215.671	0
4	27.705	63.2092	7.74672	37.4616	0
5	31.6553	60.1326	6.52669	66.691	1
6	24.0436	84.1841	6.4239	53.7893	0
7	18.654	61.3788	6.65673	93.6204	0
8	19.4621	18.7283	7.21702	68.8141	0
9	26.0858	56.0677	6.00835	152.1	0
10	25.8693	61.8832	7.07292	36.6828	0
11	28.3119	75.7736	6.165	119.696	0

Fig 17: X Train

The above figure illustrates the dataset which we are taken for an input after pre-processing and from that it split 70 % or 80% of train data using sklearn library package (from sklearn.model_selection import train_test_split)

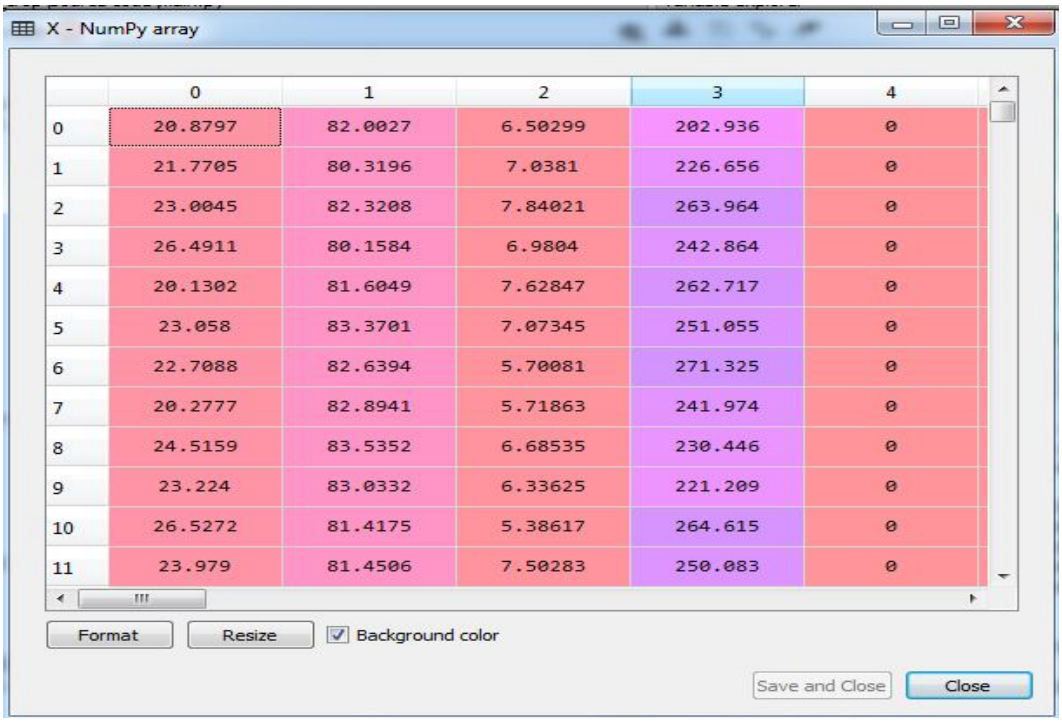


	0	1	2	3	4
0	26.8719	61.6137	6.80425	63.5182	1
1	26.3881	63.3195	4.57948	228.159	0
2	32.9013	70.3744	6.0505	134.04	0
3	27.322	34.1374	4.69775	96.5152	0
4	30.8827	52.6971	6.97324	162.913	0
5	15.7737	19.2303	5.97997	108.344	0
6	25.376	80.9931	6.65399	57.2303	0
7	26.5841	95.7892	6.2545	171.626	0
8	30.4776	61.5825	9.416	61.8663	0
9	20.8125	65.794	6.06021	79.777	0
10	26.2102	84.5816	8.29975	161.683	0
11	22.5989	93.1634	7.05822	110.093	0

Fig 18: X Test

The above figure illustrates the dataset for an input after pre-processing and from that it split 30 % or 20% of test data using sklearn library package (from sklearn.model_selection import train_test_split)

©IJRASET: All Rights are Reserved



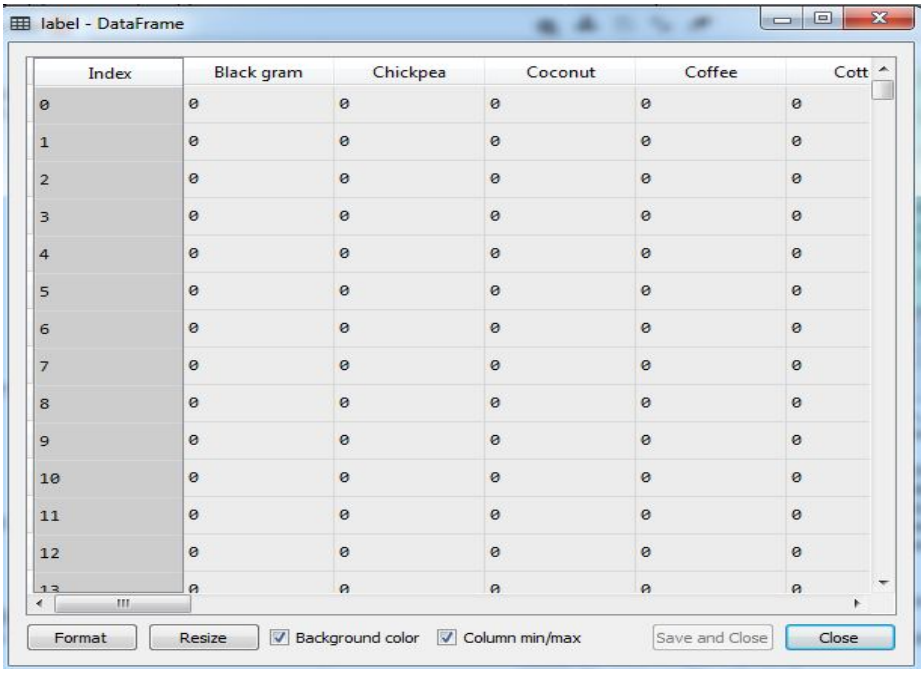
	0	1	2	3	4
0	20.8797	82.0027	6.50299	202.936	0
1	21.7705	80.3196	7.0381	226.656	0
2	23.0045	82.3208	7.84021	263.964	0
3	26.4911	80.1584	6.9804	242.864	0
4	20.1302	81.6049	7.62847	262.717	0
5	23.058	83.3701	7.07345	251.055	0
6	22.7088	82.6394	5.70081	271.325	0
7	20.2777	82.8941	5.71863	241.974	0
8	24.5159	83.5352	6.68535	230.446	0
9	23.224	83.0332	6.33625	221.209	0
10	26.5272	81.4175	5.38617	264.615	0
11	23.979	81.4506	7.50283	250.083	0

Fig 19: X Array data

The above figure illustrates the dataset for the inputs after pre-processing and here the values of the dataset got converted into the form of array stored as 'X' values.

3) Label Encoding

The below figure illustrates the dataset which shows the input after pre-processing and we label the dataset column using one-hot encoder and label encoder. Label encoder is used to convert column values into binary values and one hot encoder is used to convert the labelled column into an array.



Index	Black gram	Chickpea	Coconut	Coffee	Cott
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0

Fig 20: Label

©IJRASET: All Rights are Reserved

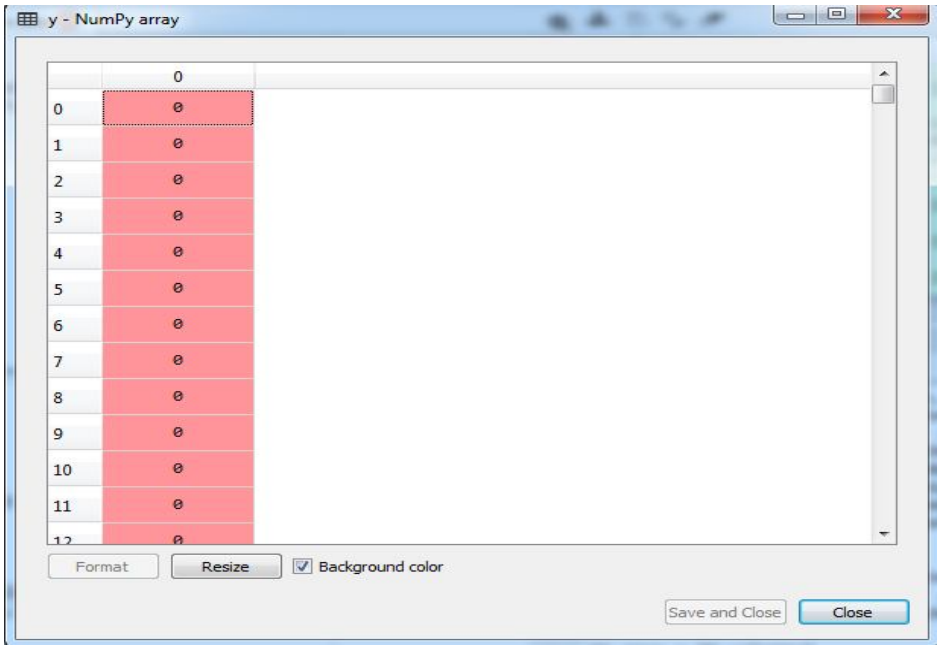


Fig 21: Y array

The above figure illustrates the dataset for an input after pre-processing and we store the class label column as a 'y' value and stored in a data frame.

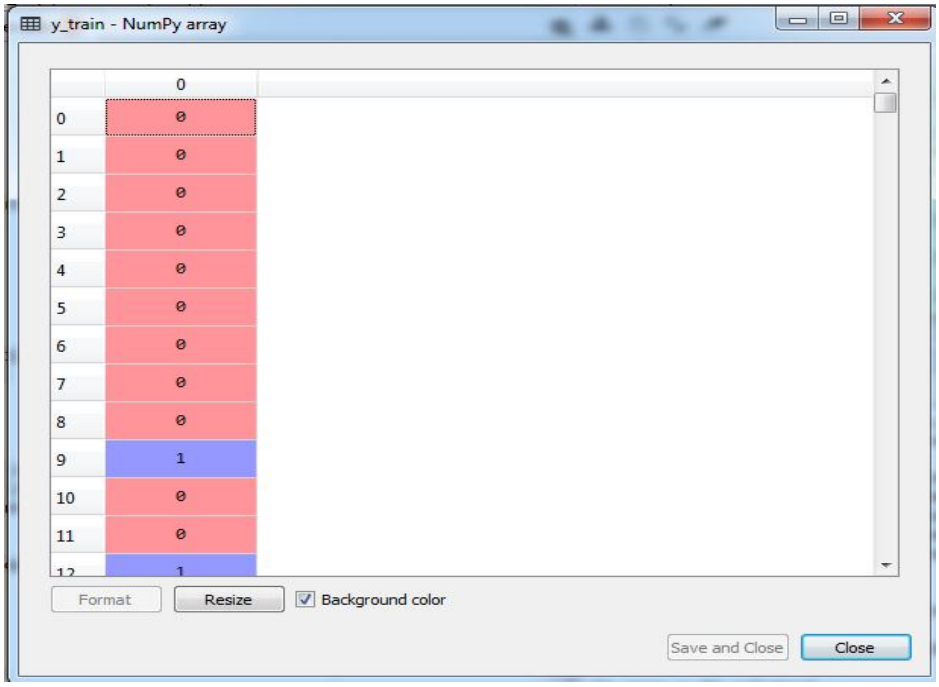


Fig 22: Y train

The above figure illustrates the results of each of the train data input that is the '1' value represented here proves the truth therefore identifies the label of the train data.

The below figure illustrates the results of each of the train data input that is the '1' value represented here proves the truth therefore identifies the label of the test data.

©IJRASET: All Rights are Reserved

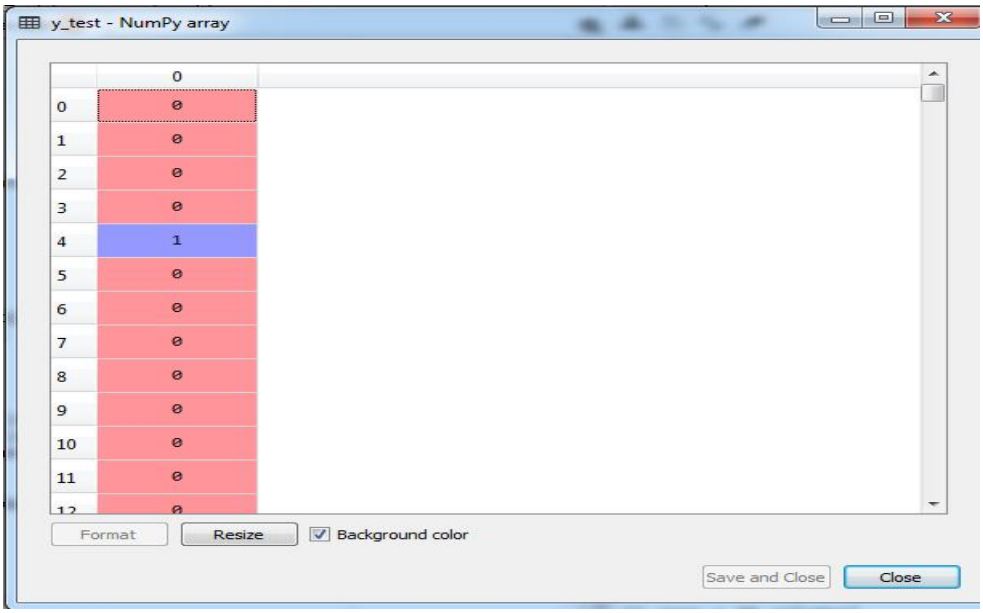


Fig 23: Y test

4) Cross Validation Over ANN and LSTM

In this paper, we analysed the crop yield like rice, wheat, watermelon, and so on. The input data was collected from the dataset repository (UCI). After that, we performed the pre-processing steps. In this step, we handled missing values. The purpose of finding missing values is to avoid the wrong prediction. After that, we have split the data into test and train. The purpose of this step is, cross validation. The data is split into test and train based on the respective ratio. Most of the data are stored in train part and smaller portion of the data is stored in test part. Here, we implemented the two classification methods such as ANN (Artificial Neural Network) and LSTM. In classification, we classified the crop yield based on both algorithms. In algorithm, it takes place in two categories. One is fit part. In fit part, we fit the training data. The other one is prediction. In prediction, we predicted the test data. The experimental results show the accuracy of each algorithm. Finally, the experimental results show the accuracy (i.e) this is overall prediction. In our process, the LSTM algorithm gave the 100% result when compare with the ANN algorithm.

- The primary distinction among ANN and LSTM is as far as which one keep up with data in the memory for the extensive stretch of time. Here LSTM enjoys upper hand over ANN as LSTM can deal with the data in memory for the significant stretch of time as contrast with ANN.
- LSTM model tackles the issue of gradients vanishing by presenting another state called cell state and having a CEC (Constant Error Carousel) which permits the blunder to spread back without disappearing.

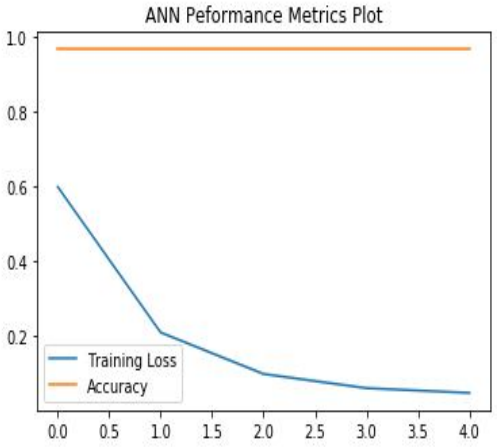


Fig 24: Validation graph for ANN

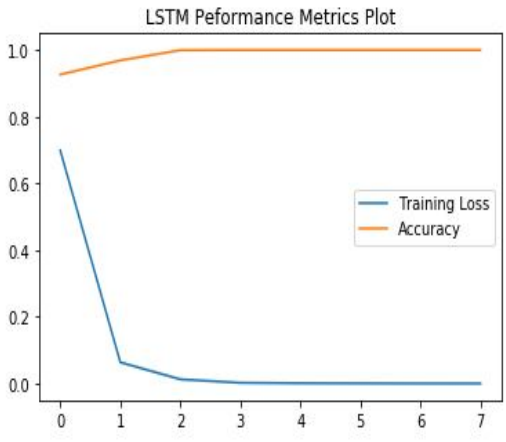


Fig 25:Validation graph for LSTM

©IJRASET: All Rights are Reserved

a) ANN

- *Classification Report*

	Precision	Recall	Fi-score	Support
0	1.00	0.97	0.98	620
1	0.00	0.00	0.00	0
Micro average	0.97	0.97	0.97	620
Macro average	0.50	0.48	0.49	620
Weighted average	1.00	0.97	0.98	620

Table 1: ANN Classification report

- *Accuracy*

The Accuracy Score :96.93548387096774

b) LSTM

Epoch 8/8
2480/2480 [==============================] - 0s 75us/sample - loss: 2.7828e-04 - acc: 1.0000
620/620 [==============================] - 0s 277us/sample - loss: 2.7753e-04 - acc: 1.0000
validation loss: 0.00027752817099568466
Validation Accuracy: 100.0

IV. CONCLUSIONS

I infer that in this interaction I attempted to give an expectation on how the profound learning calculation can change our agribusiness information. Being subject to the farming side for quite a while our nation needs to meet much among agribusiness and innovation up until this point. Along these lines, we presented a solid and cheap technique to foresee the harvest yield from the dataset. For the forecast of harvest yield utilizing profound learning calculation like ANN and LSTM. Both the strategy works straightforwardly and it delivers great execution metrices. Keeping in mind the accuracy and less loss, LSTM is a useful technique for crop yield prediction. Further, my project shall be forwarded with the aim of getting improvised by involving other agriculture related parameters into the prediction.

V. ACKNOWLEDGMENT

I would like to enunciate heartfelt thanks to our esteemed Chairman Dr. S.V. Balasubramaniam, and the respected Director Dr.M.P. Vijay kumar, for providing excellent facilities and support during the course of study in this institute.

I am grateful to Dr. Chelladurai. V, Head of Department, Agriculture Engineering for his / her valuable suggestions to carry out the project work successfully. I wish to express our sincere thanks to Faculty guide Mr. Nimkar Amey Sanjay, Assistant Professor, Agriculture Engineering for his/ her constructive ideas, inspirations, encouragement, excellent guidance and much needed technical support extended to complete our project work. I would like to thank our friends, faculty and non-teaching staff who have directly and indirectly contributed to the success of this project.

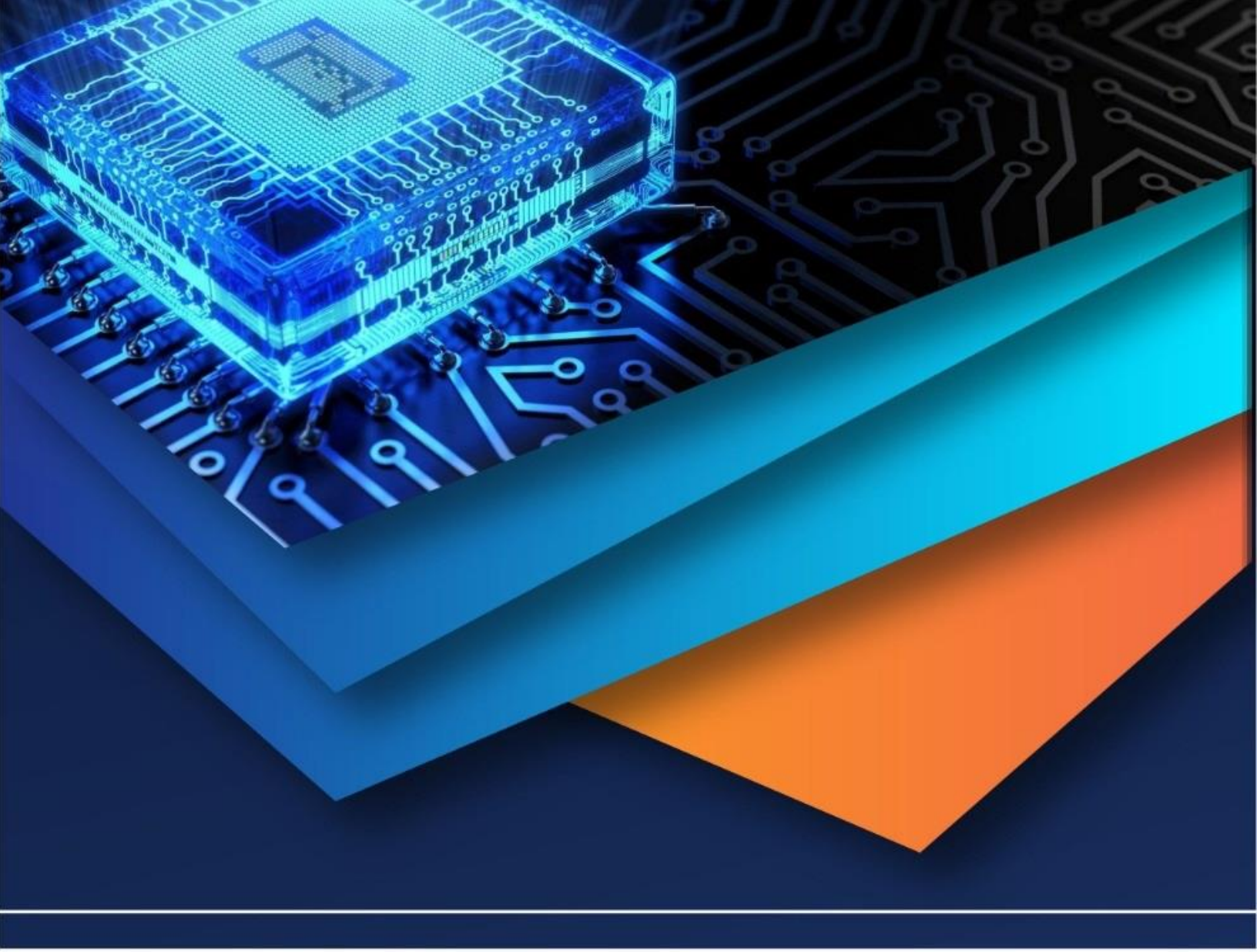
REFERENCES

[1] S. Li, S. Peng, W. Chen, and X. Lu, ‘‘INCOME: Practical land monitoring in precision agriculture with sensor networks,’’ Comput. Commun., vol. 36, no. 4, pp. 459–467, Feb. 2013.
[2] A. D. Jones, F. M. Ngure, G. Pelto, and S. L. Young, ‘‘What are we assessing when we measure food security? A compendium and review of current metrics,’’ Adv. Nutrition, vol. 4, no. 5, pp. 481–505, 2013.
[3] G. E. O. Ogutu, W. H. P. Franssen, I. Supit, P. Omondi, and R. W. Hutjes, ‘‘Probabilistic maize yield prediction over East Africa using dynamic ensemble seasonal climate forecasts,’’ Agricult. Forest Meteorol., vols. 250–251, pp. 243–261, Mar. 2018.

©IJRASET: All Rights are Reserved

[4] M. E. Holzman, F. Carmona, R. Rivas, and R. Niclòs, ‘‘Early assessment of crop yield from remotely sensed water stress and solar radiation data,’’ ISPRS J. Photogramm. Remote Sens., vol. 145, pp. 297–308, Nov. 2018.
[5] A. Singh, B. Ganapathysubramanian, A. K. Singh, and S. Sarkar, ‘‘Machine learning for high-throughput stress phenotyping in plants,’’ Trends Plant Sci., vol. 21, no. 2, pp. 110–124, 2016.
[6] R. Whetton, Y. Zhao, S. Shaddad, and A. M. Mouazen, ‘‘Nonlinear parametric modelling to study how soil properties affect crop yields and NDVI,’’ Comput. Electron. Agricult., vol. 138, pp. 127–136, Jun. 2017.
[7] Y. Dash, S. K. Mishra, and B. K. Panigrahi, ‘‘Rainfall prediction for the Kerala state of India using artificial intelligence approaches,’’ Comput. Elect. Eng., vol. 70, pp. 66–73, Aug. 2018.
[8] W. Wieder, S. Shoop, L. Barna, T. Franz, and C. Finkenbiner, ‘‘Comparison of soil strength measurements of agricultural soils in Nebraska,’’ J. Terramech., vol. 77, pp. 31–48, Jun. 2018.
[9] Y. Cai, K. Guan, J. Peng, S. Wang, C. Seifert, B. Wardlow, and Z. Li, ‘‘A high-performance and in-season classification system of field-level crop types using time-series Landsat data and a machine learning approach,’’ Remote Sens. Environ., vol. 210, pp. 35–47, Jun. 2018.
[10] X. E. Pantazi, D. Moshou, T. Alexandridis, R. L. Whetton, and A. M. Mouazen, ‘‘Wheat yield prediction using machine learning and advanced sensing techniques,’’ Comput. Electron. Agricult., vol. 121, pp. 57–65, Feb. 2016.
[11] T. U. Rehman, S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, ‘‘Current and future applications of statistical machine learning algorithms for agricultural machine vision systems,’’ Comput. Electron. Agricult., vol. 156, pp. 585–605, Jan. 2019.
[12] D. Elavarasan, D. R. Vincent, V. Sharma, A. Y. Zomaya, and K. Srinivasan, ‘‘Forecasting yield by integrating agrarian factors and machine learning models: A survey,’’ Comput. Electron. Agricult., vol. 155, pp. 257–282, Dec. 2018.
[13] M. D. Johnson, W. W. Hsieh, A. J. Cannon, A. Davidson, and F. Bédard, ‘‘Crop yield forecasting on the Canadian Prairies by remotely sensed vegetation indices and machine learning methods,’’ Agricult. Forest Meteorol., vols. 218–219, pp. 74–84, Mar. 2016.
[14] A. Kaya, A. S. Keceli, C. Catal, H. Y. Yalic, H. Temucin, and B. Tekinerdogan, ‘‘Analysis of transfer learning for deep neural network based plant classification models,’’ Comput. Electron. Agricult., vol. 158, pp. 20–29, Mar. 2019.
[15] A. Kamilaris and F. X. Prenafeta-Boldú, ‘‘Deep learning in agriculture: A survey,’’ Comput. Electron. Agricult., vol. 147, pp. 70–90, Apr. 2018.
[16] I. M. Evans, ‘‘Reinforcement, principle,’’ in International Encyclopedia of the Social & Behavioral Sciences, J. D. Wright, 2nd ed. Amsterdam, The Netherlands: Elsevier, 2015, pp. 207–210.
[17] D. Vogiatzis and A. Stafylopatis, ‘‘Reinforcement learning for rule extraction from a labeled dataset,’’ Cognit. Syst. Res., vol. 3, no. 2, pp. 237–253, Jun. 2002.
[18] D. A. Temesgene, M. Miozzo, and P. Dini, ‘‘Dynamic control of functional splits for energy harvesting virtual small cells: A distributed reinforcement learning approach,’’ Comput. Commun., vol. 148, pp. 48–61, Dec. 2019.
[19] S. Wan, Z. Gu, and Q. Ni, ‘‘Cognitive computing and wireless communications on the edge for healthcare service robots,’’ Comput. Commun., vol. 149, pp. 99–106, Jan. 2020.
[20] A. Tolba, O. Said, and Z. Al-Makhadmeh, ‘‘MDS: Multi-level decision system for patient behavior analysis based on wearable device information,’’ Comput. Commun., vol. 147, pp. 180–187, Nov. 2019.
[21] V. Hassija, V. Saxena, and V. Chamola, ‘‘Scheduling drone charging for multi-drone network based on consensus time-stamp and game theory,’’ Comput. Commun., vol. 149, pp. 51–61, Jan. 2020.
[22] S. Gheisari and E. Tahavori, ‘‘CCCLA: A cognitive approach for congestion control in Internet of Things using a game of learning automata,’’ Comput. Commun., vol. 147, pp. 40–49, Nov. 2019.
[23] S. J. Shri and S. Jothilakshmi, ‘‘Crowd video event classification using convolutional neural network,’’ Comput. Commun., vol. 147, pp. 35–39, Nov. 2019.
[24] M. Al-Ayyoub, A. Nuseir, K. Alsmearat, Y. Jararweh, and B. Gupta, ‘‘Deep learning for arabic NLP: A survey,’’ J. Comput. Sci., vol. 26, pp. 522–531, May 2018.
[25] M. Usama, B. Ahmad, J. Yang, S. Qamar, P. Ahmad, Y. Zhang, J. Lv, and J. Guna, ‘‘Equipping recurrent neural network with CNN-style attention mechanisms for sentiment analysis of network reviews,’’ Comput. Commun., vol. 148, pp. 98–106, Dec. 2019

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)