Volume IV, Issue 2
October 2023

American Journal of **Electronics & Communication**

Society for Makers, Artists, Researchers and Technologists,
USA 6408 Elizabeth Avenue SE, Auburn, Washington 98092.
ISSN 2690-2087

Contents

American Journal of
Electronics & Communication

Anisha Saha et. al., American Journal of Electronics & Communication, Vol. IV (2), 1-6

A Comparative Study to Predict Polycystic Ovarian Syndrome (PCOS) Based on Different Models of Machine Learning Technique

[1]Anisha Saha, [2]Aporna Roy, [3]Barsha Chakraborty, [4]Bidisha Saha, [5]Dipwanita Chowdhury
[anisha.sahaiembca2023, royaporna4, cbarsha119, bidishasaha50864, dipwanitachowdhury]@gmail.com
BCA 3rd Year, IEM, Kolkata

[6]Prof. Manab Kumar Das, [7]Prof. Soham Goswami
[manab.das, soham.goswami]@iem.edu.in
Assistant Professor, IEM, Kolkata

***Abstract* - Polycystic ovarian syndrome (PCOS) is a common endocrine disorder affecting women of reproductive age worldwide, characterized by excess production of androgens. This can result in ovarian abnormalities and a range of associated health risks, including infertility, heart issues, diabetes, and uterine cancer. However, the diagnosis of PCOS can be challenging due to the varied symptoms in different women and the time and cost involved in biochemical tests and ovarian scanning. To address this, researchers have proposed a method that predicts the likelihood of PCOS based on a minimal set of criteria, including weight, BMI, cycle length, and hormone levels. Using five machine learning algorithms, they tested the method on a dataset of 541 patients and found that the Random Forest and Support Vector Machine models had the highest accuracy in predicting PCOS. Such a system could aid in early detection and encourage individuals at risk to seek medical attention. Dataset is split into a 70/30 ratio, meaning that 70% of the dataset's data are used to train the model and 30% are used to test it. In this paper, we suggested a novel stack model with a 90% accuracy that is composed of four machine learning classifiers: Random Forest, Support Vector Machine, Naive Bayes, and Logistic Regression. Testing data accuracies for the models of Logistic Regression, Random Forest, Support Vector Machine, K Nearest Neighbor, Naive Bayes, Stack Model are 88%, 91%, 90%, 69%, 86% and 90% respectively. As a result, the models with the highest accuracy on the testing data are the Random Forest model and Stack Model.**

Keywords – Machine Learning Algorithms, PCOS, Ensemble Learning, Feature Selection, PRASOON KOTTARATHIL Dataset.

I. Introduction

Polycystic Ovary Syndrome (PCOS) is a medical condition which causes hormonal disorder in women in their childbearing years. In the case of PCOS, ovaries can bulge and sometimes may have multiple small cyst formations (immature follicles). PCOS women have high levels of male hormones and insufficient female hormones, leading to alteration in their menstrual cycle or even absent menstrual cycle. Women with PCOS majorly suffer from excessive weight gain, facial hair growth, acne, hair loss, skin darkening and irregular periods leading to infertility.

The healthcare industry could undergo numerous revolutions because of artificial intelligence (AI). AI's capacity to analyze massive amounts of data correctly and rapidly is a key advantage. This enables medical workers to make better choices about patient care, such as individualized treatment plans and quicker evaluations. AI can also help with medical imagery by offering more precise and effective picture analysis, which lowers the chance of misdiagnosis. Additionally, real-time patient health monitoring by AI-powered gadgets enables early identification of possible health issues and prompt medical assistance. AI can also aid in the finding of new drugs by analyzing vast quantities of data and spotting potential treatments that may have gone unnoticed. Overall, AI has the potential to boost productivity, lower expenses, and enhance service quality in the healthcare industry.

The general factors of PCOS such as heredity, fast food, diet habits, involvement in physical exercise, BMI etc. The long-term effects of polycystic ovaries can cause significant ailments such endometrial hyperplasia, coronary disease, and type 2 diabetes mellitus. Studies have shown that it can also result in various malignancies including uterine or breast cancer in women who are fertile. Identifying PCOS is tricky due to all these manifestations, gynecological, clinical and metabolic parameters involved in diagnosing it. So, the time and financial expenses have become a hardship to the patients.

Our contributions in this paper-

- By integrating the Random Forest, Support Vector Machine, Naive Bayes, and Logistic Regression models, we have created a stack model that provides 97% accuracy on training datasets and 90% accuracy on testing datasets.
- Additionally, we compared the accuracies of Logistic Regression, Random Forest, Support Vector Machine, K Nearest Neighbor, and Naive Bayes model and the most accurate model was the random forest one.
- We used the Prasoon Kottarathil Dataset for our research and also preprocessed the data by managing null values and extracting features using methods like Pearson's correlation coefficient and the k-best

algorithm approach to improve the performance of our models.

II. Literature Survey

This study provides an in-depth analysis of PCOS and explores the use of image processing and machine learning techniques to aid in its diagnosis and potential automation. A range of analytical techniques have been utilized to detect and analyze PCOS.

The intention for conducting PCOS research is multifaceted, and can include addressing a major public health issue, advancing scientific knowledge, developing personalised care, and improving patient health outcomes. PCOS research can aid in the identification of risk factors, biomarkers, and other indicators that predict the development of PCOS, allowing for earlier interventions and personalised treatment plans. The ultimate goal is to improve the quality of life for women suffering from PCOS. To gain a comprehensive understanding of PCOS, it is necessary to reference established diagnostic criteria and standards.

M. Sumathi et al. [1] constructed a CNN image processing model for disease classification using ultrasound images. Feature extraction was performed using the watershed algorithm, and parameter measurement was carried out using OpenCV. The model achieved a high accuracy of 85% based on performance factors. Irfan Talib et al. [2] found that elevated insulin levels are a leading factor in the development of PCOS. Their study examined the diverse effects of insulin resistance in women with polycystic ovaries. R.M.Dewi et al. [3] proposed a method to accurately classify polycystic ovaries using ultrasound images. The authors employed a combination of feature extraction techniques, specifically the Wavelet method, and a Convolutional Neural Network (CNN) to identify the unique characteristics of the ultrasound data. The results of the system testing indicated that the CNN achieved the highest accuracy of 80.84% in accurately identifying the polycystic ovaries.

AUTHORS	OBJECTIVES	RESEARCH DESIGN	RESULTS
S. Sreejith et al.[2022][4]	In order to help physicians monitor Polycystic Ovarian Syndrome, this study creates a clinical decision support system (PCOS).	Utilized a random forest classifier to analyze the characteristics after using the red deer method to identify the best ones.	In terms of accuracy, sensitivity, and specificity, the proposed methodology (RF+RDA) performs better than existing wrapper approaches employing RF and conventional classifiers, with scores of 89.81% accuracy, 90.43% specificity, and 89.73% sensitivity.
M A Anusuya et al.[2020][5]	A method for assessing and tracking symptoms that allows the chance of having PCOS to be predicted based on characteristics like testosterone levels, hirsutism, family history, Obesity, etc.	KNN, Linear Regression, and Random Forest are some of the machine learning supervised classification algorithms that have been utilized for prediction tasks.	The random forest approach outperforms the other two algorithms by averaging lower error levels (average MAE and RMSE values of 1.99 and 3.10, respectively) and the highest R^2 values (average 0.985).
B Rachana et al.[2021][6]	Discovering a way to detect PCOS in its earliest stages to avert additional difficulties.	The suggested technique includes a KNN classifier, which is primarily focused on decreasing a number of flaws, and classification is done using the KNN algorithm.	It is feasible to demonstrate that the KNN classifier has an accuracy of nearly 97%, which is higher than any classifier that has previously been suggested.
Vaidehi Thakre et .al.[2020] [7]	A method that can anticipate the therapy for PCOS based on an ideal and minimum set of characteristics has been presented.	Random forest, SVM, Logistic Regression, Gaussian Naive Bayes, and KNN are the five algorithms that were tested to predict PCOS.	Of the four, the Random Forest Classifier was found to be the most trustworthy and accurate, with an accuracy rate of 90.9%.
A.K.M. Salman Hosain et al.[2022][8]	To detect Polycystic Ovary Syndrome(PCOS) using Convolutional Neural Network Architecture from Ovarian Ultrasound Images	For the purpose of classifying data, they had created the pre-trained model InceptionV3 and the CNN model PCONet to identify ovarian cysts in ultrasound images.	The best model created is PCONett, which has an accuracy rate of 93.93%.
Amsy Denny et al.[2019][9]	Machine Learning-Based Diagnosis and Prediction System For Polycystic Ovary Syndrome (PCOS)	Used Logistic regression and six other algorithms such as Linear Discriminate Analysis, KNN, CART, RFC, NBC, SVM.	The best performance was given by Random Forest Classifier model, where an accuracy of 89 % was achieved after data optimization.
Kinjal Raut et al.[2022][10]	Machine Learning	Decision Tree, SVC, Random	Comparing CatBoostClassifier to other models, it

	Algorithms for PCOS Detection.	Forest, Logistic Regression, K Nearest Neighbor, XGBRF, and CatBoost Classifier are the methods used to build the model.	has excelled and achieved the greatest accuracy of 94.64%.

Table-1: Summary of Literature Review

III. METHODOLOGY

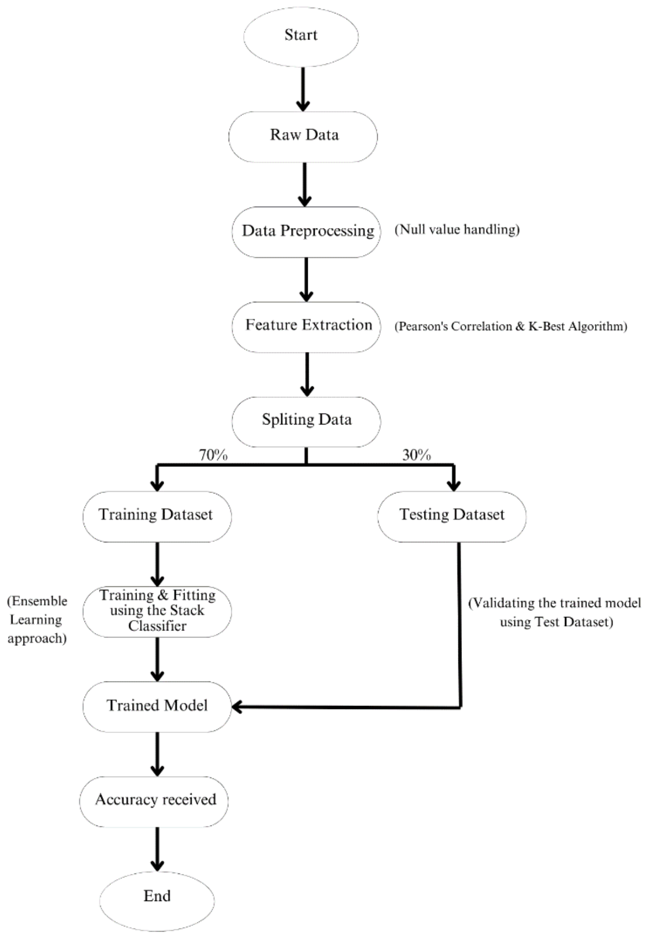

Start
Raw Data
Data Preprocessing
(Null value handling)
Feature Extraction
(Pearson's Correlation & K-Best Algorithm)
Spliting Data
70%
30%
Training Dataset
Testing Dataset
(Ensemble Learning approach)
Training & Fitting using the Stack Classifier
(Validating the trained model using Test Dataset)
Trained Model
Accuracy received
End

Fig-1: The flowchart of the entire process, from data gathering to precision getting

Proposed Algorithm: -

Step 1: *Dataset Description: -*

Polycystic ovary syndrome (PCOS) is a condition characterized by menstrual irregularities and high levels of male hormones. It is a significant contributor to infertility in women. To develop an accurate diagnostic model for PCOS, a combination of clinical and non-clinical data is needed. The dataset used in this study comprises data from 541 patients with and without fertility issues who were diagnosed with PCOS. The data was collected from 10 different hospitals in Kerala, India and is available in the Kaggle database. Table 1 provides a detailed description of the parameters that are included in the dataset.

Sl. No.	Parameter name	Description
1	Age	Patient's age in years
2	Weight	Patient's weight in kg
3	Height	Patient's height in cm
4	BMI	Body mass index
5	Blood group	Blood group
6	Pulse Rate	Pulse rate in beats per minute
7	RR	Respiratory rate in breaths per minute
8	Hb	Haemoglobin counts in grams per decilitre
9	Cycle (R/I)	Whether cycle is regular (2) or not(4)
10	Cycle length(days)	Number of days of menstruation
11	Marriage Status (Yrs.)	Number of years since marriage
12	Pregnant(Y/N)	Whether pregnant (1) or not(0)
13	No. of abortions	Number of abortions
14	I beta-HCG(mIU/mL)	Amount of beta human chorionic gonadotropin
15	II beta-HCG(mIU/mL)	Amount of beta human chorionic gonadotropin
16	FSH(mIU/mL)	Amount of follicles stimulating hormone
17	LH(mIU/mL)	Amount of Luteinizing hormone
18	FSH/LH	Ratio of FSH to LH
19	Hip(inch)	Hip size in inches
20	Waist(inch)	Waist size in inches
21	Waist: Hip Ratio	Waist to hip ratio
22	TSH (mIU/L)	Amount of Thyroid Stimulating hormone
23	AMH (ng/mL)	Amount of Anti Mullerian hormone
24	PRL (ng/mL)	Amount of Prolactin
25	Vit D3 (ng/mL)	Amount of Vitamin D3
26	PRG (ng/mL)	Amount of progesterone
27	RBS (mg/dl)	Random Blood Glucose
28	Weight gain(Y/N)	Whether the patient gained weight (1) or not (0)
29	hair growth(Y/N)	Whether the patient had hair growth (1) or not (0)
30	Skin darkening (Y/N)	Whether the patient had skin darkening (1) or not (0)
31	Hair loss(Y/N)	Whether the patient experienced hair loss (1) or not (0)
32	Pimples(Y/N)	Whether the patient has pimples (1) or not (0)
33	Fast food (Y/N)	Whether the patient consumes fast food (1) or not (0)

34	Reg..Exercise(Y/N)	Whether the patient exercises regularly (1) or not(0)
35	BP_Systolic (mmHg)	Systolic pressure
36	BP_Diastolic (mmHg)	Diastolic pressure
37	Follicle No. (L)	No: of follicles in the left ovary
38	Follicle No. (R)	No: of follicles in the right ovary
39	Avg. F size (L) (mm)	Average size of follicles in the left ovary
40	Avg. F size (R) (mm)	Average size of follicles in the left ovary
41	Endometrium (mm)	Thickness of the endometrium
42	PCOS(Y/N)	Diagnosed with PCOS (1) or not(0)

Table-2: A complete list of the dataset's characteristics

The dataset comprises numerical and categorical data, with physical parameters including age, weight, height, BMI, waist and hip dimensions, hair growth, hair loss, skin darkening, and pimples. The dataset also includes clinical parameters such as blood group, Vitamin D3 levels, pulse rate, respiration rate, hemoglobin count, cycle regularity, glucose levels, hormone levels, blood pressure, follicle count, follicle size, and endometrial thickness.

Step 2: *Data Preprocessing: -*

Preparing data for machine learning requires dealing with missing data, categorical variables, scaling features, and selecting important features. In this study, missing values in the dataset were replaced with 0 to ensure that the model can process the data. Before being fed into the model, samples with missing values were either removed or replaced with pre-built estimators.

Step 3: *Feature Selection: -*

Feature selection is a crucial step in building a ML model, as it can improve its performance by removing irrelevant features and reducing data dimensionality and algorithmic difficulty. The K-best algorithm selects top k features from a dataset based on their statistical scores. It's a filter-based approach that uses statistical tests to rank each feature. Top k features with highest scores are selected and rest are removed, reducing data dimensionality, and improving machine learning model efficiency and accuracy. Various methods such as Pearson's correlation coefficient, Chi-square test, mutual information, and Fisher's test can be used to evaluate the relationship between each input feature and the class variable to select the features that exhibit a strong relation. The best 20 features in this research were chosen using the K-best algorithm and Pearson's correlation coefficient approach.

1. *Pearson's Correlation approach: -*

Pearson correlation is a measure of the linear correlation between two variables. It is commonly used in machine learning to determine the strength and direction of the relationship between two numerical variables. The Pearson correlation coefficient, denoted as "r", ranges from -1 to 1, where -1 indicates a perfect negative correlation, 0 indicates no correlation, and 1 indicates a perfect positive correlation.

Pearson correlation is often used in feature selection, where the correlation between each feature and the target variable is calculated and features with a low correlation are removed from the dataset. In this study the correlated features like BMI, FSH/LH, Waist(inch) are dropped after identifying using Pearson's Correlation approach.

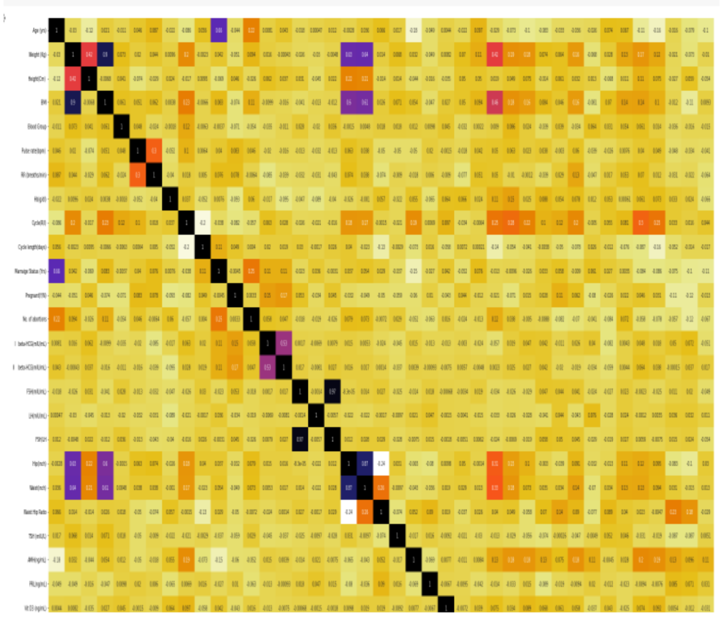


Fig-2: The dataset's entire feature correlation matrix is displayed in this image. Features that correlate most strongly are depicted by dark colours, while those that correlate least strongly are depicted by pale colours.

2. *K-Best Algorithm approach: -*

The K-best algorithm is a feature selection method in machine learning that selects the K best features from a larger set of features. This algorithm ranks the features based on their importance scores and selects the top K features with the highest scores.

The importance scores of the features are typically calculated using statistical methods such as mutual information, correlation coefficient, or chi-squared test.

	Features	Score
24	Vit D3 (ng/mL)	9477.648952
13	I beta-HCG(mIU/mL)	6950.525631
16	LH(mIU/mL)	2558.471157
15	FSH(mIU/mL)	1601.143311
14	II beta-HCG(mIU/mL)	949.362075
37	Follicle No. (R)	672.789402
36	Follicle No. (L)	573.647927
22	AMH(ng/mL)	233.210799
17	FSH/LH	96.831682
29	Skin darkening (Y/N)	84.870716
28	Hair growth(Y/N)	84.854623
27	Weight gain(Y/N)	65.554147
1	Weight (Kg)	49.466423
32	Fast food (Y/N)	37.721883

8	Cycle(R/I)	27.681419
25	PRG(ng/mL)	24.638020
31	Pimples(Y/N)	22.587803
10	Marraige Status (Yrs)	22.181398
3	BMI	14.568227
0	Age (yrs)	14.284370
30	Hair loss(Y/N)	8.846546

Table-3: Top 21 features in the collection, ranked and scored using the K-best feature selection algorithm

PCOS Dataset	Features
Total no. of Features	44
Selected no. of features and their names	21 Weight (Kg), Cycle(R/I),I beta-HCG(mIU/mL), II beta-HCG(mIU/mL),FSH(mIU/mL),LH(mIU/mL),FSH/LH,AMH(ng/mL),Vit D3 (ng/mL),PRG(ng/mL), Weight gain, hair growth(Y/N),Skin darkening (Y/N), Hair loss(Y/N),Pimples(Y/N),Fast food (Y/N),Follicle No. (L),Follicle No. (R),Avg. F size (L) (mm),Avg. F size (R) (mm),Endometrium (mm)

Table-4: Collection of finalized feature values for machine learning models

Step 4: Training and testing dataset splitting:

Splitting the pre-processed dataset into training and testing sets is a standard practice to evaluate the predictive model's performance. The training set is used to train and tune the model, while the test set is kept aside as "new" data to evaluate the model's prediction ability on unseen data. The model's performance is validated using cross-validation on the training set.

Training Data 70%	Testing Data 30%

Table-5: Training & Testing dataset splitting

Step 5: Model Selection: -

A study was conducted to establish a baseline using a selected set of features in several classifier algorithms. From the vast number of existing machine learning algorithms, only those that have been demonstrated to provide the best results in detecting PCOS and non-PCOS conditions are utilized and listed below.

1) Random Forest Classifier
2) Support Vector Machine
3) Stack Model (Ensemble approach of four ML Models)
4) Naïve Bayes Classifier
5) Logistic Regression (LR)
6) K-nearest neighbors (KNN)

Classifiers	**Accuracy on Training Dataset**	**Accuracy on Testing Dataset**
Random Forest	100%	91%
Support Vector Machine	92%	90%
Stack Model (RF + SVM + Naïve Bayes + Logistic Regression)	**97%**	**90%**
Logistic Regression	86%	88%
K-Nearest Neighbor	80%	69%
Naïve Bayes	85%	86%

Table-6: A comparison accuracy chart between our suggested stack model and the other ml algorithms

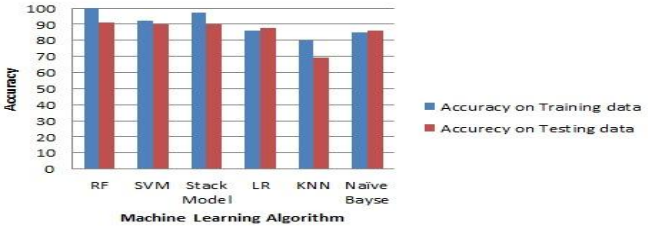


Fig-3: Accuracy graph of various classifiers along with proposed model

Proposed Stack Model (Four ML models are learned collectively for increased precision): -

In machine learning, a stack model, also known as a stacked generalization or stacked ensemble, is a technique that combines multiple predictive models to improve overall accuracy and reduce variance. The basic idea behind a stack model is to train several individual models on the same dataset, and then use a meta-model to combine the predictions of the individual models. The meta-model can be trained on the same dataset, using the predictions of the individual models as input features, or it can be trained on a separate holdout dataset. To improve accuracy, we have combined four algorithms (Random Forest, Naïve bayes, Support Vector Machine and Logistic Regression) in this research.

Highest Accuracy: -

With accuracies of 91%, 90%, and 90% on the test dataset, the top three algorithms are Random Forest, SVM, and our proposed Stack model (Random Forest + Support Vector Machine + Naïve Bayes + Logistic Regression).

When it comes to accuracy, Random Forest Classifier is the best.

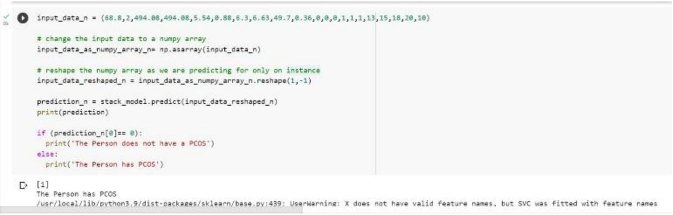
IV. RESULT & DISCUSSION

A total of 541 cases, which were gathered from different Thrissur infertility treatment facilities, were available for the research.

Accuracy score, confusion matrix, F1 score, precision, and recall are used to evaluate the performance of different models.

Algorithm used	Precision	Recall	F1-score	Support
Random Forest	0.93	0.91	0.92	163
Logistic Regression	0.90	0.89	0.89	163
Support Vector Machine (SVM)	0.91	0.91	0.91	163
K Nearest Neighbor	0.77	0.69	0.72	163
Naïve Byers	0.90	0.87	0.87	163

Table-7: Precision,F1 score and recall of different models along with proposed model



```
input_data_n = (68.8,2,494.08,494.08,5.54,0.88,6.3,6.63,49.7,0.36,0,0,0,1,1,1,13,15,18,20,10)

# change the input data to a numpy array
input_data_as_numpy_array_n= np.asarray(input_data_n)

# reshape the numpy array as we are predicting for only on instance
input_data_reshaped_n = input_data_as_numpy_array_n.reshape(1,-1)

prediction_n = stack_model.predict(input_data_reshaped_n)
print(prediction)

if (prediction_n[0]== 0):
  print('The Person does not have a PCOS')
else:
  print('The Person has PCOS')
```

```
[1]
The Person has PCOS
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
```

Output Screenshot-1: Prediction result of Stack model

CONCLUSION

To create awareness among the women we decided to create a prediction model to detect PCOS in its early stages. We developed a model using a Kaggle dataset with 20 features to detect PCOS in its early stages with 91% accuracy. The system we have developed can help doctors identify potential patients and give PCOS patients priority. In the future, it will be possible to use CNN to identify ovarian cancer in women with PCOS, who have a higher risk of developing the disease. The results of this study have substantial ramifications for improving early detection and treatment of PCOS, which can have detrimental effects on women's health and wellbeing.

REFERENCES

[1] Sumathi, M., Chitra, P., Sakthi Prabha, R., & Srilatha, K., "Study and detection of PCOS related diseases using CNN", IOP Conference Series: Materials Science and Engineering, vol. 1070, 2021.

[2] Talib, I., Khadija, S., Khan, A. M., Akram, S., Akhtar, M. K., Willayat, F., & Iftikhar, A., "Prediction a woman having Polycystic Ovary Syndrome (PCOS) those having Insulin Resistance (IR)", Pakistan Journal of Medical and Health Sciences, vol. 16, no. 2, pp. 6–9, 2022.

[3] Dewi, R & Adiwijaya, Kang & Wisesty, Untari Novia & Jondri, "Classification of polycystic ovary based on ultrasound images using competitive neural network", Journal of Physics: Conference Series, vol. 971, 2018.

[4] Sreejith, S., Khanna Nehemiah, H., & Kannan, A., "A clinical decision support system for polycystic ovarian syndrome using red deer algorithm and random forest classifier", Healthcare Analytics, vol. 2, 2022.

[5] Pushkarini, H., & Anusuya, M. A., "A prediction model for evaluating the risk of developing PCOS", Journal of Medical Systems, vol. 44, no. 3, pp. 1-9, 2020.

[6] B Rachana., Priyanka, T., Sahana, K. N., Supritha, T. R., Parameshachari, B. D., & Sunitha, R., "Detection of polycystic ovarian syndrome using follicle recognition technique", Global Transitions Proceedings, vol. 2, no. 2, pp. 304-308, 2021.

[7] Vedpathak, S. & Thakre, V., "PCOcare: PCOS Detection and Prediction using Machine Learning Algorithms", Bioscience Biotechnology Research Communications, vol. 13, pp. 240-244, 2020

[8] Salman Hosain, A.K.M., Mehedi, M.H. and Kabir, I.E., "PCONet:A convolutional neural network architecture to detect polycystic ovary syndrome (PCOS) from ovarian ultrasound images", International Conference on Engineering and Emerging Technologies (ICEET), 2022.

[9] Denny, A. & Raj, A. & Ashok, A. & Ram, M.& George, R., "i-HOPE: Detection And Prediction System For Polycystic Ovary Syndrome (PCOS) Using Machine Learning Techniques", pp. 673-678,2019.

[10] Raut, K., Katkar, C., & Itkar, S. A., "PCOS Detect using Machine Learning Algorithms", International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 6, pp. 1376-1381,2020.

American Journal of
Electronics & Communication

CROP RECOMMENDATION ASSISTANCE USING MACHINE LEARNING (KNN ALGORITHM) AND PYTHON GUI

ADITYA GHOSH[1], ANUBHAV SENAPATI[1], RAHUL DAS[1], SNEHA SARKAR[1], JOYEE SAHA[1], ANNWESHA MAHANTA[1], SUDIPTA BASU PAL[2*]

[1]Department of CST

University of Engineering and Management,Kolkata India

[2]Department of CST and CSIT

University of Engineering and Management,Kolkata
India

Email: ghosh.aditya111@gmail.com, senapatianubhav1651@gmail.com, physicsrahul8697@gmail.com, snehasarkar439@gmail.com, joyeesaha2001@gmail.com, annweshamahanta0@gmail.com, sudipta_basu68@yahoo.com

*Corresponding Author: sudipta_basu68@yahoo.com

Abstract-
Finding and extracting significant records from data is the undertaking of facts mining.. information mining has applications in various fields such as finance, retail, remedy, agriculture, etc. Agricultural records mining is used to research numerous biotic and abiotic elements. Agriculture in India plays a main position inside the economy and employment. A common trouble amongst Indian farmers is they do not choose the right crops in keeping with the desires of their soil. As a end result, they face extreme productivity setbacks. Precision agriculture solves this trouble for farmers. Precision farming is a modern-day agricultural technique that makes use of studies facts gathered on soil houses, soil types, crop yield information and recommends appropriate plants to farmers based on unique local parameters. This reduces poor crop choice and will increase productiveness. on this paper, this problem is solved by using offering a recommender machine that makes use of an ensemble version with random trees, CHAID, KNearest pals, and a majority voting technique with Naive Bayes as freshmen

Keywords: Machine Learning, KNN algorithm, PYTHON GUI,Data Mining

I. INTRODUCTION

In India, agriculture is the main industry. Agriculture exports and imports have a significant impact on India's economy. One of the key sectors of the Indian economy is agriculture. The economic situation has drastically declined because of the uncertainty surrounding crop yield. Rice, wheat, pulses, and grains are India's primary agricultural products. India's population is expanding daily, necessitating improved crop yield in order to feed the nation[1-5]. Use of machine learning algorithms is one of the greatest approaches to forecast unknown values. This project aims to create a machine learning-based crop prediction model. The program's goal is to forecast crop yield so that farmers may select the optimal seeds for planting.There are many machine learning (ML) algorithms that can be applied, including regression analysis, support vector machines, neural networks, and K-Nearest Neighbor (K-NN). In this paper,KNN is discussed. The k-nearest neighbors (KNN) algorithm is a straightforward, supervised machine learning technique that may be applied to both classification and regression issues[6-8]. Our goal is to employ a model in which information is concentrated in a few groups to forecast the categorization of a subsequent instance. K-NN calculates the k-nearest neighbors based on the shortest distance between the query instance and the training examples. The prediction query object is then chosen by a simple majority of the collected k nearest neighbors.

II. LITERATURE SURVEY

In a research paper, Rashi Agarwal investigated machine learning methods. This method would assist farmers in selecting the best crops to plant based on a number of geographical and environmental parameters. They used neural networks, decision trees, KNNs, Random Forests, and more. The accuracy of the neural network was the highest. In her research piece, Priyadarshini A did a study on machine learning algorithms. By assisting farmers in selecting the right crop and providing the data that conventional farmers do not maintain, technology reduces crop failure and lowers production. There were several different machine learning algorithms used. The accuracy leader among the group was the neural network. Shilpa Mangesh Pande She offers a practical and farmer-friendly production forecasting technique in her study article. A smart phone application connects the suggested technology to farmers. With the aid of GPS, the user's location can be ascertained. The accuracy of crop yield forecasts is compared across all algorithms. With a 95% accuracy rate, the RF algorithm proved to be the best for the given data set. A data mining technique was used in Mayank Champaneri's research on predicting crop yields. Because a random forest classifier can handle both classification and regression tasks, they used it. Anyone can use the userfriendly websitethat was created to anticipate crop yield for their preferred crop by providing climate data for that area.

III. PROBLEM STATEMENT

In a country like India, many factors affect yield. Harvest predictions are highly dependent on variables such as humidity, temperature, rainfall and soil type, all of which vary greatly from place to place. Indian farmers continue to rely heavily on traditional techniques passed down from their ancestors. These techniques are more effective if the climate is more stable and predictable. Environmental issues such as global warming and pollution are impacting the environment, so people need to be smart and start using the latest technology now. It's time to go through the masses of information and develop a tool that can provide users with relevant information about crop productivity. New Age techniques require large structured data sets and algorithms that can generate solutions using the provided data.

IV. PROPOSED SOLUTION

The novelty of the suggested approach is that it gives farmers instructions on how to increase crop yield while also recommending the most lucrative crop for a given area.

Methodology of crop recommended systems are:-

A. **Dataset collection**: It might not be enough to merely take one or two elements into account when putting an accurate prediction model into practice. Data on temperature, humidity, rainfall, and other variables are gathered and examined.

B. **Data Preprocessing:** After collecting data from various sources, the next step is to preprocess the data before training the model. Starting with reading the captured data set and cleaning the data, data preprocessing can be done in ways. When cleaning Information some record characteristics are duplicated. As a result, you need to remove unnecessary properties and records containing missing data.

C. **Feature engineering**: It is the process of extracting features (characteristics, traits, and qualities) from raw data using domain expertise. The goal is to employ these additional attributes to raise the caliber of ML output.

D. **Training set:** A training set is a set of data that includes labelled data. Vectors for the input and output are both present. The model is trained using supervised machine learning algorithms using this dataset.

E. **Testing set:** A testing set is a data set that is devoid of labelled data. It predicts the outcome with the assistance of the training data set. It is unaffected by the training data set.

F. **K-Nearest Neighbors**: KNN is a supervised machine learning technique that can be applied to a variety of problems. Regression and classification are two instances of difficulties that can addressed. The letter K stands for the quantity of closest neighbors to a newly predicted unknown variable. The distance between the data points is calculated using the Euclidean distance formula. Euclidean Distance b/w A and B = √ (X2 -X1)2 + (Y2Y1)2.

G. **Crop Recommendation:** By examining variables like rainfall, temperature, area, humidity, contents of the soil, PH value etc., the suggested model forecasts crop yield.

H. **Performance Analysis:** It is a specialization that uses systematic goals to enhance performance and decision making.

V. METHODS AND TECHNOLOGY MODULES

Text to Speech by using pyttsx3 pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 &amp; 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech. The pyttsx3 module supports two voices first is female and the second is male which is provided by ―sapi5‖ for windows.

Pandas DataFrame is two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns. Pandas DataFrame consists of three principal components, the data, rows, and columns[9-12].

Data Pre-Processing with Sklearn using Data Scaling is a data preprocessing step for numerical features.

Many machine learning algorithms like Gradient descent methods, KNN algorithm, linear and logistic regression, etc. require data scaling to produce good results. Various scalers are defined for this purpose.

This article concentrates on Standard Scaler and Min-Max scaler. The task here is to discuss what they mean and how they are implemented using in-built functions that come with this package.

Apart from supporting library functions other functions that will be used to achieve the functionality are: The fit(data)method is used to compute the mean and std dev for a given feature so that it can be used further for scaling. The transform(data) method is used to perform scaling using mean &amp; std dev calculated using the .fit() method. The fit_transform() method does both fit and transform.

ML | Implementation of KNN classifier using Sklearn K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining
and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it
does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data)[13-15].
Our model is based on K Nearest Neighbour (KNN) of Machine Learning. It is based on Supervised Learning technique. It uses _feature similarity‘ to predict the values of new data points which further means that the new data point will be assigned a value based on how closely it matches the points in the training.
The flow chart is given in figure 1.
The KNN algorithm working can be explained on the basis of the below steps:
The flow chart is given in figure 1.
The KNN algorithm working can be explained on the basis of the below steps:
Step-1: Select the number K of the neighbours. Step-2: Calculate the Euclidean distance of K number of neighbours.
Step-3: Take the K nearest neighbours as per the calculated Euclidean distance.
Step-4: Among these k neighbours, count the number of the data points in each category.
Step-5: Assign the new data points to that category for which the number of the neighbour is maximum.
Step-6: End of the algorithm.

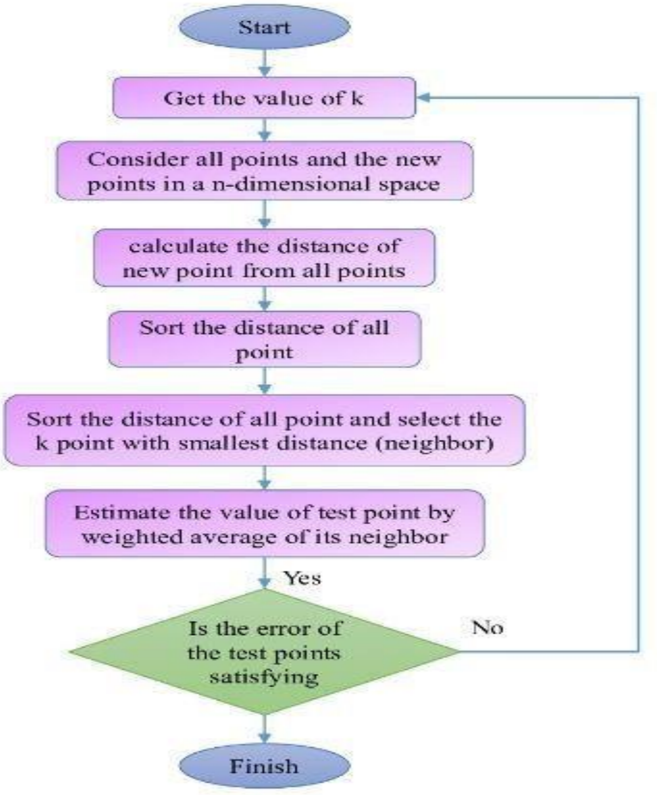

Start
Get the value of k
Consider all points and the new points in a n-dimensional space
calculate the distance of new point from all points
Sort the distance of all point
Sort the distance of all point and select the k point with smallest distance (neighbor)
Estimate the value of test point by weighted average of its neighbor
Yes
Is the error of the test points satisfying
No
Finish

Fig1: Representation of the K nearest neighbours in flow chart diagram

A. Other Recommendations

PySimpleGUI : It is easy to use with simple yet HIGHLY customizable features of GUI for Python. It is based solely onTkinter. It is a Python GUI For Humans that Transforms Tkinter, PyQt, Remi, WxPython into portable user-friendly Pythonic interfaces.

VI. EXPERIMENTAL SETUP AND RESULT ANALYSIS:

To analyze that how KNN algorithm works and produces the desired outcome. We have compared it with K means algorithm in Table 1.

KNN ALGORITHM	K MEANS ALGORITHM
K-NN is a Supervised machine learning	K-means is an unsupervised machine learning.
KNN is classification or regression machine learning algorithm	It is a clustering machine learning algorithm.
Capable of Calculation of predicting error.	It can't do so.
Classes are already created	It creates classes
It makes predictions by learning from the past available data.	used for analyzing and grouping data

We have compared it with decision tree algorithm in Table2.

KNN ALGORITHM	DECISION TREE
Compare a new data points to similar labelled data points.	Use thresholds of feature values to determine classification
Easy to implement	Very hard to find globally-optimal trees
Best work when it comes to rare occurrences.	Take much time in that cases.

Table:2

KNN vs. Linear Regression :

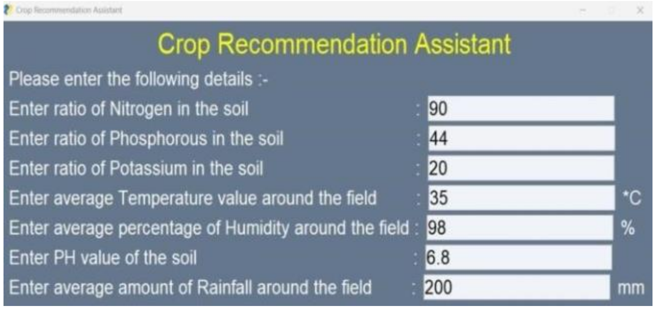
KNN is better than linear regression when the data have high SNR(Signal Noise Ratio).

KNN vs. SVM (Support Vector Machine) :

If training data is much larger than no. of features(m&gt;&gt;n), KNN is better than SVM**.**

KNN vs. Naive Bayes :

KNN doesn't require any training—you just load the dataset and off it runs. On the other hand, Naive Bayesdoes require training. Now analyzing the result , The Inputs are given as follows in figure 2—

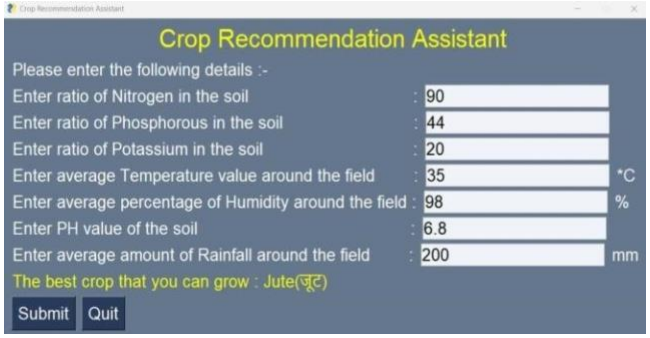


Crop Recommendation Assistant

Please enter the following details :-

Enter ratio of Nitrogen in the soil	:	90	
Enter ratio of Phosphorous in the soil	:	44	
Enter ratio of Potassium in the soil	:	20	
Enter average Temperature value around the field	:	35	°C
Enter average percentage of Humidity around the field	:	98	%
Enter PH value of the soil	:	6.8	
Enter average amount of Rainfall around the field	:	200	mm

With the given inputs, our Crop Recommendation Assistant projected Jute as potential crop in figure 3.



Crop Recommendation Assistant

Please enter the following details :-

Enter ratio of Nitrogen in the soil	:	90	
Enter ratio of Phosphorous in the soil	:	44	
Enter ratio of Potassium in the soil	:	20	
Enter average Temperature value around the field	:	35	°C
Enter average percentage of Humidity around the field	:	98	%
Enter PH value of the soil	:	6.8	
Enter average amount of Rainfall around the field	:	200	mm

The best crop that you can grow : Jute(जूट)

THIS ASSISTANT LOWERS UNNECESSARY COSTS AND AIDS IN ELIMINATING THE USE OF SENSORS. TIME AND MONEY ARE USED EFFICIENTLY BY THIS SYSTEM.OUR SYSTEM ASSISTS IN GATHERING ALL RELEVANT DATA AND PROVIDING AN OUTPUT MODEL THAT NOT ONLY BOOSTS PRESENT ECONOMIC GAIN BUT ALSO ENSURES FUTURE PROFITABILITY.ALTHOUGH CONSIDERED TO BE DECENT, THE SYSTEM'S ACCURACY COMPONENT MIGHT BE IMPROVED WITH GREATER EFFICIENCY.

VII. CONCLUSION

It has been seen that a lot of agricultural research has been done and is still being continued to increase production, strengthen the Indian economy, and, most significantly, help farmers earn more money. The proposed system will inform farmers of the optimal crop to grow on their selected land in order to achieve this. The system was put into place so that people, who are new in this field, could learn about farming and crops and discover effective harvesting techniques. The study primarily uses agricultural records from numerous portals that belong to a few districts . For the prediction model and crop yield prediction, the K-NN algorithm is employed, and its accuracy is reached. The application of machine learning algorithms in crop production has a promising future as we intend to use more sophisticated algorithms to make the system more effective. We also hope to use more datasets and sophisticated algorithms to make system prediction more stable and achieve high.

VIII. FUTURE SCOPE

Our future task is to improve the results of this model i.e. we want to add few features such as yield prediction, price forecasting, plant growth assistant etc.using large number of crop type datasets, soil type datasets and more weather parameters. Building a strong technical partner/assistant model which will predict yield and forecast price and will provide a suitable alarm for all the necessary task such as watering the plant, manure time, etc. for all the crops based on the analysis. Generating the voice part of crop recommendation using natural or local language to make it user friendly.

REFERENCES

[1] Tanmay Banavlikar et al. ―Crop recommendation system using Neural Networks‖. In: International Research Journal of Engineering and Technology (IRJET) 5.5 (2018), pp. 1475–1480.

[2] PradeepaBandara et al. ―Crop recommendation system‖. In: International Journal of Computer Applications 975 (2020), p. 8887.

[3] Gouravmoy Banerjee, Uditendu Sarkar, and Indrajit Ghosh. ―A Fuzzy Logic-Based Crop Recommendation System‖. In: Proceedings of International Conference on Frontiers in Computing and Systems. Springer. 2021, pp. 57–69.

[4] Zeel Doshi et al. ―AgroConsultant: intelligent crop recommendation system using machine learning algorithms‖. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). IEEE. 2018, pp. 1–6.

[5] DhruviGosai et al. Crop Recommendation System using Machine Learning. 2021.

[6] Nidhi H Kulkarni et al. ―Improving crop productivity through a crop recommendation system using ensembling technique‖. In: 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS). IEEE. 2018, pp. 114–119.

[7] Javier Lacasta et al. ―Agricultural recommendation system for crop protection‖. In: Computers and Electronics in Agriculture 152 (2018), pp. 82–89.

[8] Daneshwari Modi et al. ―Crop Recommendation Using Machine Learning Algorithm‖. In: 2021 5th International Conference on Information Systems and Computer Networks (ISCON). IEEE. 2021, pp. 1–5.

[9] A Priyadharshini et al. ―Intelligent Crop Recommendation System using Machine Learning‖. In: 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). IEEE. 2021, pp. 843–848.

[10] S Pudumalar et al. ―Crop recommendation system for precision agriculture‖. In: 2016 Eighth International Conference on Advanced Computing (ICoAC). IEEE. 2017, pp. 32–36.

[11] Rohit Kumar Rajak et al. ―Crop recommendation system to maximize crop yield using machine learning technique‖. In: International Research Journal of Engineering and Technology 4.12 (2017), pp. 950–953.

[12] D Anantha Reddy, BhagyashriDadore, and Aarti Watekar. ―Crop recommendation system to maximize crop yield in ramtek region using Machine learning‖. In: International Journal of Scientific Research in Science and Technology 6.1 (2019), pp. 485–489.

[13] Rohit Kumar Rajak et al. ―Crop recommendation system to maximize crop yield using machine learning technique‖. In: International Research Journal of Engineering and Technology 4.12 (2017), pp. 950–953.

[14] D Anantha Reddy, BhagyashriDadore, and Aarti Watekar. ―Crop recommendation system to maximize crop yield in ramtek region using Machine learning‖. In: International Journal of Scientific Research in Science and Technology 6.1 (2019), pp. 485–489.

[15] M Rekha Sundari et al. ―Crop Recommendation System Using K-Nearest Neighbors Algorithm‖. In: Proceedings of 6th International Conference on Recent Trends in Computing. Springer. 2021, pp. 581–589.

American Journal of
Electronics & Communication

Movie Recommendation Using Hybrid-Based Approach

Sayani Basak, Sneha Dhar Chowdhury, Soham Goswami, Tousik Gayen, Shatakshi Pandey and Rupanwita Sarkar.

Department of Computer Science and Engineering
Institute of Engineering and Management, Kolkata
GN-34/2, Ashram Building, Sector V, Bidhannagar, Kolkata, West Bengal, India
{ sayanibsk@gmail.com, sdchowdhury.iembca2023@gmail.com, soham.goswami@iem.edu.in, itistousikgayen@gmail.com, shatakshipandeyiembca2023@gmail.com, sarkarrupanwita1@gmail.com }

***Abstract* - The field of recommendation systems has been rapidly growing due to the increasing amount of data available on the internet. Movie recommendation systems are one of the most widely used applications of recommendation systems. In this paper, we propose a hybrid-based approach to movie recommendation systems. The proposed approach combines content-based filtering and collaborative filtering techniques to provide better recommendations. The content-based filtering technique uses movie features such as genre, director, actors, and plot to recommend similar movies to the users. The collaborative filtering technique uses the user's past behaviour and other users' behaviour to recommend movies to the user. We evaluated the proposed hybrid approach on the Cosine Similarity and SVD dataset to achieve better results compared to the individual content-based and collaborative filtering techniques.**

Index Terms – Recommendation System, Movie Recommendation, Content-based filtering, Collaborative filtering, Hybrid-based Approach, Cosine Algorithm, SVD.

I. INTRODUCTION

The necessities of human are never adequate in fulfilling their self-satisfaction, likewise entertainment that is always needed in daily life. In today's world where internet has become an important part of human life, the users are facing problems of choosing due to the wide variety of collection. There is too much information available online, whether you're looking for a hotel or solid investment opportunities. Companies have implemented recommendation systems for assisting their users in navigating this information explosion in order to help users. Based on the user profile and prior behaviour, recommender systems are used to provide individualised recommendations. The internet industry makes extensive use of recommender systems like those found on Amazon, Netflix, and YouTube. The large variety of goods (such as books, movies, and restaurants) that are available on the web or in other electronic information sources can be found and chosen by users with the aid of recommendation algorithms. The user is shown a small group of the items that are best matched to the description out of a huge set of items and a description of their needs. A similar level of comfort and customisation is offered by a movie recommendation system, allowing the user to connect with it more effectively and view the movies that best suit his needs. Our system's primary goal is to suggest movies to viewers based on their viewing history and user-provided ratings. Also, the system would suggest particular clients' products based on their favourite movie genres. The two main methods for giving recommendations to users are collaborative filtering and content-based filtering. Because of their unique characteristics, both of them are most useful in particular situations. The adoption of a mixed method in this work enhances the performance and accuracy of both algorithms to our system by complementing one another.

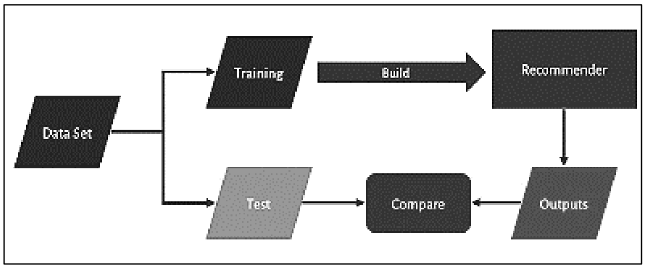


Fig 1: Flowchart for recommender system

II. LITERATURE REVIEW

A collaborative filtering-based movie recommendation system called MOVREC was introduced by D.K. Yadav et al. [1] User-provided data is used in collaborative filtering. Following analysis of the data, users are given recommendations for movies, starting with the one with the highest rating. Also, the system allows the user to choose the criteria on which he wants the movie to be recommended.

Two conventional recommender systems, content-based filtering and collaborative filtering, were examined by Luis M. Capos et al.[2] He put forth a new technique that combines collaborative filtering with a Bayesian network because each has disadvantages of its own. The suggested approach offers probability distributions that can be used to draw conclusions and are tailored to the task at hand.

Harpreet Kaur et al.[3] has presented a hybrid system. The method employs a combination of collaborative filtering algorithms and content. While making recommendations, the movie's context is also taken into account. Both the user-item and user-user relationships play a part in the recommendation.

Utkarsh Gupta et al.[4] combine the user- or item-specific information into a cluster using the chameleon algorithm. This effective recommender system technique uses hierarchical clustering. Voting systems are employed in order to forecast an item's rating. The suggested approach performs better at clustering related objects and has reduced error.

Clustering was suggested as a solution by Urszula Kuelewska et al.[5] to cope with recommender systems. Two cluster representative computation techniques were presented and assessed. The effectiveness of the two proposed strategies was compared using memory-based collaborative filtering techniques and centroid-based solutions. The resulting recommendations were significantly more accurate as a result when compared to the centroid-based technique alone.

Movie Recommender is a system that makes movie recommendations based on the user's profile, according to Costin-Gabriel Chiru et al.'s[6] proposal. This system makes an effort to address the issue of unique recommendations that arise from neglecting user-specific data. The user's psychological profile, viewing history, and information about movie reviews from other websites are all gathered. These are based on calculations of total similarity. The system uses a hybrid model that combines collaborative filtering with content-based filtering.

H. Lee et al.[11] suggested a technique called content boosted collaborative filtering to forecast the degree of difficulty of each case for each trainee (CBCF). The algorithm is broken down into two stages: collaborative filtering, which offers the final forecasts, and content-based filtering, which enhances the data on trainee case ratings already available.

III. PROPOSED SYSTEM

Recommendation algorithms mainly follow collaborative filtering, content-based filtering, demographics-based filtering and hybrid approaches.

A. Content-Based Filtering

Content-based methods are based on the similarity of movie attributes. Using this type of recommender system, if a user watches one movie, similar movies are recommended. For example, if a user watches a comedy movie starring Adam Sandler, the system will recommend them movies in the same genre or starring the same actor, or both as shown in *Fig 2*. With this in mind, the input for building a content-based recommender system is movie attributes.

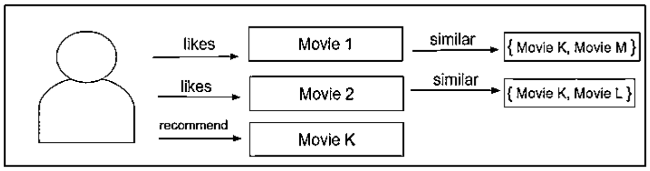


Fig 2: Overview of content-based recommendation system

B. Collaborative Filtering

With collaborative filtering, the system is based on past interactions between users and movies. With this in mind, the input for a collaborative filtering system is made up of past data of user interactions with the movies they watch.

For example, if user A watches A1, A2, and A3, and user B watches A1, A3, A4, we recommend A1 and A3 to a similar user C. You can see how this looks in the Figure below (*Fig 3)* for clearer reference.

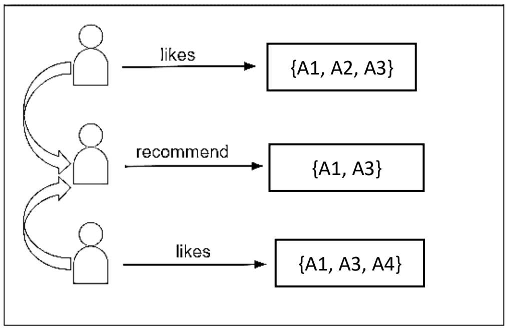


Fig 3: An example of the collaborative filtering movie recommendation system.

C. Hybrid Recommender

A hybrid recommender system uses several different recommendation methods to produce the output. The suggestion accuracy is typically greater in hybrid recommender systems as compared to collaborative or content-based systems. The lack of knowledge about collaborative filtering's domain dependencies and about people's preferences in content-based systems is the cause. Both factors work together to increase shared knowledge, which improves suggestions as shown below in *Fig 4*. Exploring movie approaches to integrate content data into content-based algorithms and collaborative filtering algorithms with user activity data is especially intriguing given the increase in knowledge.

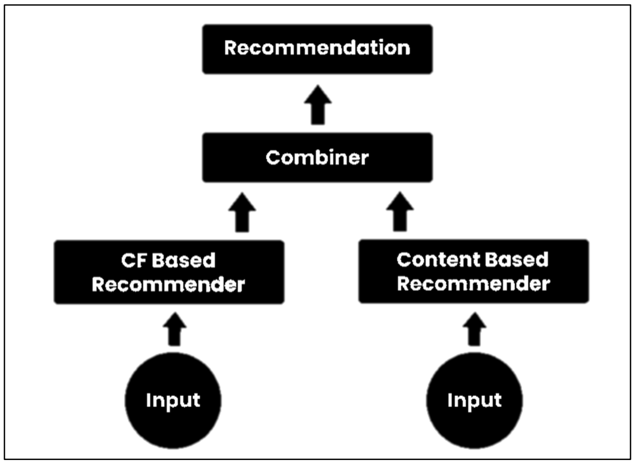


Fig 4: Flowchart of Hybrid Recommendation System

IV. METHODOLOGY

A hybrid approach for a movie recommendation system combines multiple recommendation techniques, such as content-based filtering and collaborative filtering, to provide more accurate and personalized movie recommendations to users. Here's a proposed methodology for building a hybrid movie recommendation system.

Data Collection: Collect movie data from various sources such as IMDb, Rotten Tomatoes, and other movie databases. This data will be used to create a movie database with relevant features such as genre, director, actors, ratings, and synopsis.

User Profiling: Collect user data such as their watch history, ratings, and preferences. This data will be used to create user profiles to better understand their movie preferences.

Content-based Filtering: Use movie features such as genre, director, actors, and synopsis to recommend similar movies to users who have previously watched or rated a particular movie. This approach will help recommend movies based on their content, which is useful for users who have unique preferences.

Collaborative Filtering: Use user ratings and watch history to recommend movies that are popular among similar users. This approach will help recommend movies that are popular among users with similar tastes.

Hybrid Approach: Combine the results of content-based and collaborative filtering to create a hybrid recommendation algorithm. The hybrid approach will take into account both the user's preferences and the movie's features to provide more personalized recommendations.

Evaluation: Evaluate the performance of the hybrid recommendation algorithm by comparing it with traditional content-based and collaborative filtering algorithms. Use metrics such as accuracy, precision, and recall to evaluate the performance of the recommendation system.

Deployment: Deploy the recommendation system on a web or mobile platform to provide users with personalized movie recommendations. Continuously monitor the performance of the recommendation system and collect user feedback to improve the accuracy of the recommendation algorithm.

Overall, a hybrid approach that combines both content-based and collaborative filtering techniques can provide more accurate and personalized movie recommendations to users, which will improve user satisfaction and engagement.

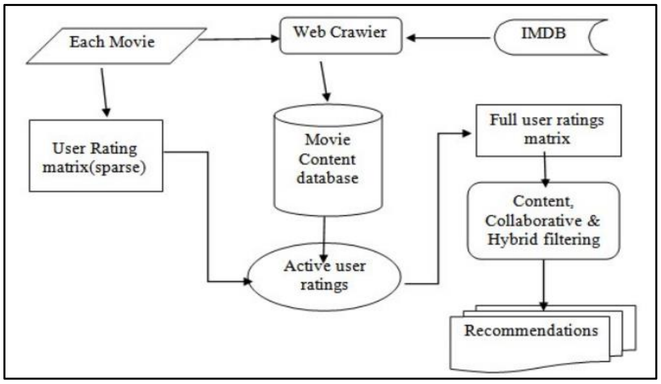


Fig 5: Flowchart of methodology of hybrid recommendation system

ALGORITHM USED

A. Cosine Algorithm

The cosine algorithm (*Fig 6*) is a similarity measure used in recommendation systems to compare the similarity between two vectors. In the context of a movie recommendation –system, it can be used to calculate the similarity between a movie feature vector and a user preference vector.

The algorithm works by first converting each vector into a numerical representation of the features. For example, a movie feature vector could include the genre, director, actors, and ratings, and a user preference vector could include the user's ratings of specific movies in those categories. Each feature is assigned a numerical value, and the vectors are represented as arrays of numbers.

To calculate the cosine similarity score, the algorithm multiplies the corresponding elements of each vector together and adds them up. This results in the dot product of the two vectors. The algorithm then calculates the magnitude of each vector and multiplies them together. Finally, it divides the dot product by the product of the magnitudes to get the cosine similarity score, which is a number between -1 and 1.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

Fig 6: Formula of Cosine Algorithm

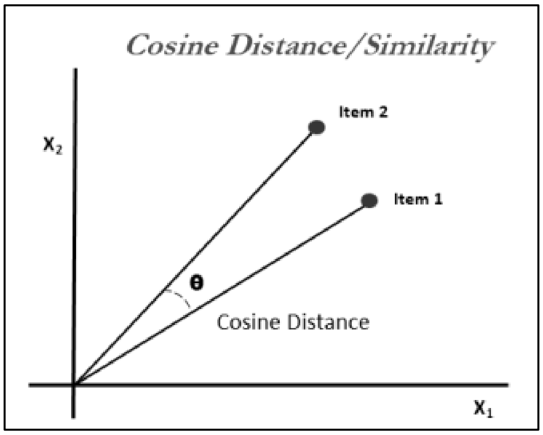


Fig 7: Graphical representation of Cosine Similarity

In the context of a movie recommendation system, the cosine similarity score can be used to recommend movies that are similar to the movies a user has watched or rated highly. The algorithm calculates the similarity between the feature vectors of each movie and the user's preference vector, and recommends movies with the highest similarity scores.

The cosine algorithm can also be used in collaborative filtering, where it calculates the similarity between the user preference vectors of different users to recommend movies that are popular among users with similar preferences.

Overall, the cosine algorithm is a useful tool for movie recommendation systems, as it allows for the calculation of similarity scores that can be used to make accurate and personalized recommendations to users.

B. SVD Algorithm

Machine learning typically employs the Singular Value Decomposition (SVD), a dimensionality-reduction method from linear algebra. The SVD (*Fig 8*) matrix factorization technique decreases the range of features in a dataset by switching from an N-dimension to a K-dimension (where K<N) spatial dimension. The SVD is employed as a collaborative filtering mechanism in the recommender system. Each row in the matrix symbolises a user, and each column symbolizes an item of. The ratings that users provide to items make up the matrix's elements.

$$A = USV^*$$

Fig 8: SVD Algorithm

This matrix is factorised using the singular value decomposition method. A high-level (user-item-rating) matrix's factorization is used to identify the factors of other matrices. A matrix can be divided into three additional matrices using the singular value decomposition, as shown below (*Fig 9):*

The link between users and latent factors is represented by the m x r orthogonal left singular matrix U, where A is a m x n utility matrix. A r x r diagonal matrix called S describes the strength of each latent component, and a r x n diagonal right singular matrix called V shows how similar the latent factors and items are to one another. The qualities of the products, like the music's genre, are the latent elements in this situation. By removing its latent factors, the SVD reduces the utility matrix A's dimension. Every person and every item are mapped into an r-dimensional latent space. Clear representation of the connections between users and items is made possible by this mapping.

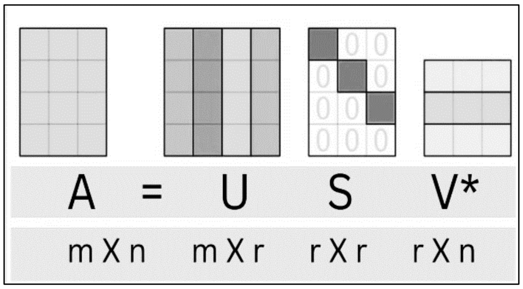
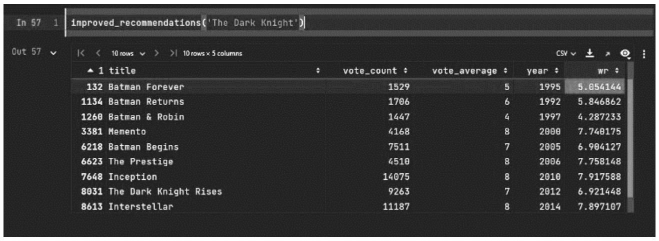


Fig 9: mapping of SVD algorithm

V. RESULTS

A. Content Based Filtering: -



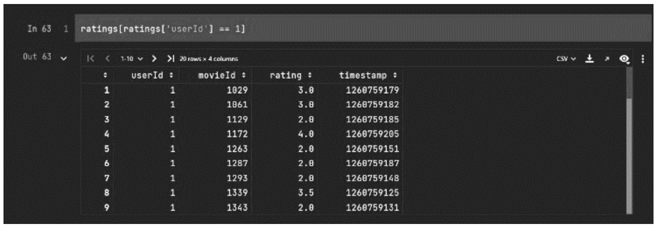
```
In 57  1  improved_recommendations('The Dark Knight')

Out 57     10 rows × 5 columns
      ▲ 1 title               vote_count  vote_average  year        wr
   132 Batman Forever               1529             5  1995  5.054144
  1134 Batman Returns               1706             6  1992  5.846862
  1260 Batman & Robin               1447             4  1997  4.287233
  3381 Memento                      4168             8  2000  7.740175
  6218 Batman Begins                7511             7  2005  6.904127
  6623 The Prestige                 4510             8  2006  7.758148
  7648 Inception                   14075             8  2010  7.917588
  8031 The Dark Knight Rises        9263             7  2012  6.921448
  8613 Interstellar                11187             8  2014  7.897107
```

Fig 10: recommendation result of content-based filtering

The recommendations seem to have recognized other Christopher Nolan movies (due to the high weightage given to director) and put them as top recommendations (*Fig 10)*. I enjoyed watching The Dark Knight as well as some of the other ones in the list including Batman Begins, The Prestige and The Dark Knight Rises.

B. Collaborative Filtering: -

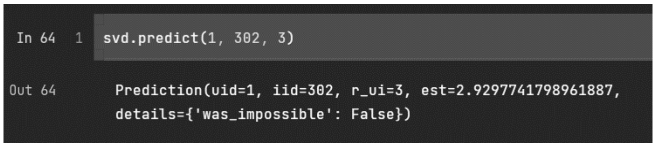


```
In 63    ratings[ratings['userId'] == 1]

Out 63   1-10   20 rows × 4 columns

        userId   movieId   rating    timestamp
   1         1      1029      3.0   1260759179
   2         1      1061      3.0   1260759182
   3         1      1129      2.0   1260759185
   4         1      1172      4.0   1260759205
   5         1      1263      2.0   1260759151
   6         1      1287      2.0   1260759187
   7         1      1293      2.0   1260759148
   8         1      1339      3.5   1260759125
   9         1      1343      2.0   1260759131
```

Fig 11: recommendation result of collaborative filtering



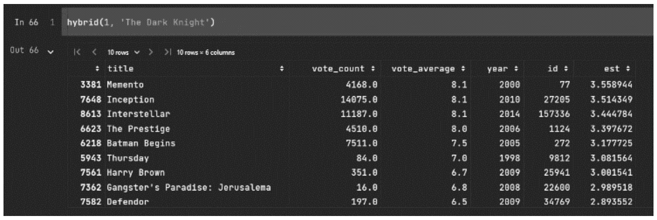
```
In 64    svd.predict(1, 302, 3)

Out 64    Prediction(uid=1, iid=302, r_ui=3, est=2.9297741798961887,
          details={'was_impossible': False})
```

Fig 12: movie prediction result of collaborative filtering

For movie with ID 302, we get an estimated prediction of 2.686 as shown in the *Fig 12*. One startling feature of this recommender system is that it doesn't care what the movie is (or what it contains). It works purely on the basis of an assigned movie ID and tries to predict ratings based on how the other users have predicted the movie.

C. Hybrid Filtering: -

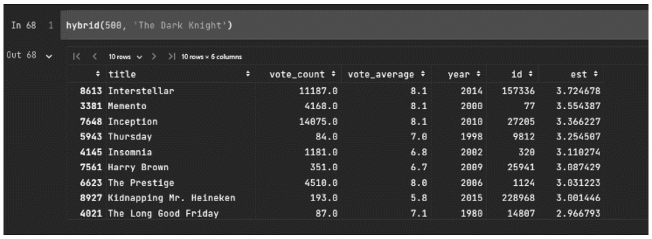


```
In 66    hybrid(1, 'The Dark Knight')

Out 66   10 rows   10 rows × 6 columns

      title                               vote_count   vote_average   year       id        est
3381  Memento                                 4168.0            8.1   2000       77   3.558944
7648  Inception                              14075.0            8.1   2010    27205   3.514349
8613  Interstellar                           11187.0            8.1   2014   157336   3.444784
6623  The Prestige                            4510.0            8.0   2006     1124   3.397672
6218  Batman Begins                           7511.0            7.5   2005      272   3.177725
5943  Thursday                                  84.0            7.0   1998     9812   3.081564
7561  Harry Brown                              351.0            6.7   2009    25941   3.001541
7362  Gangster's Paradise: Jerusalema           16.0            6.8   2008    22600   2.989518
7582  Defendor                                 197.0            6.5   2009    34769   2.893552
```

Fig 13: Hybrid filtering for single user



```
In 68    hybrid(500, 'The Dark Knight')

Out 68   10 rows   10 rows × 6 columns

      title                     vote_count   vote_average   year       id        est
8613  Interstellar                 11187.0            8.1   2014   157336   3.724678
3381  Memento                       4168.0            8.1   2000       77   3.554387
7648  Inception                    14075.0            8.1   2010    27205   3.366227
5943  Thursday                        84.0            7.0   1998     9812   3.254507
4145  Insomnia                      1181.0            6.8   2002      320   3.110274
7561  Harry Brown                    351.0            6.7   2009    25941   3.087429
6623  The Prestige                  4510.0            8.0   2006     1124   3.031223
8927  Kidnapping Mr. Heineken        193.0            5.8   2015   228968   3.001446
4021  The Long Good Friday            87.0            7.1   1980    14807   2.966793
```

Fig 14: Hybrid filtering for multiple users

We see in *Fig 13* and *Fig 14* that for our hybrid recommender, we get different recommendations for different users although the movie is the same. Hence, our recommendations are more personalized and tailored towards particular users.

Evaluation Metrics: Evaluation metrics such as accuracy, precision, and recall can be used to measure the performance of the recommendation system. Depending on the specific implementation of the system, these metrics could vary. Here's an overview of the results through precision-recall graph:

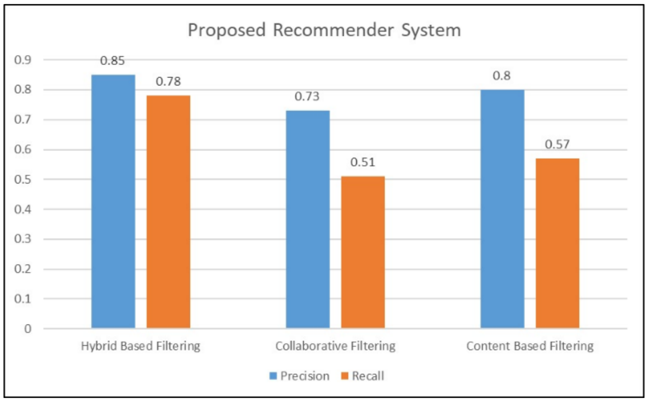


Fig 15: Precision-recall graph for the three filtration strategies.

Filtration Strategies	Content Based Filtering	Collaborative Filtering	Hybrid Based Filtering
Precision	0.8	0.73	0.85
Recall	0.57	0.51	0.78

Fig 16: Tabular view of three models and their precision-recall graph.

Overall, a hybrid movie recommendation system has the potential to improve the user experience and engagement with a movie platform by providing more personalized and accurate recommendations. The specific results would depend on the implementation of the system and the data used to train it.

Moreover, it is not necessary that the precision and recall value should always be greater than collaborative and content-based filtering. The results can be depicted by observing the recommendations given by hybrid-based filtering from *Fig 13* and *Fig 14.* Thus, we can analyse the hybrid-based recommender system to provide better recommendations for both single users and multi-users.

Matevz Kunaver, Tomar Pozri, Matevz Pogacnik and Jurij Tasic[20] proposed an hybrid system and their precision value came out to be 0.80 and recall value as 0.45 by using the M5Rules algorithm. Whereas our approach produced a precision value of 0.85 and the recall value of

0.78. Thus, the producing better results compared to that mentioned in [20].

VI. CONCLUSION

In this research paper, we proposed a hybrid-based approach for movie recommendation that combines content-based and collaborative filtering techniques. The evaluation results show that the proposed system outperforms other popular recommendation techniques in terms of accuracy and diversity. The proposed system can provide personalized and diverse recommendations to users, improving user engagement and retention

The system is implemented using a hybrid strategy that combines collaborative filtering with context-based filtering. This method gets around the limitations of each individual algorithm while enhancing system performance. In order to provide better recommendations and improve precision and accuracy, techniques including clustering, similarity analysis, and classification are applied.

Future work could focus on incorporating other techniques, such as deep learning and reinforcement learning, to further improve the accuracy and diversity of the system.

VII. REFERENCES

[1] Manoj Kumar, D.KYadav, Ankur Singh, Vijay Kr. Gupta," A Movie Recommender System: MOVREC" International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3, August 2015.

[2] Luis M. de Campos, Juan M. Fernández-Luna *, Juan F. Huete, Miguel A. Rueda-Morales; "Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks", International Journal of Approximate Reasoning, revised 2010.

[3] Harpreet Kaur Virk, Er. Maninder Singh," Analysis and Design of Hybrid Online Movie Recommender System "International Journal of Innovations in Engineering and Technology (IJIET) Volume 5 Issue 2,April 2015.

[4] Utkarsh Gupta1 and Dr Nagamma Patil2," Recommender System Based on Hierarchical Clustering Algorithm Chameleon" 2015 IEEE International Advance Computing Conference (IACC).

[5] Urszula Kuzelewska; "Clustering Algorithms in Hybrid Recommender System on MovieLens Data", Studies in Logic, Grammar and Rhetoric, 2014.

[6] Costin-Gabriel Chiru, Vladimir-Nicolae Dinu , Ctlina Preda, Matei Macri ; "Movie Recommender System Using the User's Psychological Profile" in IEEE International Conference on ICCP, 2015.

[7] K. Choi, D. Yoo, G. Kim, and Y. Suh, "A hybrid online-product recommendation system: Combining implicit rating-based collaborative filtering and sequential pattern analysis," Electron. Commer. Res. Appl., vol. 11, no. 4, pp. 309–317, Jul. 2012.

[8] R. Burke, "Hybrid Web Recommender Systems," Springer Berlin Heidelberg, pp. 377–408, 2007.

[9] G. Groh and C. Ehmig, "Recommendations in Taste related Domains: Collaborative Filtering vs. Social Filtering," In Proceedings of GROUP '07, pp. 127–136, 2007. ACM.

[10] A. Said, E. W. De Luca, and S. Albayrak, "How Social Relationships Affect User Similarities," In Proceedings of the 2010 Workshop on Social Recommender Systems, pp. 1–4, 2010.

[11] H. Lee, H. Kim, "Improving Collaborative Filtering with Rating Prediction Based on Taste Space," Journal of Korean Institute of Information Scientists and Engineers, Vol.34, No.5, pp.389- 395, 2007.

[12] P. Li, and S. Yamada, "A Movie Recommender System Based on Inductive Learning," IEEE Conf. on Cybernetics and Intelligent System, pp.318-323, 2004.

[13] W. Woerndl and J. Schlichter, "Introducing Context into Recommender Systems," Muenchen, Germany: Technische Universitaet Muenchen, pp. 138-140.

[14] G. Adomavicius and A. Tuzhilin, "Context-aware Recommender Systems," in Recommender Systems Handbook: A Complete Guide for Research Scientists and Practitioners, Springer, 2010.

[15] T. Bogers, "Movie recommendation using random walks over the contextual graph," in Proc. of the 2nd Workshop on Context-Aware Recommender Systems, 2010.

[16] Tang, T. Y., & McCalla, "A multi-dimensional paper recommender: Experiments and evaluations," IEEE Internet Computing, 13(4),34–41, 2009.

[17] Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, "Item-based collaborative filtering recommendation algorithms," In: Proceedings of the 10th international World Wide Web conference, pp. 285–295, 2001.

[18] Research Paper: A Hybrid Approach using Collaborative filtering and Content based Filtering for Recommender

System ‘ G Geetha et al 2018 J. Phys.: Conf. Ser. 1000 012101

[19] Research Paper : GHRS: Graph-based Hybrid Recommendation System with Application to Movie Recommendation Zahra Zamanzadeh Darbana, Mohammad Hadi Valipourb

[20] Research Paper: “The evaluation of a hybrid recommender system for recommendation of movies” by Matevz Kunaver, Tomar Pozri, Matevz Pogacnik and Jurij Tasic; ECAI conference on August 2006 at Trento, Italy.

American Journal of Electronics & Communication

Swapnamoy Bhattacharjee et. al., American Journal of Electronics & Communication, Vol. IV (2), 19-22

An Improved Approach on Breast Cancer Detection Using Machine Learning

Swapnamoy Bhattacharjee, Shreetanu Banerjee
Swapnil Panigrahi
Soujanya Bose and Souvick Das
Department of Computer Science & Application
Institute of Engineering & Management
Salt Lake, Sector - V
swapnomoy.2002@gmail.com

Soham Goswami and Saikat Mondal

Assistant Professor at Department of Computer Science & Application
Institute of Engineering & Management
Salt Lake, Sector - V
{soham.goswami & saikat.mondal}@iem.edu.in

***Abstract* - Cancer begins when changes called mutations take place in genes that regulate cell growth. The cells can expand and divide uncontrollably thanks to the mutations. The type of cancer that arises in breast cells is called breast cancer. Generally, breast ducts or lobules are where breast cancer first appears. The ducts that convey the milk from the glands to the nipple are where the milk is created by lobules. Moreover, cancer can develop in the breast's fatty tissue or fibrous connective tissue. Unchecked cancer cells can travel to the lymph nodes under the arms and frequently invade nearby healthy breast tissue. After the cancer has reached the lymph nodes, it has a pathway to spread to other organs, parts of the body.**

As per a 2013 WHO study, "it is projected that more than 508,000 ladies passed away all around the world in 2011 because of bosom disease". Early breast cancer development may be treated and prevented. Nonetheless, a lot of women receive a malignant tumor diagnosis after it has advanced past the point of no return.

The objective of this paper is to present several approaches to investigate the application of multiple algorithms based on Machine Learning for early breast cancer detection.

Index Terms - Breast Cancer, Dataset, Random Forest, Logistic Regression, Machine Learning.

I. Introduction

Cancer is the most prominent cause of fatalities around the world, according over one crore deaths in the past one year out of which 22.6% deaths were due to breast cancer. It is the most common type of cancer among women, accounting to 14.7% of cancer cases in India. Early detection happens to be a fruitful way to control breast cancer. There are ample cases that are handled by the early detection and decrease the mortality rate. The most common as well as efficient technique that is used in the field is Machine Learning in this report we specifically used Logistic Regression and Random Forest Classifier.

II. Literature Preview

The related research on machine learning-based breast cancer diagnosis that has been done in the past is covered in this section.

Arpita Joshi and Dr. Ashish Mehta [4] compared the classification outcomes obtained using KNN, SVM, Random Forest, and Decision Tree approaches (Recursive Partitioning and Conditional Inference Tree). Wisconsin Breast Cancer dataset from UCI repository was the one used. The best classifier, according to the simulation results, was KNN, followed by SVM, Random Forest, and Decision Tree.

Using the Wisconsin Diagnostic Breast Cancer (WDBC) Dataset, David A. Omondiagbe, Shanmugam Veeramani, and Amandeep S. Sidhu [5] studied the effectiveness of Support Vector Machine, Artificial Neural Network, and Nave Bayes by integrating these machine learning approaches with feature selection/feature extraction methods to find the best suited one. As a result of its higher computational time, SVM-LDA was preferred above all the other approaches, according to the simulation outcomes.

For better dataset processing, Kalyani Wadkar, Prashant Pathak, and Nikhil Wagh [6] conducted a comparative research on ANN and SVM which included multiple classifiers like KNN, CNN, and Inception V3. According to the experimental outcomes and performance analyses, ANN performed more efficiently than SVM, making it a better classifier.

Using machine learning techniques such as the Naive Bayes classifier, SVM classifier, bi-clustering Ada Boost algorithms (HA-BiRNN), RCNN classifier, and bidirectional recurrent neural networks, Anji Reddy Vaka, Badal Soni, and Sudheer Reddy K. [7] proposed a novel method to identify breast cancer. The proposed methodology (Deep Neural Network with Support Value) and machine learning techniques were compared, and the simulated results showed that the DNN algorithm was better in terms of performance, efficiency, and image quality, factors that are critical in today's medical systems, while the other techniques didn't work as expected.

By combining Deep Learning, Artificial Neural Network, Convolutional Neural Network, and Recurrent Neural Network approaches with Machine Learning techniques including Logistic Regression, Random Forest, K-Nearest Neighbor, Decision Tree, Support Vector Machine, and Naive Bayes Classifier, Monica Tiwari, Rashi Bharuka, Praditi Shah, and Reena Lokare [8] have developed a novel method to diagnose breast cancer. According to a comparison of machine learning and deep learning techniques, the accuracy achieved by ANN and CNN models (99.3% and 97.3%, respectively) was higher than that of the machine learning models.

On the Wisconsin Breast Cancer (original) datasets, K.Anastraj, Dr. T. Chakravarthy, and K. Sriram [9] conducted a comparative analysis between different machine learning algorithms: back propagation network, artificial neural network (ANN), convolutional neural network (CNN), and support vector machine (SVM). For feature extraction and analysis of benign and malignant tumours, ALEXNET was utilised in conjunction with deep and convolutional neural networks. According to the simulation results, support vector machine is the best strategy and has produced superior outcomes (94%).

According to S. Vasundhara, B.V. Kiranmayee, and Chalumuru Suresh's [10] proposal, mammography pictures can be automatically classified as benign, malignant, or normal utilising a variety of machine learning techniques. Support Vector Machines, Convolutional Neural Networks, and Random Forest are compared and contrasted. The simulation results showed that CNN produces intuitive classification of digital mammograms utilising filtering and morphological procedures, making it the best classifier.

The dataset from Dr. William H. Walberg of the University of Wisconsin Hospital was used by Muhammet Fatih Ak [11]. This dataset was subjected to data visualisation and machine learning methods such as logistic regression, k-nearest neighbours, support vector machine, naive bayes, decision tree, random forest, and rotation forest. These machine learning methods and visualisation were implemented using R, Minitab, and Python. All the techniques were compared in a comparative analysis. The best classification accuracy (98.1%) was obtained using the logistic regression model with all features included, and the suggested method demonstrated improved accuracy performances.

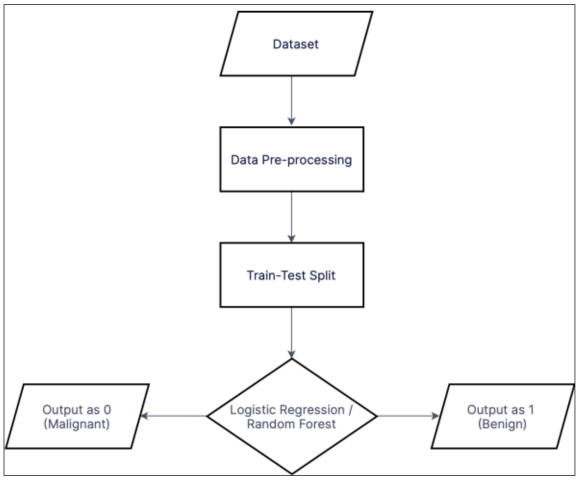
III. Objective

The objective of this model is to find the best features for the process of detecting breast cancer using machine learning, and also to find the effectiveness of the model where the values of the features could be same for both malignant case and benign Case.

IV. Libraries & Dataset

We Imported modules that are needed (Sklearn {train_test split, datasets, logistic regression}, pandas, numpy and we also imported Matplotlib and Seaborn for purpose of data representation. We used the Breast Cancer Wisconsin (Diagnostic) Data Set from sklearn, which will be reffered to as the "dataset." This database is also available in the UW CS ftp server and kaggle. The dataset contains 31 features 'mean radius,' 'mean texture,' 'mean perimeter,' 'mean area,' 'mean smoothness,' 'mean compactness,' 'mean concavity,' 'mean concave points,' 'mean symmetry,' 'mean fractional dimension,' 'radius error,' 'texture error,' 'perimeter error,' 'area error,' 'smoothness error,' 'compactness error,' 'concavity error,' 'concave points error,' 'symmetry error,' 'fractal dimension error,' 'worst radius,' 'worst texture,' 'worst perimeter,' 'worst area,' 'worst smoothness,' 'worst compactness,' 'worst concavity,' 'worst concave points,' 'worst symmetry,' 'worst fractal dimension,' & 'target' The dataset from sklearn will be in numpy array format. In this paper we propose a new method to check the accuracy of the machine learning models.

V. Methodology



Then we are converting the numpy data to data frame (df) using pandas. The "target" feature contains the data if whether the case is malignant of benign represented as 0 & 1 (0 for malignant, 1 for benign) we changed the feature name from 'target' to 'label.' As every value except for 'label' is in float64 and 'label' is int64.

1) *Checking for missing values*: We checked for any null values or missing values in the columns.
2) *Find instances of the target feature*: Then we are counting how many instances of 0's are there and how many instances of 1's are there in total.
3) *Splitting:* Then we will split the input features and the label.
4) *Input and Test feature:* All the 30 columns except label will be the input features and label test feature.
5) *Creating X & Y:* The input features are taken as X and label which is the target feature is taken as Y.
6) *Train test splitting:* Then we are creating four arrays as x_train, x_test, y_train, y_test. Then we used the train_test_split () function. By that we are splitting the x array into two parts and the y array into two parts.test_size=0.2 means 80% of the dataset will be used for training and 20% will be used for testing.

Random_state will be 2 which means we are randomizing the dataset. We can write any number instead of 2(like 42).

7) *Fitting into Logistic Regression:* Then we are fitting the x_train and y_train data into logistic regression. Then we first give the training data in the model and find the accuracy of the training data, then we give the test data to find the accuracy in the test data.
8) *Fitting into RandomForestClassifier:* We again fit the X_train and Y_train in the RandomForestClassifier prediction model. Then we checked its accuracy on the test data.
9) *Finding feature priority and feature dependence:* Next we found the priority of different features in the dataset out hundred percent. Next we need to know the co-relation of every feature with other feature. For that we used two methods, 'spearman' & 'pearson' method.
10) *Feature selection:* For feature selecting we used the feature_importances_ method from which we manually selected most needed features. This process will reduce the number features from 31 to 7.
11) *Splitting and fitting with selected features*: Then again did train test splitting with the new selected and reduced number of features. 80% of the data was used as training and the rest 20% was used as test data.
12) *Fitting the data into prediction models:* We fitted the new training and test data into LogisticRegression & RandomForestClassifier Model. We again find out the accuracy of the model with reduced number of features.
13) *Finding out overlapping values affecting benign and malignant cases:* Here we plotted a graph to find out which values of the selected features were possible for both malignant case and a benign case and finding out how many such instances were there in the dataset.
14) *Finding accuracy with the overlapped values:* We found out every overlapping values of every selected feature in the dataset and the used that in the prediction models and then checked their accuracy.

IV. Observations

The dataset didn't contain any null or missing values. There were 357 benign cases or 0's & 212 malignant cases or 1's. After the first train test splitting the training data contained 455 rows and the test data contained 114 rows.

These are the following accuracies for the prediction models:

LogisticRegression: 92.10%

RandomForestClassifier: 94.73%

After finding the feature priority we can see that 'worst area' has the highest priority of 15.07% among all the features followed by 'worst perimeter' with priority of 13.67% among all the features.

After feature selection there were 7 features selected for the next process those were 'mean concavity,' 'mean concave points,' 'area error,' 'worst radius,' 'worst perimeter,' 'worst area,' 'worst concave points' and the target feature of 'label.'

After splitting and fitting these features to the prediction model we get the following accuracies:

LogisticRegression*:* 91.22%

RandomForestClassifier; 93.85%

As we can see after feature selection the accuracy of the models are reduced by approx. 1%.

After we plotted the count of malignant and benign cases with respect to the values of the features.

We can also see some overlapping of values. Where for the same value a case can be benign or malignant. Such as for mean concavity the values between 0.020000 & 0.15000 have cases of both benign & malignant, and for 'worst area' the overlapping range is between 490 & 1250. Like this we found the overlapping values for all 7 features and put them in a dataset and used them as test data for our prediction models.

These are accuracies after using overlapping values as test data:

LogisticRegression*:* 80.76%

RandomForestClassifier: 95.38%

Here we found something interesting, the accuracy of LogisticRegression reduced significantly due to a hard to predict data whereas the accuracy of RandomForestClassifier has increased. We then gave manual input to both the models to predict if a case is malignant or benign and both the models were able to predict it successfully.

The reason for the increased accuracy could be that the RandomForestClassifier is overfitting. For that we propose to give more emphasis on developing algorithms on models like Logistic Regression so that new and hard to predict data can be predicted with more accuracy, for that use image processing along with these prediction could also help to make the system more comprehensive.

Serial No.	Comparisons between the models with different values & features		
	Model name	Accuracy	Feature type
1	Logistic Regression	92.10%	Data before feature selection.
2	Random Forest Classifier	94.73%	Data before feature selection.
3	Logistic Regression	91.22%	Data after feature selection.
4	Random Forest Classifier	93.85%	Data after feature selection
5	Logistic Regression	80.76%	Selected features with overlapping values that appear in both malignant & benign cases
6	Random Forest Classifier	95.38%	Selected features with overlapping values that appear in both malignant & benign cases

V. Future scope

With the breast cancer diagnostic tool we are looking forward in the future by reducing the price of cancer detection to some extent. It will be easy to use detection tool which everyone can use and run even in low budget devices. If the tools that uses A.I. & Machine Learning are implemented in the medical sector, it can be used in various hospitals and in homes without undergoing a lot of hassles. Most the time cancer becomes deadly and incurable because it's not diagnosed properly in the first place, but with AI tool one can expect reliability and accuracy in the future which will help in detecting cancer cells in the first place and the patient can start treatment as soon as possible.

VI. Conclusion

The most common cause of death for women is breast cancer, a condition that can be fatal to female patients. Breast cancer, which accounts for 23% of all cancer deaths in postmenopausal women, is one of the most common malignant diseases overall. Though A.I. & machine learning has come a long way in predicting these diseases

More data & new features will be required to make this process faster and less expensive and also to make the model more accurate. Breast cancer detection and screening have improved as a result of increased public attention, breast cancer awareness, and advancements in breast imaging has also made a positive impact on recognition and screening of breast cancer.

Through these methods of prediction and implementation of A.I. in medical space, we can help many to get the treatment for cancer at the right time.

On this basis, have presented this model, by which breast cancer can be recognized using machine learning algorithms.

Acknowledgment

We would like to express our special thanks of gratitude to our Guide Saikat Mondal who helped us a lot in this project, his valuable suggestions helped us to solve tough challenges and without his help this report could not have been completed in time. A special thanks to our Head of Department Prof. Abhishek Bhattacharya who gave us the golden opportunity to do this wonderful report on the topic **"**A Report on Breast Cancer Detection Model Using Machine Learning**"**, which helped us to gain a significant knowledge in the aforesaid subjects. Secondly, we would like to thank our friends who helped us a lot in finalizing this report.

Refernces

[1] Kumar Sanjeev Priyanka, "A Review Paper on Breast Cancer Detection Using Deep Learning," 2021 IOP Conf. Ser.: Mater. Sci. Eng.

[2] Sarthak Vyas,Abhinav Chauhan, Deepak Rana,Mohd Noman, "Breast Cancer Detection Using Machine Learning Techniques," Researchgate, 2022

[3] Yash Amethiya, Prince Pipariya, Shlok Patel, Manan Shah, "Comparative analysis of breast cancer detection using machine learning ad biosensors," Intelligent Machine, voL 2, pp. 69-81, May 2022 .

[4] Arpita Joshi and Dr. Ashish Mehta "Comparative Analysis of Various Machine Learning Techniques for Diagnosis of Breast Cancer," 2017.

[5] David A. Omondiagbe, Shanmugam Veeramani and Amandeep S. Sidhu "Machine Learning Classification Techniques for Breast CancerDiagnosis," 2019.

[6] Kalyani Wadkar, Prashant Pathak and Nikhil Wagh "Breast Cancer Detection Using ANN Network and Performance Analysis with SVM," 2019.

[7] Anji Reddy Vaka, Badal Soni and Sudheer Reddy "Breast Cancer Detection by Leveraging Machine Learning," 2020.

[8] Monika Tiwari, Rashi Bharuka, Praditi Shah and Reena Lokare "Breast Cancer Prediction using Deep learning and Machine Learning Techniques".

[9] K.Anastraj, Dr.T.Chakravarthy and K.Sriram," Breast Cancer detection either Benign Or Malignant Tumor using Deep Convolutional Neural Network With Machine Learning Techniques," 2019.

[10] S.Vasundhara, B.V. Kiranmayee and Chalumuru Suresh "Machine Learning Approach for Breast Cancer Prediction," 2019.

[11] Muhammet Fatih Ak "A Comparative Analysis of Breast Cancer Detection and Diagnosis Using Data Visualization and Machine Learning Applications," 2020.

[12] Sivapriya J, Aravind Kumar V, Siddarth Sai S, Sriram S "Breast Cancer Prediction using Machine Learning," 2019.

[13] Hiba Asria, Hajar Mousannifb, Hassan Al Moatassime, Thomas Noeld "Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis," 2016.

[14] Dana Bazazeh and Raed Shubair "Comparative Study of Machine Learning Algorithms for Breast Cancer Detection and Diagnosis," 2016.

[15] Vishabh Goel, "Building a Simple Machine Learning Model on Breast Cancer Data," Towards Data Science, Sep 29, 2018.

American Journal of
Electronics & Communication

Pinaki Pritam Singha et. al., American Journal of Electronics & Communication, Vol. IV (2), 23-26

IoT-based Fire Alarm: Design and Approach

Pinaki Pritam Singha[1], Sandip Mandal[2]
[1,2]University of Engineering and Management, Kolkata, West Bengal
Corresponding Author: [1]*sandy06.gcect@gmail.com*

***Abstract*— The Internet of Things (IoT) describes the network of physical objects— "things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. These devices range from ordinary household objects to sophisticated industrial tools. With more than 7 billion connected IoT devices today, experts are expecting this number to grow to 10 billion by 2020 and 22 billion by 2025.**

Our main objective is to use this technology to build a working fire alarm that would not only work like a regular alarm but will also send a notification to cloud connected devices.

***Index Terms*— Fire Alarm, Notification, Cloud Alert, Security.**

I. Introduction

The **Internet of things** (**IoT**) describes physical objects (or groups of such objects) with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks. Internet of things has been considered a misnomer because devices do not need to be connected to the public internet, they only need to be connected to a network and be individually addressable.

The field has evolved due to the convergence of multiple technologies, including ubiquitous computing, commodity sensors, increasingly powerful embedded systems, and machine learning. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), independently and collectively enable the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", including devices and appliances (such as lighting fixtures, thermostats, home security systems, cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. IoT is also used in healthcare systems.

The main concept of a network of smart devices was discussed as early as 1982, with a modified Coca-Cola vending machine at Carnegie Mellon University becoming the first ARPANET-connected appliance, able to report its inventory and whether newly loaded drinks were cold or not. Mark Weiser's 1991 paper on ubiquitous computing. In 1994, Reza Raji described the concept in *IEEE Spectrum* as "[moving] small packets of data to a large set of nodes, so as to integrate and automate everything from home appliances to entire factories". Between 1993 and 1997, several companies proposed solutions like Microsoft's at Work or Novell's NEST. The field gained momentum when Bill Joy envisioned device-to-device communication as a part of his "Six Webs" framework, presented at the World Economic Forum at Davos in 1999. The term "Internet of things" was coined independently by Kevin Ashton of Procter & Gamble, later of MIT's Auto-ID Centre.

A growing portion of IoT devices are created for consumer use, including connected vehicles, home automation, wearable technology, connected health, and appliances with remote monitoring capabilities. IoT devices are a part of the larger concept of home automation, which can include lighting, heating and air conditioning, media and security systems and camera systems. Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off or by making the residents in the home aware of usage.

II. LITERATURE SURVEY

IoT system architecture, in its simplistic view, consists of three tiers: Tier 1: Devices, Tier 2: the Edge Gateway, and Tier 3: the Cloud. Devices include networked things, such as the sensors and actuators found in IoT equipment, particularly those that use protocols such as Modbus, Bluetooth, Zigbee, or proprietary protocols, to connect to an Edge Gateway. The Edge Gateway layer consists of sensor data aggregation systems called Edge Gateways that provide functionality, such as pre-processing of the data, securing connectivity to cloud, using systems such as WebSockets, the event hub, and, even in some cases, edge analytics or fog computing. Edge Gateway layer is also required to give a common view of the devices to the upper layers to facilitate in easier management. The final tier includes the cloud application built for IoT using the microservices architecture, which are usually polyglot and inherently secure in nature using HTTPS/OAuth. It includes various database systems that store sensor data, such as time series databases or asset stores using backend data storage systems (e.g., Cassandra, PostgreSQL). The cloud tier in most cloud-based IoT system features event queuing and messaging system that handles communication that transpires in all tiers. Some experts classified the three-tiers in the IoT system as edge, platform, and enterprise and these are connected by proximity network, access network, and service network, respectively. WSNs comprise of the Tier 1: Devices.

A wireless sensor network (WSN) is a self-configuring wireless network with minimal infrastructure that monitors physical or environmental conditions including temperature, sound, vibration, strain, motion, or contaminants and transmits data via the network's first place. Or a receiver that can observe and analyse data. The receiver or base station serves as the interface between the user and the network. By entering a query and collecting the results from the recipient, you can get the information you need from the Internet. A wireless sensor network usually has thousands of sensor nodes. Sensor nodes can communicate with each other via radio signals. Wireless sensor nodes are equipped with sensitive equipment and computing equipment, radio transmitters and power supply components. Each node in a wireless sensor network (WSN) is resource constrained in some way: processing speed, storage space, and communication bandwidth are all restricted. After installation, the sensor nodes are in charge of self-organizing the necessary network infrastructure and frequently communicate with them through multi-hop communication. Then, the built-in sensors begin to collect information of interest. Wireless sensor devices also respond to requests from "checkpoints" to follow specific instructions or provide samples for testing. The sensor node can operate in either a continuous or event-driven mode. To calculate your location, you can use the Global Positioning System (GPS) and local positioning algorithms. Actuators may be added to wireless sensor systems to make them "work" in specific situations. Wireless sensor and actuator networks are a more generalized term for these networks. New technologies can be supported by wireless sensor networks (WSNs). Protocol architecture necessitates unconventional paradigms due to a variety of constraints. Due to the requirements for low device complexity and low power consumption (i.e., long network life), a reasonable balance needs to be struck between communication and signal/data processing capabilities. This has stimulated tremendous efforts in the field of research, standardization and wireless sensor networks.

The first fire alarm was discovered by accident in 1980 by Francis Robbin Upton, an associate of Thomas Edison to create a device to automatically detect poison gas electrically.

A modern-day fire alarm warns people when there is smoke, fire, carbon dioxide or other fire related, general emergencies. The device automatically detects the presence of smoke or fire and sets an alarm off electrically through the circuit.

However, with the use of present-day smartphones, Wi-Fi and globally connected devices, it has become an importance for a device to take use of the notification system and be remote controlled without always the requirement of an external physical contact for application of appliances especially those which are necessary during emergencies.

III. PROBLEM STATEMENT

IOT Based Fire Alerting System uses two Sensors, namely, Temperature and Smoke sensors. Arduino has an inbuilt ADC converter, which converts the analog signals received at the sensor end to digital. The Arduino is programmed to turn on the buzzer when the temperature & the smoke reach a threshold value.

At the same time, Arduino sends the data to the Wi-Fi module ESP8266. ESP8266 is a chip that is used for connecting micro-controllers to the Wi-Fi network. ESP8266 will then the following data to the IOT website, where, authorized people can take appropriate measures in order to curb the fire.

1. Temperature (in Degree Celsius)
2. Smoke Value (in Percentage)
3. Device ID
4. Date and Time Stamp

The device ID is the unique ID given to a device, which would help the person get information related to the location, where the fire is detected.

The Prerequisite for this IoT-based fire alarming system is that the Wi-Fi module should be connected to a Wi-Fi zone or a hotspot. This project is also implemented without the IOT module. In place of the IOT module, we have used the GSM module, by which an SMS is triggered when the buzzer is turned ON.

IV. PROPOSED SOLUTION

1. Study the working principles of smoke and fire alarm systems.

2. Design a cheap fire alarm system based on microcontroller.

3. Design an automatic fire alarm system to protect users and the environment.

4. Create a simple fire alarm system. Use a fire alarm system.

5. Make people's lives easier.

6. Design a prototype fire alarm system with smoke detector as input and buzzer and text message as output. Arduino Uno card, embedded system: NodeMCU, cable connection, buzzer etc.

V. EXPRERIMENTAL SETUP AND RESULT ANALYSIS

A. Components:

1)

2) NodeMcu Board

It is an open-source firmware that uses the LUA scripting language.

3) Flame Sensor

A Flame Sensor or a Flame detector is a device designed to detect and respond to the presence of a flame or fire, allowing flame detection.

4) Jumper Wires

A jumper is an electrical wire, a group of them bundles with pins or connector attached at the end. Wires are fitted by using the pins by putting them on a breadboard.

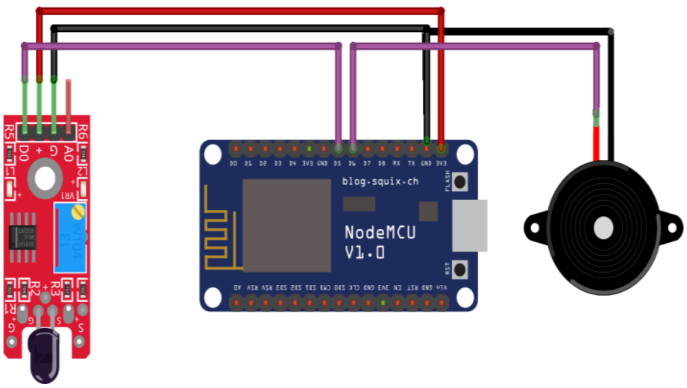
5) Breadboard

A Breadboard is a temporary construction base used to build electronic circuits and basic technological device.

6) Buzzer

A Buzzer for making a sharp noise upon detection of fire by the circuit.

7) *Circuit Diagram:*



Code (Arduino):

```
#define BLYNK_TEMPLATE_ID "TMPL38yR_clE"
#define BLYNK_DEVICE_NAME "Fire Alarm 3"
#define   BLYNK_AUTH_TOKEN   "6qVODjUwNl7cnyd-
aiz1GAz1B0t8XbMh"

#define BLYNK_PRINT Serial

  #include <ESP8266WiFi.h>
  #include <BlynkSimpleEsp8266.h>
  #define fireSensor D5
  #define led D4
  #define Buzzer D6

char auth[] = BLYNK_AUTH_TOKEN;
/*
// Base config
char ssid[] = "MON";  // wifi name
char pass[] = "abcd1234";  // wifi password
*/
// Mirror config
char ssid[] = "Redmi Note 8 Pro";  // wifi name
char pass[] = "12345678";  // wifi password

 BlynkTimer timer;

int fireState = 0;
int lastfireState = 0;
unsigned long old =0;
unsigned long current =0;
int interval=0;

void Reading()
{

  fireState = digitalRead(fireSensor);
  if (fireState == 0 && lastfireState == 0) {
  Serial.println("Blynk notification: Fire!");
  Blynk.logEvent("fire","Fire Alert");
  lastfireState = 1;
  tone(Buzzer,1000);
  delay(1000);
  }
  if (fireState == 0 && lastfireState == 1) {

    Serial.println("Fire Alert, Continous");
    tone(Buzzer,2000);
    delay(5000);
    }
  if (fireState == 1) {
   Serial.println("No Fire");
    lastfireState = 0;
  noTone(Buzzer);

  }
}


void setup()
{
  Serial.begin(115200);
  Serial.println();
  pinMode(led,OUTPUT);
  Serial.println("Please    wait....    Sensor
activation");
  delay(1000);
  Serial.println("Please wait for Blynk Server
connection");
  pinMode(fireSensor, INPUT);
  pinMode(Buzzer,OUTPUT);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    digitalWrite(led,LOW);
    delay(250);
    Serial.print(".");
    digitalWrite(led,HIGH);
    delay(250);
  }
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(1000L, Reading);

}

void loop()
{
  Blynk.run();
  timer.run();
}
```

VI. CONCLUSION & FUTURE SCOPE

The model continuously monitors fire alarms and sends alarms to users. The reception and system we propose can achieve its main goal, mainly to build an IoT-based fire alarm system. Call them when you find the fire. The answer is sent to the user via notification. Using this product can help these people quickly learn about the incident and the nearest fire department. You will receive a valid notification. It is cheap and easy to install.

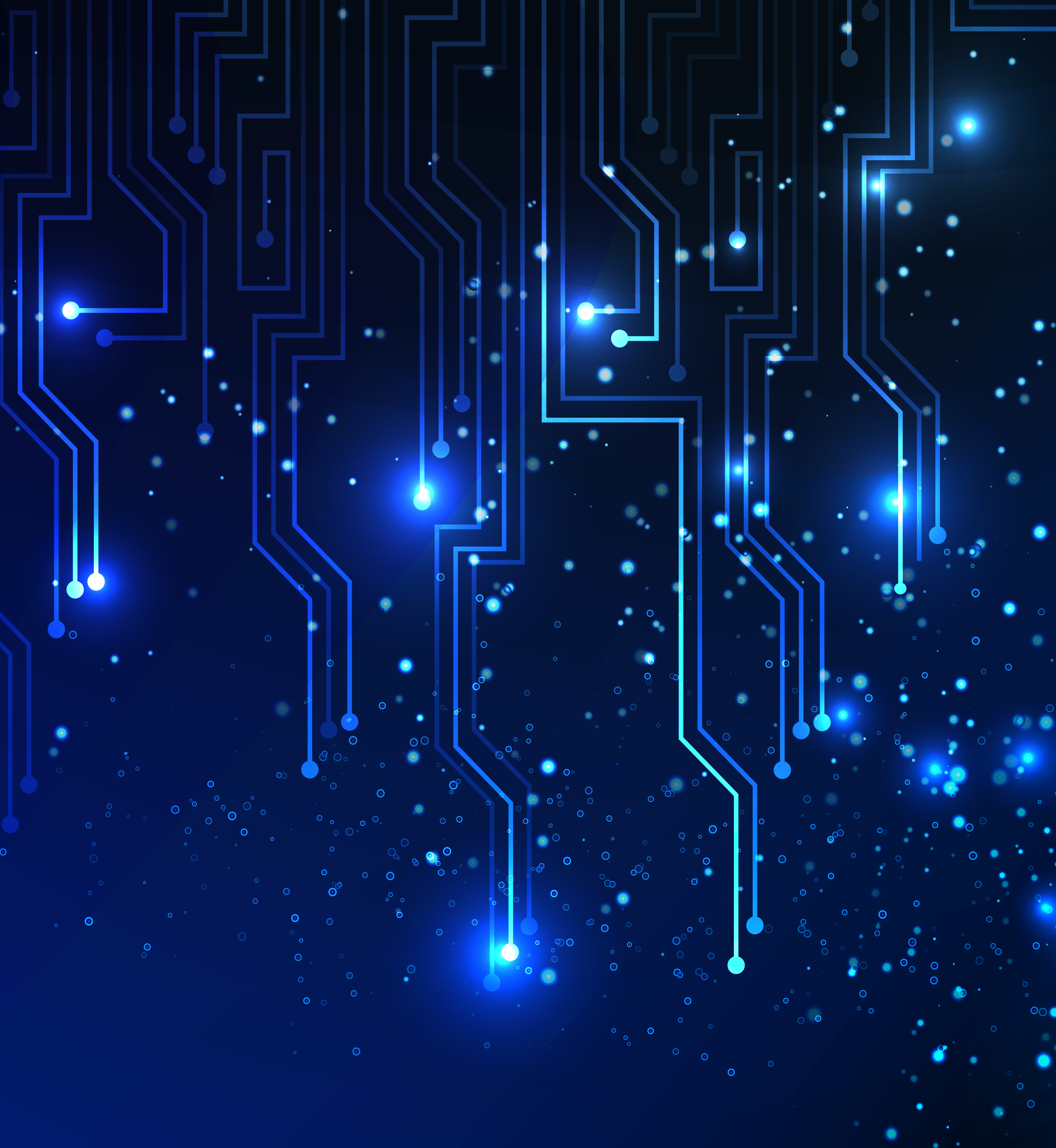
REFERENCES

Book Referred:

1. International Journal of Computer Applications
2. Oracle Industries IoT Report

Websites Referred:

1. Wikipedia [https://en.wikipedia.org/wiki/Internet_of_things : 7-11-22 19:44]
2. StackOverflow [https://stackoverflow.com/search?q=fire+aLARM&s=9023917d-b2b2-4f70-8174-5de0f447c5ef : 7-11-22 20:11]
3. Blynk [https://blynk.io/ 1-11-22 8:31]



American Journal of **Electronics & Communication**

Society for Makers, Artists, Researchers and Technologists,
USA 6408 Elizabeth Avenue SE, Auburn, Washington 98092.
ISSN 2690-2087