NEAR EAST UNIVERSITY

JOURNAL FOR ARTIFICIAL INTELLIGENCE AND INTERNET OF THINGS

Volume: 1 Issue:1

ISSN: 3062-1968

www.neu.edu.tr

JOURNAL OF ARTIFICIAL INTELLIGENCE AND INTERNET OF THINGS
International, Refereed Journal

February , 2022
Cilt-Volume 01/ Sayı-Issue 01

Foundation Year of the Journal
2022

Address and Contact
Yakın Doğu Üniversitesi İnovasyon ve Bilişim Teknolojileri Merkezi
International Research Center for AI and IoT
Yakın Doğu Bulvarı, PK: 99138Lefkoşa / KKTC Mersin 10 –TÜRKİYE
Tel:+90 (392) 223 64 64/+90 (392) 680 20 00Faks:+90 (392) 223 64 61
http://dergi.neu.edu.tr/ https://iot.neu.edu.tr/

Publication Contact

Prof. Dr. Fadi AL-TURJMAN

CONTENTS

A Comparative Analysis of Stress among Undergraduate Students Using Logistic Regression and Random Forest Techniques

Uzma Uzma, Subhabrata Kanjilal, Nishi Yadav*

[1]School of Studies in Engineering and Technology, Guru Ghasidas Vishwavidyalaya Bilaspur, India

uzma.shine2017@gmail.com, rohan525kanjilal@gmail.com, nyadav.cse@gmail.com

Abstract: Mental illness has become a major problem for youngsters nowadays. Our project deals with calculation of stress as we know that overall collegiate performance and social obligation have created a pressurized cerebral as well as emotional state for students. With limited college seats, and high number of post metric students applying to get into the top universities and colleges, it could be difficult to get into the college one wished for. Same is the case for a student in his last year of graduation. There is lot of pressure that one undergoes like pressure of getting placed, pressure of getting into a top college for PG and many more. Lastly the stress caused due to the pandemic can be least ignored. Students weren't able to attend online classes properly due to lack of resources which resulted students to undergo a lot of stress about their academics. We collected the data through a google survey form which was send to all the students known to us. We collected 101 responses and then converted them into numerical value and lastly implemented the logistic regression and random forest algorithm, where we got our f1 score = 0.9411 and Accuracy score (Random Forest) = .0.8571.

Keywords— Stress, Logistic regression, Random Forest, Mental illness, Predictive Analysis, Stress in Students, Machine Learning, Data Analysis, Causes & Effects of Stress.

1. INTRODUCTION

Stress can be described as a condition where someone is mentally ill because of the adversities/changes that he/she is undergoing or underwent [1]. There are many types of stress such as physical stress which includes trauma, endocrine and/or biochemical disparity, nutritive stress, Exsiccosis, chemical abuse, toothache, and human locomotor system, misalignment or imbalance. The second type can be psychological stress which includes emotional stress, cognitive stress, and perceptual stress. Third category can be Psychosocial stress: Relationship/marriage difficulties, lacking of social help, lacking of assets for better survival, loss of job/money/saving, loss of your dearest, bankruptcy, house expropriate, and separation. Lastly the psycho-spiritual stress which can be a exigency of manners, significance, and motive; melancholy fighting (apart from being fruitful, rewarding, meaningful be gloomy); and also moving out of track within one's core faith or beliefs. As a optimistic result, stress can urge us to perform an operation that we wanted to do. As a negative impact it can lead to certain health related issues like headache, increase in heart rate, increase in blood pressure etc. Due to these health issue sometimes, the person consecutively leads to depression, anger or sometimes distrust. It differs from person to person so as how he/she deals with a particular situation. When a person gets too much happy or sad the stress within him/her readjusts itself in accordance with the current circumstances that the person is undergoing.

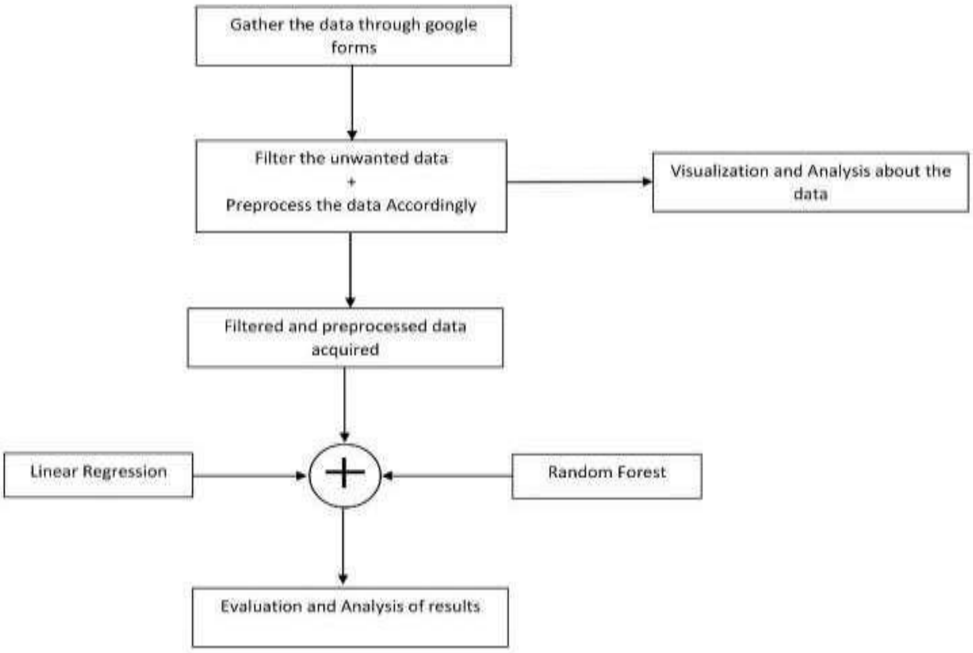

Gather the data through google forms
Filter the unwanted data + Preprocess the data Accordingly
Visualization and Analysis about the data
Filtered and preprocessed data acquired
Linear Regression
Random Forest
Evaluation and Analysis of results

Fig. 1. Graphical Abstract

Table.1. Causes & Effects of Stress reported by students

Behavioral effects: - i. Alteration in activity levels ii. Decrease in ability and efficacy iii. Difficultly in communicating iv. Increased sense of humour/gallows humour v. Irritability, outbursts of anger, frequent arguments vi. Ability to nap, chill or let down vii. Modification in eating regime viii. Insomnia ix. Change in work execution x. Periods of bawl xi. Increment in intoxication, sugar or caffeine xii. Attentive about safety or the surrounding habitat xiii. Avoidance of things or places that prompt memories xiv. Accident prone	**Causes: -** i. Study overload ii. Lack of financial Support iii.Family issues iv.Issues with friend or significant partner v. Health Related Issue vi.Involvement in clubs/organisation
Psychological or emotional effects: - i. Feeling fearless, joyful or invincible ii. Contradiction in thoughts iii. Anxiety or alarmed iv. Worry about security of yourself and others v. Get irritated or angry easily vi. Anxious or nervous vii. Misery, sullen, regret or depression viii. Realistic or anguish dreams ix. Apologetic or "survivor guilt" x. Feeling speechless, incapable or desperate	**Cognitive effects: -** i. Memory problems/forgetfulness ii. Disorientation iii. Confusion iv. Slowness in thinking, analysing, or comprehending v. Difficulty in calculating, setting preference giving opinion vi. Difficulty focusing vii. Short notice span viii. Loss of aim

xi. Feeling deserted, off-track, desolate or abandoned xii. Boredom xiii. Compulsive behaviour Feeling misinterpreted or unacknowledged	ix. Foggy about the disaster or an incident happened
Bodily effects – i. Tachycardia ii. Increased blood pressure iii. Upset stomach, nausea, diarrhoea iv. Increased or decreased appetite which may be accompanied by weight loss or gain v. Sweating or chill vi. Tremors or muscle twitching vii. Muffled hearing viii. Tunnel vision ix. Feeling uncoordinated x. Headaches xi. Pain in muscles xii. Photosensitive sight xiii. Lumbago xiv. Globus sensation xv. Easily startled xvi. Tiredness that does not go with sleep xvii. Menstrual cycle changes Change In sexual desire or response	

2. MATERIALS AND METHODS

A. Study Population

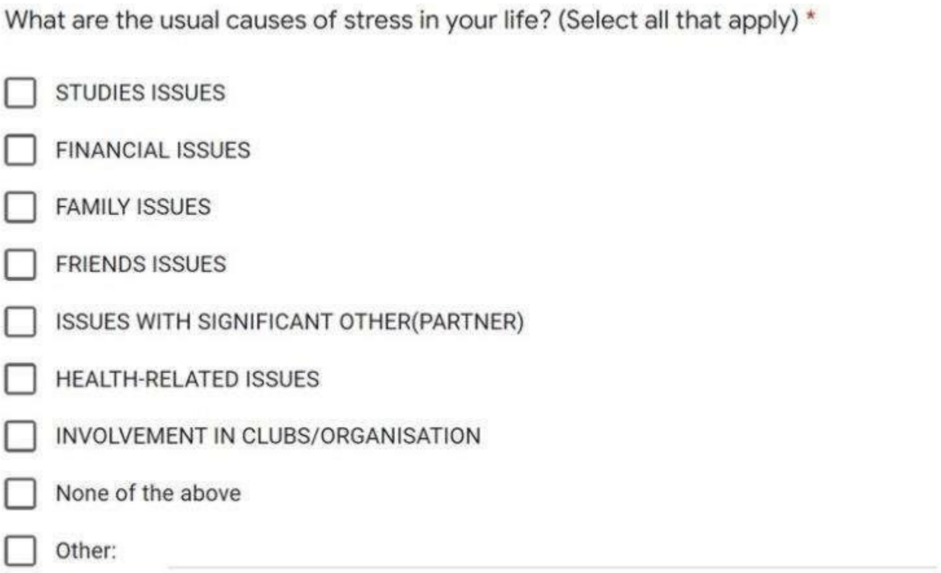
We created a google form having 12 questions asking about the person's experience about how he dealt with the stress. We also tried to put certain questions where we enquired about whether he feels or not that he is in stress by scaling himself from the number 1 to 10 , where 1 meant not at all stressed and 10 meant very stressed. Similarly, we enquired about how much he can handle the stress by scaling himself from the number 1 to 10 , where 1 meant cannot handle at all and 10 meant can to handled easily . The questionnaire also dealt with what were the causes of stress and how did it affected the person

B. Filtering the data

We received about 109 responses but after filtering we were left with only 101. We went through each and every minute details of the responses received and deleted the responses which were duplicate or were not felt genuine. We then created another xml sheet which dealt with only the information required excluding the information like gender, study, year of birth etc.

C. Conversion of the data

The responses were then converted into numerical like if a person has ticked one option for the usual cause of stress as study issues, then we gave 1 to the vertical corresponding study issue and gave 0 to all other usual causes of stress. Similarly, it was done for all the left 81 responses. Finally, we added all the values of a particular row and stored them in their corresponding row. Then we took the average of all the values and concluded that a person whose values were added below 34 were not in stress whereas above 34 could be in stress. For the people who got a score below 34 we checked their answer for the question - How do you usually experience stress, if the answer to this particular was found relevant then too we concluded that the person is in stress even though he has a score below 34. As we can see in the screenshot below that the first response was that the usual cause of stress for that student was issue with significant other partner so we gave 1 to this particular vertical and the left other columns were marked 0.



What are the usual causes of stress in your life? (Select all that apply) *

□ STUDIES ISSUES

□ FINANCIAL ISSUES

□ FAMILY ISSUES

□ FRIENDS ISSUES

□ ISSUES WITH SIGNIFICANT OTHER(PARTNER)

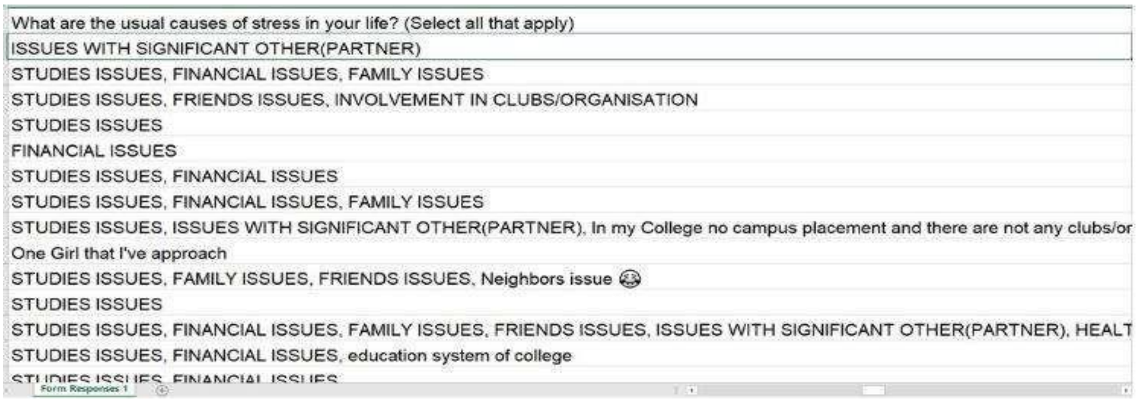
□ HEALTH-RELATED ISSUES

□ INVOLVEMENT IN CLUBS/ORGANISATION

□ None of the above

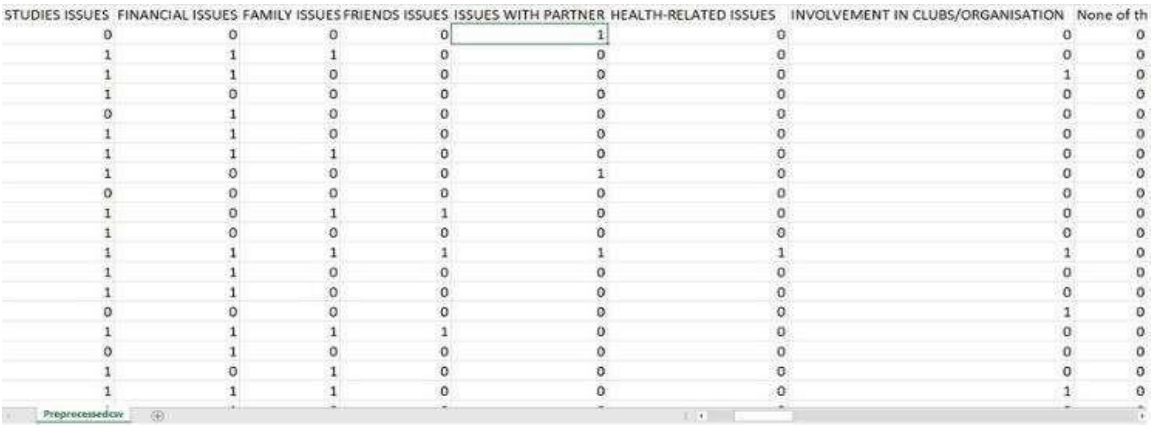
□ Other: ____

Fig. 2. Screenshot of the Google Form



What are the usual causes of stress in your life? (Select all that apply)
ISSUES WITH SIGNIFICANT OTHER(PARTNER)
STUDIES ISSUES, FINANCIAL ISSUES, FAMILY ISSUES
STUDIES ISSUES, FRIENDS ISSUES, INVOLVEMENT IN CLUBS/ORGANISATION
STUDIES ISSUES
FINANCIAL ISSUES
STUDIES ISSUES, FINANCIAL ISSUES
STUDIES ISSUES, FINANCIAL ISSUES, FAMILY ISSUES
STUDIES ISSUES, ISSUES WITH SIGNIFICANT OTHER(PARTNER), In my College no campus placement and there are not any clubs/or
One Girl that I've approach
STUDIES ISSUES, FAMILY ISSUES, FRIENDS ISSUES, Neighbors issue 😂
STUDIES ISSUES
STUDIES ISSUES, FINANCIAL ISSUES, FAMILY ISSUES, FRIENDS ISSUES, ISSUES WITH SIGNIFICANT OTHER(PARTNER), HEALT
STUDIES ISSUES, FINANCIAL ISSUES, education system of college
STUDIES ISSUES, FINANCIAL ISSUES

Fig. 3. Screenshot of the Responses



STUDIES ISSUES	FINANCIAL ISSUES	FAMILY ISSUES	FRIENDS ISSUES	ISSUES WITH PARTNER	HEALTH-RELATED ISSUES	INVOLVEMENT IN CLUBS/ORGANISATION	None of th
0	0	0	0	1	0	0	0
1	1	1	0	0	0	0	0
1	1	0	0	0	0	1	0
1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0
1	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0
1	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0
0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0
1	1	1	0	0	0	1	0

Fig. 4. Screenshot of the converted data

D. Related work

In [1], the author founded analytically notable difference between bottom line and stress periods for seven from the complete parameters used. Given the determined noteworthy similarities between some of the variables. They concluded that three EDA variables (average SCL, number of 14 peaks and duration) and one HRV parameter (average HR/ average RR/HF) can be used in a hand-on stress analysis procedure [1].

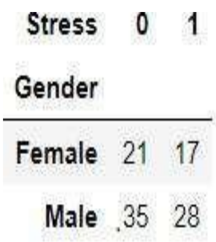
In [2], the authors found that out of these four algorithms Naïve Bayes, K-Nearest Neighbors, Support Vector Machine, Random Forest. SVM has performed good as its quantity of data is and its geometric way of categorization is also low. Finding and Analyzing methods like PSS with increased exact conclusions and small cost can help improve the cerebral health of each person and make everyone sound fine (mentally) [2]. In [3], the authors developed a tool using Naive Bayes and Sentimental Analysis which was capable of classifying student tweets into a different set of categories based on the student expressed emotions [3]. In [4], the authors compared two algorithms that were Naive Bayes and K-Nearest Neighbors, after all the [13] processing done they concluded that the switch in the amount of information has affected accuracy, f1-score, recall & precision, both through the percentage split tests and k-cross validation [4]. In [5], the authors worked on the Logistic Regression, SVM (Linear Kernel), KNN, Decision Tree, Random Forest and Proposed Ensemble Model and they concluded that more than 74% of the students experienced stress [5]. In [6], the authors described stress as the actuality that is worldwide existed, among all the students of almost each and every stream, irrespective of gender, age, class and other attributes. They performed a parallelepiped questionnaire-based look-over before coming to this conclusion [6]. In [7], the authors used two methods namely the Coping Inventory for Stressful Situations (CISS) & Alcohol Use Dependency Identification Test (AUDIT), and concluded that there is no co-relation among drinking intentions, alcohol dependency risk and stress handling styles in the observed group of people [7]. In [8], the authors observed that the relationship between ACEs, and stress report by the individual was managed by the PTSD-S. This shows that students who report PTSD-S following childhood misfortune undergo higher levels of stress. Traditional and non-traditional students have a difference in their ICLRE scale responses [8].

3. DATA PROCESSING

A. Algorithms

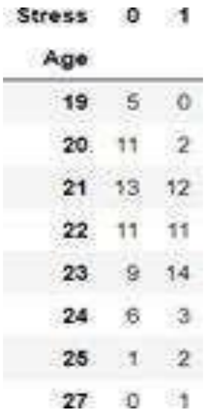
Logistic regression has been used to estimate the probability that whether a student is suffering from high stress (1) or is in low stress (0), using the causes and effects that are thought to be related to or influence such cause [14]. Here in this paper, we have implemented Binary Logistic Regression which has a dependent variable “Stress” represented by a target variable, which has two values labelled ”0” which represents low stress and ”1” which represents high stress [9]. Logistic regression has been used to estimate the probability that whether a student is suffering from high stress (1) or is in low stress (0), using the causes and effects that are thought to be related to or influence such cause [14]. The second Algorithm used is Random Forest, it considers so many decision tresses thus forming a forest. It uses selecting feature randomly for building every individual decision tree and then try to create an uncorrelated forest of trees whose prediction by a certain group of features is more accurate than that of any individual decision tree [13].

B. Data Analysis



Stress	0	1
Gender		
Female	21	17
Male	35	28

Fig. 5. In the above crosstab, shows the no. of males and females under high stress (1) / low stress (0).



Stress	0	1
Age		
19	5	0
20	11	2
21	13	12
22	11	11
23	9	14
24	6	3
25	1	2
27	0	1

Fig.6. Age to Stress Comparison

In the above crosstab, we can conclude that people who are in age range of 18 -21, most of them are under low stress (0), the people who are in age range of 20-22, some of them are under low stress (0) and some of them are under high stress (1). But in the age range of 22-26, most of the people are in high stress (1).

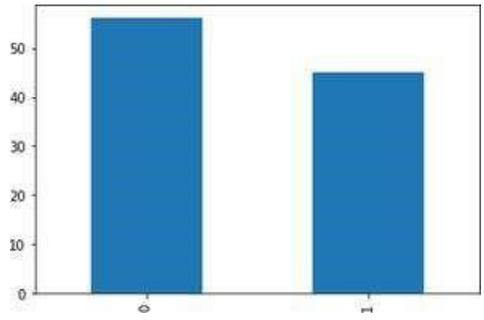


Fig. 7. Distribution of Stress

C. Performance Parameters

Accuracy: Overall, how often has the classifier correctly classified the data?

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FN+FP} \quad [15] \quad \text{-------------} \quad (1)$$

where, TP = True Positive, FP = False Positive, TN = True Negative, FN = False Negative [15].

True Positive Rate (TPR): It can be defined as when the actually *data* denotes that the person is actually in low stress but the system predicts low stress?

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \text{ [15] ---------- (2)}$$

Also known as" **Recall**" or" **Sensitivity**". [15]

False Positive Rate (FPR): It can be defined as when the actually data denotes that the person is actually in high stress but the system predicts low stress [15].

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}} \text{ [15] ----- ----- (3)}$$

True Negative Rate(TNR): It can be defined as when the actually data denotes that the person is actually in high stress but the system predicts high stress.

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{FP+TN}} \text{ [15] ---------- (4)}$$

It is equivalent to (1-FPR), also known as" **Specificity**" [15].

Precision: How much the data is actually low stressed, when the model predicts low stress correctly? Precision = $\frac{\mathrm{TP}}{\mathrm{TP+FP}}$ [15] ---------- (5)

Precision will be calculated out of all the positive classes (i.e. low stress) the model predicted correctly, versus the number of classes that are actually positive (i.e. the data is actually of low stress) and Accuracy will be the number of class that were predicted correctly out of all the classes [16].

f1 Score: It is highly unfavorable to validate a model which a high recall and a low precision value or vice-versa. In order to make the model comparable, we use f1-Score. It is basically the harmonic mean between precision and recall [17] i.e.

$$\text{f1 score} = \frac{2*\mathrm{Precision}*\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}} \text{ [15] ----------- (6)}$$

It uses Harmonic mean instead of Arithmetic mean, thus pushing the values to their extreme ends. This helps the model to be more dependable/comparable [16].

4. RESULT

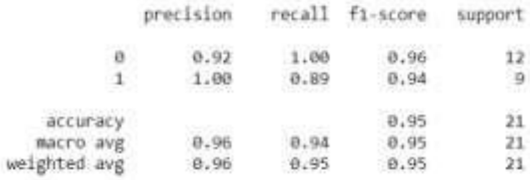
In this paper, we have used logistic regression and random forest algorithm and calculated f1 score and accuracy respectively for each algorithm. We found that majorly the students ranging from 21-23 aged group were in high stress (1) and also the ratio between male is to female in stress was found to be equal.

The performance parameters we considered above (in section III.C) were calculated as follows:

Table 2. Results Calculated for the performance parameters

Parameter s	Logistic Regression	Random Forest
Accuracy	95.23%	90.47%
TPR	92.31%	91.66%
TNR	88.88%	88.88%
FPR	0%	11.11%
Precision	100%	91.66%
f1-score	0.9411	0.8888

We can conclude that our data analysis and implementation is performing well, giving an accuracy of 95.23% and test f1 score as 0.9411 using logistic regression



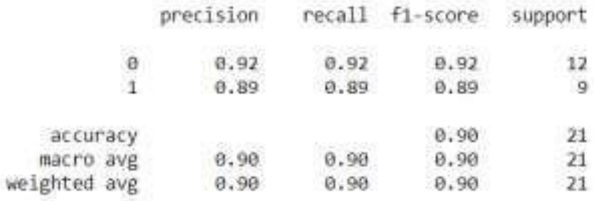
```
              precision    recall  f1-score   support

           0       0.92      1.00      0.96        12
           1       1.00      0.89      0.94         9

    accuracy                           0.95        21
   macro avg       0.96      0.94      0.95        21
weighted avg       0.96      0.95      0.95        21
```

Fig 8. Classification report for Logistic Regression

and accuracy = 90.47% and f1 score = 0.8888 using random forest.



```
              precision    recall  f1-score   support

           0       0.92      0.92      0.92        12
           1       0.89      0.89      0.89         9

    accuracy                           0.90        21
   macro avg       0.90      0.90      0.90        21
weighted avg       0.90      0.90      0.90        21
```

Fig 9. Classification report for Random Forest

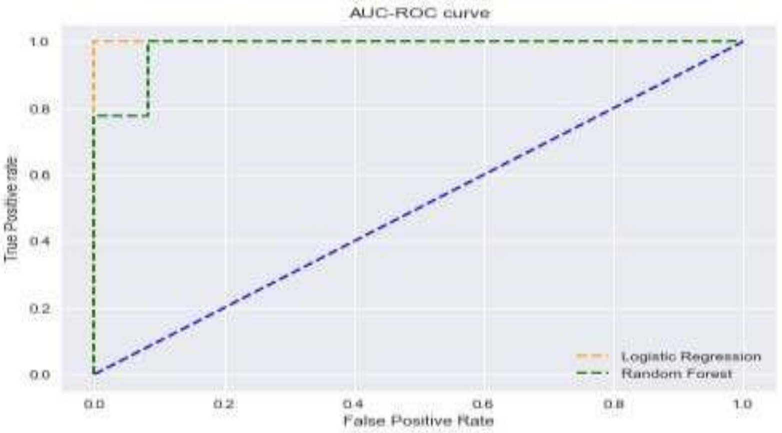


Fig. 10. Plot of AUC-ROC Curve

5. CONCLUSION

In this paper till now we have used two algorithms that are random forest and logistic regression [12] and after calculating we found that the results were good enough with accuracy of 95.23% and 90.47% respectively. Due to less availability of data, our model predicted almost everything correctly as 114 attributes were a key factor in model's performance. In future, our main target will be to acquire more and more data so the precision of model becomes more and more sharp and also, we'll try to implement the data with various other models like Naive Bayes, SVM, and may-be we can try many other different validation techniques like k-fold, cross or may be even hybrid, which might improve results.

REFERENCES

[1]. C. Goumopoulos and E. Menti, "Stress Detection in Seniors Using Biosensors and Psychometric Tests," *Procedia Computer Science,* vol. 152, pp.18-27, 2019.

[2]. R. Ahuja and A. Banga, "Mental stress detection in university students using machine learning algorithms," *Procedia Computer Science,* vol. 152, pp. 349-353, 2019.

[3]. T. Kovilpatti and V. Kalaivani, "Analyzing social media data for understanding students learning experiences and predicting their psychological pressure," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 7, pp. 513-521, 2018.

[4]. Y. C. Tapidingan and D. Paseru, "Comparative Analysis of Classification Methods of KNN and Naïve Bayes to Determine Stress Level of Junior High School Students," *Indonesian Journal of Information Systems*, vol. 2, no. 2, pp. 80-89, 2020.

[5]. G. Verma and H. Verma, "Model for predicting academic stress among students of technical education in India," *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 4, 2020.

[6]. P. S. Behere, R. Yadav, and P. B. Behere, "A comparative study of stress among students of medicine, engineering, and nursing," *Indian journal of psychological medicine*, vol. 33, no. 2, pp. 145-148, 2011.

[7]. M. Goran-Stanišić, M. Michalak, and A. Posadzy-Małaczyńska, "Drinking alcohol as a way of coping with stress in students of medical faculties," *Psychiatr. Pol*, vol. 54, no. 2, pp. 265-277, 2020.

[8]. K. A. Kalmakis et al., "Adverse childhood experiences, post-traumatic stress disorder symptoms, and self-reported stress among traditional and nontraditional college students," *Journal of American college health*, vol. 68, no. 4, pp. 411-418, 2020.

[9]. J. Tolles, and W. J. Meurer, "Logistic regression: relating patient characteristics to outcomes," *Jama*, vol. 316, no. 5, pp. 533-534, 2016.

[10]. D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *arXiv preprint arXiv:2010.16061*, 2020.

[11]. N. Kardani et al., "Improved prediction of slope stability using a hybrid stacking ensemble method based on finite element analysis and field data," *Journal of Rock Mechanics and Geotechnical Engineering*, vol.13, no.1, pp. 188-201, 2021.

[12]. S. Kour, R. Kumar and M. Gupta, "Analysis of student performance using Machine learning Algorithms," *Proceedings of the Third International Conference on Inventive Research in Computing Applications (ICIRCA-2021) DVD Part Number: CFP21N67DVD*; ISBN: 978-0-7381-4626-3, 2-4, September 2021. (Indexed: Scopus) Available at: https://ieeexplore.ieee.org/document/9544935.

[13]. P. Sharma, R. Kumar, and M. Gupta, "Impacts of Customer Feedback for Online-Offline Shopping using Machine Learning," In *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)* (pp. 1696-1703). IEEE, Oct 2021.

[14]. M. Gupta, R. Jain, A. Gupta, and K. Jain, "**Real-time analysis of Covid-19 Pandemic on Most populated countries worldwide**," CMES-Computer Modeling in Engineering & Sciences, This article belongs to this Special Issue: Computer Modelling of Transmission, Spread, Control and Diagnosis of COVID-19). ISSN: 1526-1506, 14th Sept 2020, DOI:10.32604/cmes.2020.012467.

[15]. T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, no. 8, pp.861-874, 2006.

[16]. M. Gupta and D. Dahiya, “Performance Evaluation of Classification Algorithms on Different Data Sets”, Indian Journal of Science and Technology, vol. 9, no. 40, pp. 1-6, DOI: 10.17485/ijst/2016/v9i40/99425, Oct. 2016. ISSN (Print): 0974-6846 ISSN (Online):0974-5645.

[17]. M. Gupta, V. Kumar-Solanki and V. Kumar-Singh, “A Novel Framework to Use Association Rule Mining for Classification of Traffic Accident Severity”, IngenieríaSolidaria, vol. 13, no. 21, pp. 37-44, April, 2017.doi:10.16925/issn.1900-3102.

Malaria Detection Using Blood Smear Images

Nishi Yadav*, Dasari Gayathri, Maddula Sai Sunamdha Harinhi

[1]School of Studies in Engineering and Technology Guru Ghasidas Vishwavidyalaya Bilaspur, India

nyadav.cse@gmail.com, amma.vsp@gmail.com, sunamdhamaddula999@gmail.com

Abstract—Malaria is one of the deadliest diseases. A perilous disease provoked by parasites that can disseminated to people from the nips of infected female mosquitoes which belongs to Anopheles genus. It is preventable and curable. However, Malaria interpretation entail close scrutiny of the blood smear images at 100x enlargement. This is followed by a freehand computing process in which adepts tally the count of Red blood cells influenced by parasites. Perception of images of malaria blood smear is a upgradeable self-activating quick fix which extricate a sea of time for medical sector plod along struggle odds with this pernicious disease. In this work, we set out to recognize from images of blood smear using deep learning methods to predict in case the sample is taken from healthy person or not. Here, we use SVM_HOG a deep learning technique to classify images in Parasitized/Uninfected images in which we use almost 19290 images of cells where it contain similar amount of both infected and uninfected images from the Kaggle database, from image we extracted the HOG Features after extracting features we feed to a classifier SVM to predict whether itis a Parasitized/Uninfected images the accuracy of our model using the data set of malaria blood smear images, we attained an accuracy of 92.69% using Linear SVM as a classifier . The results suggest that it has high accuracy on comparison with other techniques.

Keywords—Images of blood smear, Linear Support_Vector_Machine(SVM), Histogram_Oriented _gradients-HOG features, malaria parasites.

1. INTRODUCTION

Malaria a lethal disease causes because of parasites called malaria vectors and disseminated to people with the nips of infected female mosquitoes which belongs to Anopheles genus. It is avoidable and treatable. The approximate value of people died due to malaria are 409000 in 2019. Children are aged under 5 years the most exposed group affected by malaria; they reckoned about 274000 which is around 67% demises due to malaria at global in 2019.WHO African nation schleps a gratuitous high allowance of malaria worldwide load. African zone was abode to94% malaria victim or demises in 2019[1]. Ambiguous upshots such as False positive clinical outcome give rise to needless avail of antiprotozoal drugs which successively leads to stomach-ache, diarrhoea, illness, vomiting etc. whereabout false negative clinical outcome causes superfluous use of medication, subsequent confirmation, and which boost dreadful conditions of malaria [2]. So, the evolution of self-moving procedure for treatment of malaria is an enchanting probing intent for modifying each and every patient treatment and controlling. Automatic parasite spotting has large head start such as it can provide a betterand reliable treatment, especially when limited resource available, and clinical costs will decrease. Each slide of blood on microscopic probe, as well as measurable parasite perception and genus identification takes a specialist in microscopy around fifteen to thirty proceedings. In a view of 100-1000 of blood smears are scrutinized manually on an annual basis this leads to a enormous economic battle needed for diagnosis of malaria.[3] Detection of malaria in earlier stage helpful for treatment of patient and can reduce dangerous consequences. Giemsa stain on microscopic inspection of malaria parasite is glaring [4]. Other copious techniques such as expeditious diagnostic trials, reaction of polymerase chain in blood smears to identify on the presence of antigens. Even though different tests surmount in malaria identification, anyhow microscopy is ubiquitous due to inexpensive and less cumbersome and its success rely upon pathologist prowess [5].

2. RELATED WORKS

Diaz [6] This paper is about categorization images of blood smear automatically, techniques of machine learning are used e.g., Diaz et al. blood smear pictures categorization utilizing a Support_Vector_Machine (SVM) to spot stage of their infection the contaminated red blood cells. This technique was given out good results with 94.0 percent sensitivity on a data which contains four fifty pictures. Tek [7], this paper is about computer vision- on malarial parasite identification studies eg., Tek et al. made a use of developed KNN following normalization and correcting of colour among nine blood input picture films for binary categorization. In this paper [8] they proposed a CNN architecture which was stacked which make detection of the malaria much better using procedures like five- folded cross validation on around 27558 pictures which consists of equal amount of both healthy and infected pictures of cell and they got an accuracy of 99.98 percent with a hundred percent precision and 99.9% recall. This paper [9] proposes the CNN based-model on sixteen layers parasite of malaria detection which differentiate the blood cells as contaminated or decontaminate. This framework gained 97.0 percent accuracy, and a higher specificity and sensitivity of than learning was upskilled with roughly 27000 images with image size forty-four x forty-four pixels. The recognition of smears of blood for plasmodium using concentrated Leishman stain on stacked images proposed by Gopa kumar et al., [10] with an architecture that is customized CNN gave a result of 97.0% sensitivity (97.0%) and specificity (98.0%). The paper [11], the malarial parasite automatic observation was explained by evaluating at patient level and thumbnails with gross 94.1% specificity ,89.7%precision, and 89.7% sensitivity to upgrade the user confidence in system findings. This paper [12] the main attention is using convolution neural network to find full count of pictures of blood smear. If malarial pathogens are available, the network is upskilled to spot them. Demonstrations specifies that the mean average precision of the gross production of the system over 0.95 when contrasted with the base truth. Additionally, the system forecasts the images containing malarial parasites as contaminated 100percent of the time. For quick prototyping, they ported the software to minimum priced microcomputer. This paper [13] suggests an artificial neural network combined with autoencoder that is stacked sparsely they used softmax classifier with 2 nodes in output layer along with they used 10-fold cross validation which proves that it work with new dataset using they got an accuracy of around 89.10% and sensitivity 93.9% and specificity of 83.1%. In this research [15], a reduction approach of a hybrid dimension suggests a algorithm which is genetic and optimize to get a proper subset of features from available data. Kernel classifiers used is Support_Vector_Machine's (SVM) classifier utilized the reduced malaria vector dataset to assess the classification performance of the experiment. In this paper [16] with the help of photo acoustics surface acoustic wave sensor (PA_SAW) malaria can be treated in earlier stage. And its sensing system categorize ordinary blood from the blood which is infected of one percent concentration is identified. In this work [17] they concentrated on study of mosquitoes which spread malaria at first, they collected mosquitos' wings as well as bodies scattering properties and observed 808 nm polarized near infrared light befit for identifying Anopheles mosquitos' wings. In this [18] research they proposed a sensor which is portable, reusable which is sensitive and available at low price it identifies hemozoin pictograms at (12.7,25.4) it mainly deals with varying hemoglobin magnetic property due to parasite of malaria. In this paper [19], they used pipeline with image size of large 5312×2988 for identification of red blood cells and adding up images of thin blood smear called RBCNET they used 2 lakh labeled cells of 965 images from around 193 patients which give high true positive rate. In this paper they concentrated on detecting hemozoin at less concentration. At first for malaria treatment, they reviewed a lot of Raman spectroscopy methods with sample postprocessing they confirmed the Nano silver particles after disintegration of parasite and RBC cells that helps to judge quality of antimalaria drugs. In [20] [21], the authors used image data analysis for finding healthcare disease using machine learning technique.

3. MATERIALS AND METHODS

In this part we extracted features of image using HOG, or Histogram of Oriented Gradients, which is used very often for getting the features from image as data and followed by we feed up our model against SVM classifier for malaria parasite detection on malaria blood smear images.

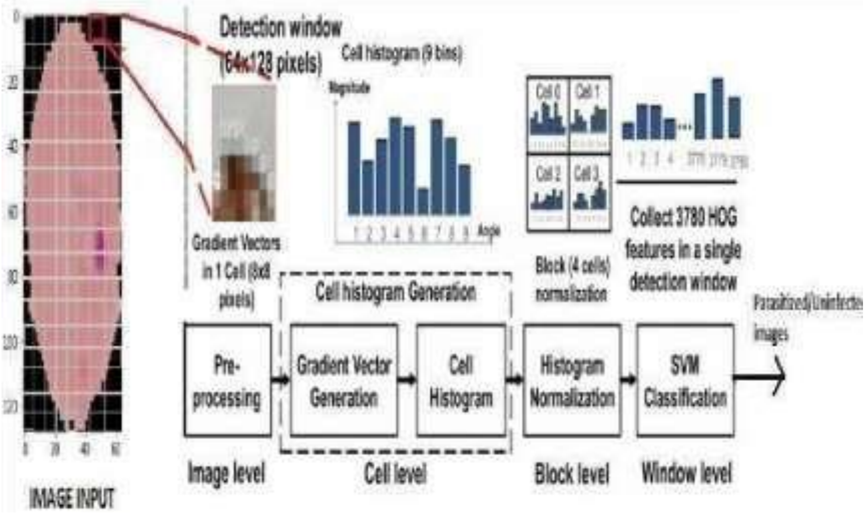


Fig-1 Hog_Svm for perception of malaria [14]

3.1.*Dataset and Preprocessing*

- **Dataset Collection**

The images used in making both the training validation datasets are a collection of 19290 blood smear images that are acquired from Kaggle datasets.

- **Data Preparation**

We created cell images folder contains all the images of the dataset and the file train.csv contain image names belonging to dataset and their corresponding labels that is Parasitized/Uninfected and set the base directory for reading images as all the images of the dataset. It is observed that our train set consists of equal samples of both the classes thus we will not face any problem due to class imbalance in the dataset. since having textual labels for our images i.e., parastized/Uninfected in the train.csv file so we have to change them to numerical labels i.e. 0 or 1. And we divide the dataset into two parts such as training and validation sets. Training set is the subset of the dataset that is used for training our model when it comes to validation set tells whether our model working upto the mark of the model after each epoch. We can observe there are images of different shapes in fig-2. It is necessary to have images in shape size before going ahead with modeling process and it is also dependent on which feature extractor tool. <Figure size 1080x1080 with 0 Axes>

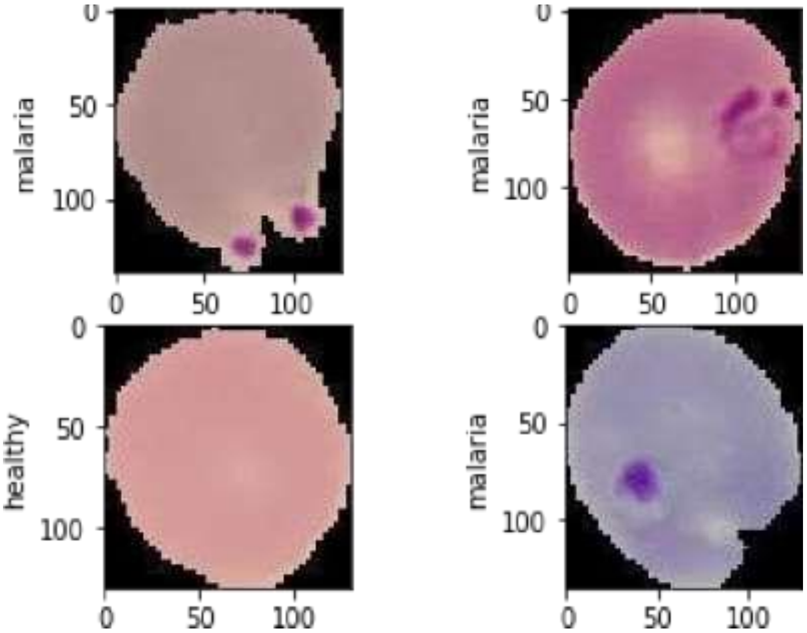


Fig-2. Images of cells

3.2. ***HOG_ Features and Linear_ SVM***

Extraction of Images: Histogram of Oriented Gradients (HOG) is an identifier of a feature which is used often for feature extraction from images. In fields such as computer vision HOG entrenched. At begin Histogram_of_oriented_gradients features put to use by Navnet Dalal which highlights finding the of human. However dissimilar from other geometry features which considered integrity of an image, HOG feature, at first divides the picture into small parts known as cells and after that we calculate gradients and accumulates the histogram of gradient of the cell over each pixel. The whole blocks concatenated histograms form the HOG vectors. For extracting HOG feature primarily, the image has to be made into small pieces and for each piece we calculated gradient magnitude as well as gradient direction. We then plot Histogram using the Gradient magnitude and direction.

“The descriptor of HOG feature determines the gradient orientation contingency in the image localized portions.”

- Techniques of calculating the Histogram of Oriented Gradients

From the image below which is of 128x128 we have to extract Hog features.

- Preprocess the Data

Before extracting of features from image we need to resize it to 64x128 as shown in figure 3. after that we divide the image into 8*8 and 2*2 patches in both vertical and horizontal respectively.

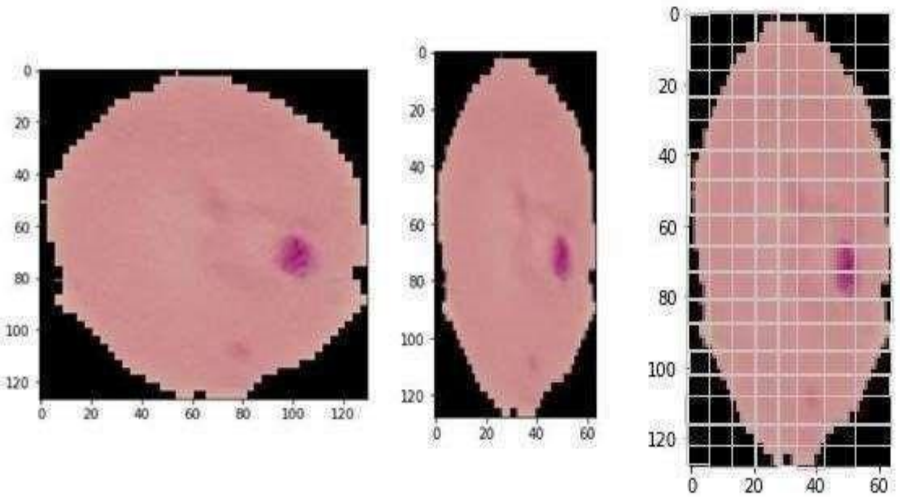


Fig-3. Preprocessed (64*128) and made(8*16) cells

Gradients Calculation in both X and Y directions: For calculation of gradient for every patch as shown in Fig 4. Variation in x and the y directions treated as gradients. From the highlighted patch of the image, we calculated the gradients. And from that we generated pixel matrix the matrix shown below is just for explanation they are not actual pixels.

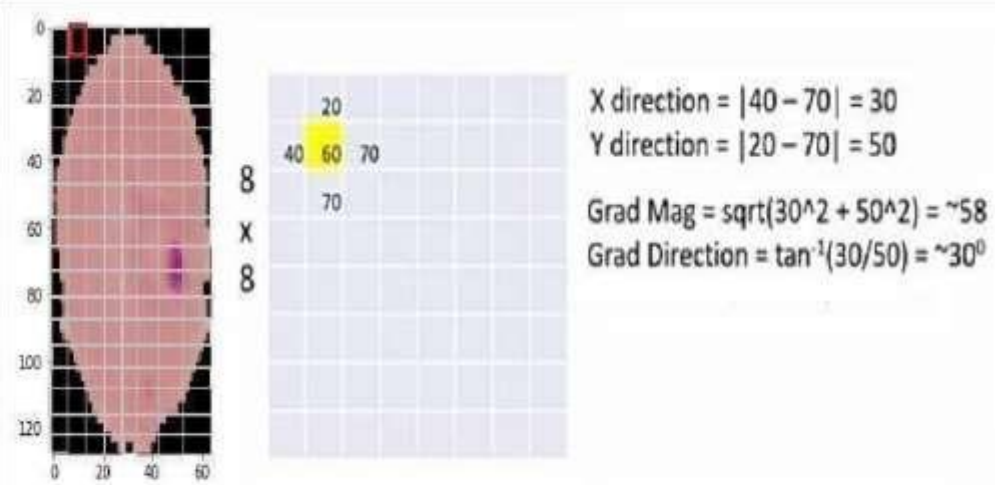

20
40 60 70
70
8
x
8
X direction = |40 – 70| = 30
Y direction = |20 – 70| = 50
Grad Mag = sqrt(30^2 + 50^2) = ~58
Grad Direction = tan⁻¹(30/50) = ~30⁰

Fig-4. Measuring gradients magnitude and their direction

A histogram plot shown on fig-5 is the continuous distribution of a group of data. We took variable bins on x axis and their frequencies on y-axis. In Short Magnitude on y- axis and orientation x-axis. This is how histograms are as shown in fig 6.

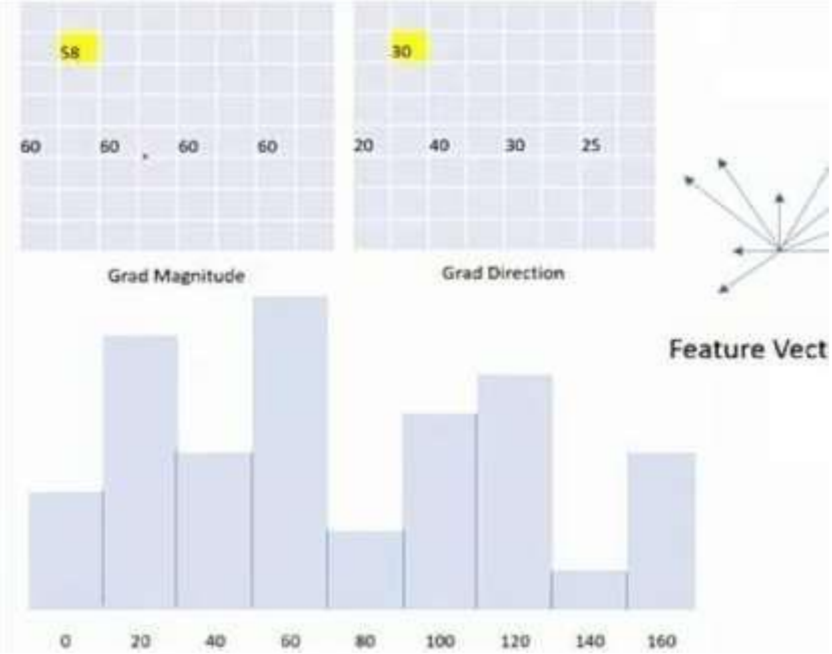
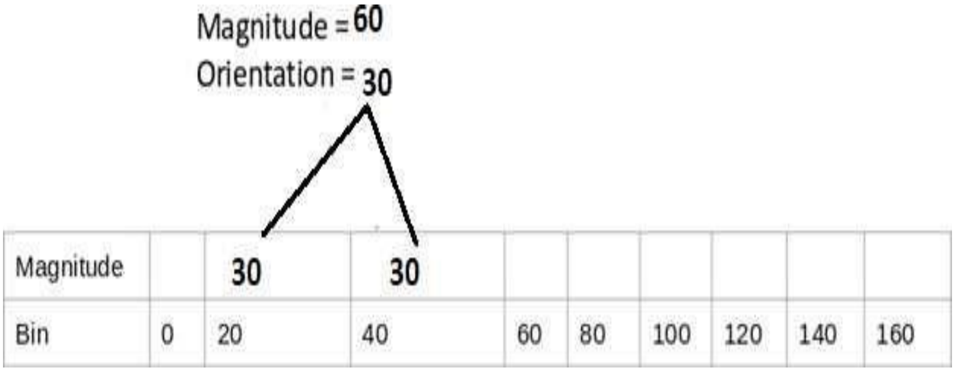


Fig-5: Histogram



Magnitude		30	30						
Bin	0	20	40	60	80	100	120	140	160

Fig-6. Measuring magnitude and orientation.

1) *Calculation of Histogram of Gradients in 8×8 cells and conversion into 9×1 vector.*

The single patch is divided into 8x8 matrix and from that patch we generate pixel matrix after generation we calculate gradient orientation as well as direction after which we plot a histogram with nine bins which is none another than vector of size (9*1) matrix.

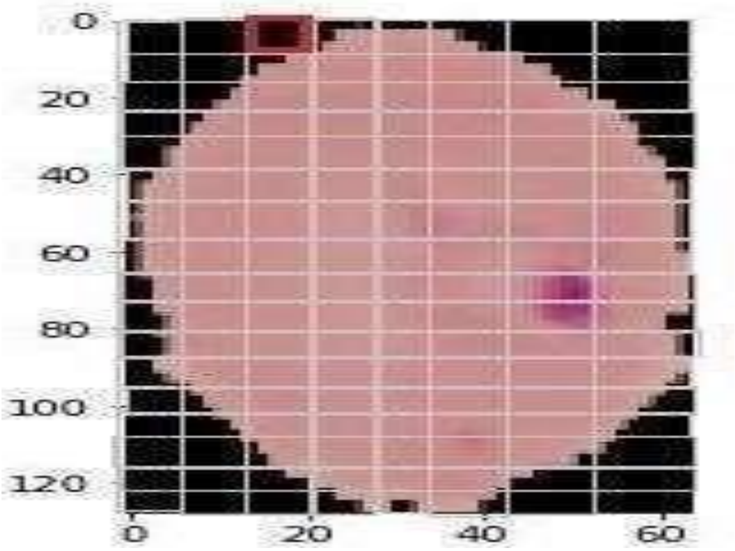


Fig-7. Highlighting single cell next important step is to normalize the histogram. Such that change in brightness won't affect our results.

- Normalize the gradients to (36x1) pixel matrix that is 16×16 cell : As of now, the calculated HOG features for the patch having 8×8 cells of blood smear, the gradients of the image are tactful to the altogether brightness. It is none another than particular portions of the image is brighter compared to other. As it is not possible to eliminate this problem completely but we can reduce it some extent with the help of normalization by considering 2*2 patches having pixel matrix of size 16*16 as shown in fig-8:

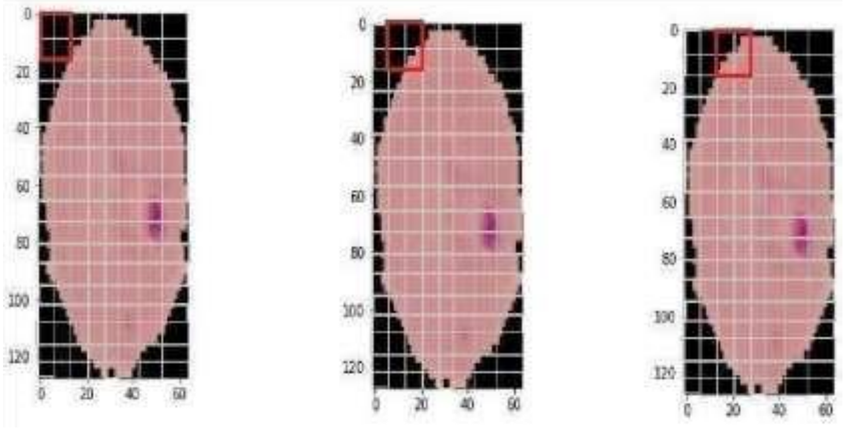


Fig-8 Highlighting (2*2) patches

So here we have to combine four 8x8 matric esto form a big matrix of size 16x16. As of now from one 8x8 matrix we get a vector of size nine and from four matrices we geta vector of size 9*4=36, we can also call this vector as a matrix of36x1. For normalizing we divide the matrix with root of sum of squares of the numbers . Given vec9tormathematically can be written as. a.

V=[b1,b2,b3,….b36]

At first we have to calculate the below equation–(2) where we root value of sum of squares:

b. p= √(b1)2+ (b2)2+ …. +(b35)2+ (b36)2

After getting we use this p to divide each andevery value of vector:

c. Vector after normalization={b1/p,b2/p,b3/p……,b36/p }

The result is a vector which is normalized of 36x1size.

- Features Of Full image: Now we reached the last step for the calculation of Features of Hog. Till now we have generated features of a 16x16 matrix size. Now, we have to combine all of these features to acquire the final image. At first we have to find out total number of such 16×16blockswe would get for a image of 64×128 as shown in fig-9.

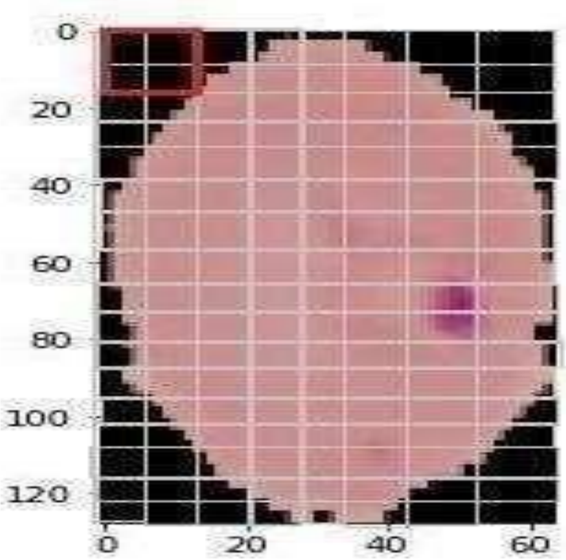
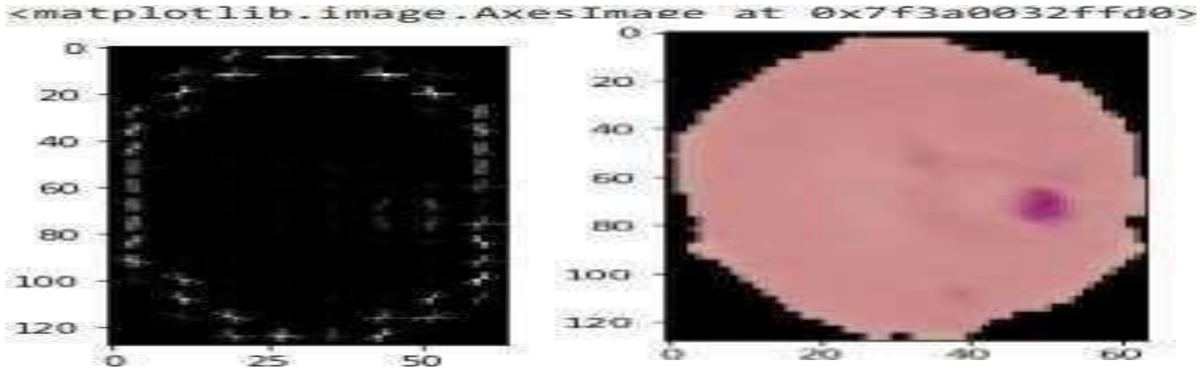


Fig-9. Highlighting (2*2) patches

In horizontal we get seven and in vertical we get fifteen in totalwe get 7x15 =105 pixel matrices of size 16x16 from each matrix we get a vector of size 36x1. So in total we get 36x105=3780 features. From below image we can see the extracted HOG features as shown in fig-10. Finally, we feed Hog_ features for linear SVM for classification of images as Parasitized/Uninfected.



```
<matplotlib.image.AxesImage at 0x7f3a0032ffd0>
```

Fig-10. Hog_features and its cell image

4. RESULTS AND CONCLUSION

Other Deep Learning Based model comparatives

FLANN + SSAE MALARIA PARASITE DETECTION METHOD

The trained their model with the FLANN – SSAE techniques for malaria parasite detection from images of blood smears. For detection of malaria parasite, they proposed a CAD model to identify whether the image is taken from a infected person or not [13].

4.1. Experiments and Results

The Support vector Machine (SVM) using HOG features and the proposed CAD model both of them got judged using malaria blood smear images data set.

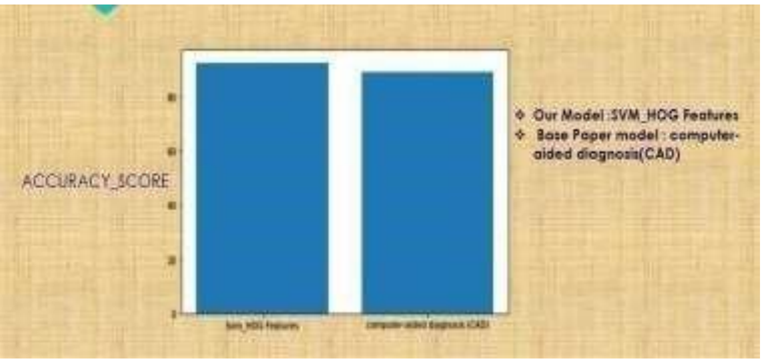


Fig-11 Accuracy-score

TABLE 1. Accuracies deep learning techniques such as Support vector Machine (SVM) using HOG features and proposed CAD scheme.

Deep Learning Techniques	Accuracy (%)
Support vector Machine (SVM) using HOG T features	**92.69%**
hProposed CAD e scheme (Base Paper) [13]	**89.10%**

AUC ROC Curve

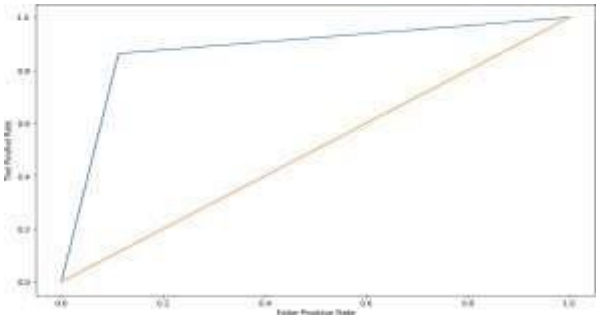
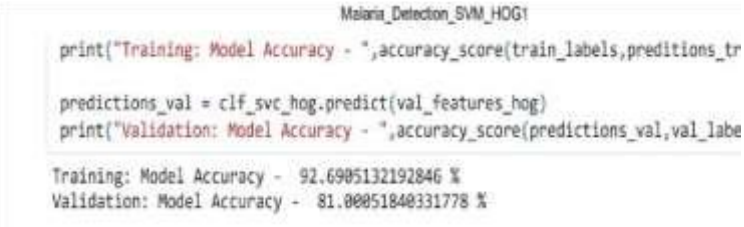


Fig-12. AUC-ROC CURVE

AUC ROC graph or curve the malaria dataset using the Support vector Machine (SVM) using HOG features representing the blood smear image and area under curve give the accuracy of our model as shown in fig- 11.



Malaria_Detection_SVM_HOG1

```
print("Training: Model Accuracy - ",accuracy_score(train_labels,preditions_tr

predictions_val = clf_svc_hog.predict(val_features_hog)
print("Validation: Model Accuracy - ",accuracy_score(predictions_val,val_labe
```

```
Training: Model Accuracy -  92.6905132192846 %
Validation: Model Accuracy -  81.00051840331778 %
```

Fig-13. Result obtained using SVM_HOG features

5. CONCLUSION AND FUTURE WORK

The machine learning methods. We used such as HOG Features, SVM have shown limited accuracy for malarial parasite detection from blood smear pictures which is around 92% ,So as part of future work will we try better feature extraction techniques and will train our model on more deep learning techniques and we also try to ensembles different machine learning methods to improve the accuracy.

REFERENCES

[1] World Health Organization. World malaria report 2019 (World Health Organization), 2020.

[2] Poostchi, M., Silamut, K., Maude, R. J., Jaeger, S., & Thoma, G. (2018). Image analysis and machine learning for detecting malaria. *Translational Research*, *194*, 36-55.

[3] Arco, J. E., Górriz, J. M., Ramírez, J., Álvarez, I., & Puntonet, C. G. (2015). Digital image analysis for automatic enumeration of malaria parasites using morphological operations. *Expert Systems with Applications*, *42*(6), 3041-3047.

[4] Makhija, K. S., Maloney, S., & Norton, R. (2015). The utility of serial blood film testing for the diagnosis of malaria. *Pathology*, *47*(1), 68-70.

[5] World Health Organization. (2016). *Malaria microscopy quality assurance manualversion 2*. World Health Organization.

[6] Díaz, G., González, F. A., & Romero, E. (2009). A semi-automatic method for quantification and classification of erythrocytes infected with malaria parasites in microscopic images. *Journal of biomedical informatics*, *42*(2), 296-307.

[7] Tek, F. B., Dempster, A. G., & Kale, I. (2010). Parasite detection and identification for automated thin blood film malaria diagnosis. *Computer vision and image understanding*, *114*(1), 21-32.

[8] Mustafa, W. A., Santiagoo, R., Jamaluddin, I., Othman, N. S., Khairunizam, W., & Rohani, M. N. K. H. (2018, August). Comparison of detection method on malaria cell images. In *2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA)* (pp. 1-6). IEEE.

[9] Umer, M., Sadiq, S., Ahmad, M., Ullah, S., Choi, G. S., & Mehmood, A. (2020). A novel stacked CNN for malarial parasite detection in thin blood smear images. *IEEE Access*, *8*, 93782-93792.

[10] Gopakumar, G. P., Swetha, M., Sai Siva, G., & Sai Subrahmanyam, G. R. K. (2018). Convolutional neural network-based malaria diagnosis from focus stack of blood smear images acquired using custom-built slide scanner. *Journal of biophotonics*, *11*(3), e201700003.

[11] Mehanian, C., Jaiswal, M., Delahunt, C., Thompson, C., Horning, M., Hu, L., ... & Bell, D. (2017). Computer-automated malaria diagnosis and quantitation using convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 116-125).

[12] Chowdhury, A. B., Roberson, J., Hukkoo, A., Bodapati, S., & Cappelleri, D. J. (2020). Automated complete blood cell count and malaria pathogen detection using convolution neural network. *IEEE Robotics and Automation Letters*, *5*(2), 1047-1054.

[13] Pattanaik, P. A., Mittal, M., & Khan, M. Z. (2020). Unsupervised deep learning cad scheme for the detection of malaria in blood smear microscopic images. *IEEE Access*, *8*, 94936-94946.

[14] Nguyen, N. D., Bui, D. H., & Tran, X. T. (2019, November). A novel hardware architecture for human detection using HOG-SVM co-optimization. In *2019 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)* (pp. 33-36). IEEE.

[15] Arowolo, M. O., Adebiyi, M. O., Adebiyi, A. A., & Okesola, O. J. (2020). A hybrid heuristic dimensionality reduction methods for classifying malaria vector gene expression data. *IEEE Access*, *8*, 182422-182430.

[16] Wang, S., Yang, C., Preiser, P., & Zheng, Y. (2020). A Photoacoustic-SurfaceAcoustic-Wave Sensor for Ring-Stage Malaria Parasite Detection. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *67*(5), 881-885.

Jansson, S., Atkinson, P., Ignell, R., & Brydegaard, M. (2018). First polarimetric investigation of malaria mosquitoes as lidar targets. *IEEE Journal of Selected Topics in Quantum Electronics*, *25*(1), 1-8.

[17] Hole, A. P., & Pulijala, V. (2020). An Inductive-Based Sensitive and Reusable Sensor for the Detection of Malaria. *IEEE Sensors Journal*, *21*(2), 1609-1615.

[18] Kassim, Y. M., Palaniappan, K., Yang, F., Poostchi, M., Palaniappan, N., Maude, R. J., ... & Jaeger, S. (2020). Clustering-based dual deep learning architecture for detecting red blood cells in malaria diagnostic smears. *IEEE Journal of Biomedical and Health Informatics*, *25*(5), 1735-1746.

[19] Chen, K., Perlaki, C., Xiong, A., Preiser, P., & Liu, Q. (2016). Review of surface enhanced Raman spectroscopy for malaria diagnosis and a new approach for the detection of single parasites in the ring stage. *IEEE Journal of Selected Topics in Quantum Electronics*, *22*(4), 179-187.

[20] M. Gupta, H. Wu, S. Arora, A. Gupta, G. Chaudhary, and Q. Hua, "Gene Mutation Classification Through Text Evidences Facilitating Cancer Tumour Detection," *Journal of Healthcare Engineering, Hindawi, pp. 1-16, vol. 2021, Doi:* https://doi.org/10.1155/2021/8689873.

[21] R. Jain, M. Gupta, S. Taneja, and J. Hemanth, "Deep Learning based detection and analysis of COVID-19 on Chest X-Ray Images," Applied Intelligence (APIN), 10.1007/s10489-020-01902-1, APIN-D-20-01185R3, Oct 2020

Sentiment Analysis: An Assessment of Diverse Methods

Hrithik Goswami, Vaibhav Gupta, Rachna Jain

[1]Department of Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, Delhi, India hrithikgoswami.cse1@bvp.edu.in, vaibhavgupta.cse1@bvp.edu.in, rachna.jain@bharatividyapeeth.edu

Abstract: Digitalization over the years has greatly impacted the inevitability of consumer reviews in the online sphere. Analysing a review given to a product has always been a crucial need, and these reviews are very vital as they shape the overall product, thereby allowing the customer to gain hindsight about the product that they might intend to buy. But a single product can itself have a colossal number of reviews, and thus it becomes very difficult at times for the customer to choose a product. Therefore, if there is a suitable mechanism that can help the buyer and seller to analyze the products, then it can greatly solve the problem of decidability. Hence, we have carried out this research in which we compared five machine learning classifiers: Multinomial Naïve Bayes, Support Vector Machine, Logistic Regression, Decision Trees, and Random Forest Classifier, on the Amazon phone reviews. We utilized the feature extraction technique of TF-IDF to convert the textual data into numerical form and used evaluation metrics such as precision, recall, f1-score, and accuracy to assess our models. Our evaluation and analysis show that a Random Forest gives the best possible suitable result for the chosen data; this was additionally evaluated by tuning the hyperparameters of the Random Forest using out-of-bag error and 3-fold cross-validation techniques, and it showcases an improvement in accuracy with the former method.

Keywords: Sentiment Analysis; Mobile Phone Reviews; TF-IDF; Multinomial Naïve Bayes; Support Vector Machine; Logistic Regression; Decision Tree; Random Forest Classifier.

1. Introduction

In the ever-growing and evolving virtual world, the needs of people have drastically shifted towards the online marketplace, and they also rely mainly on reviews to make a final decision. And not only that, but organizations also have a greater dependency on reviews as it helps them to improve and gives them a chance to meet the needs of the consumer and thus enhance their products. As there are enormous reviews available, there is a huge need to come up with a method that would classify and understand these reviews in a real sense, thereby helping the customer by giving them a general idea about the product. And for this, sentiment analysis and classification play a key role by extracting the important sentiments given in the form of the reviews and classifying them accordingly. Sentiment Analysis is a process that uses biometrics, natural language processing, computational linguistics, and analysis of text to extract information conveyed in a chunk of text that defines the human sentiments delivered through it [1]. In this paper, we have focused on this issue and have worked on the phone reviews given on Amazon.com, as it is one of the biggest e-commerce platforms that provide a range of mixed reviews. We worked with various classifiers and did sentiment analysis of the reviews and classified them as positive, negative, or neutral, thus helping the producer and consumer by providing them with a state of mind about the products.

With the advent of growth in the e-commerce market, the reliability of online reviews has considerably increased over the past years. It has become very crucial to classify the reviews to meet the deeper needs of both the buyer and seller on the online platform. The classified reviews can thereby greatly help to form a mindset for the product, as solely relying on the whole reviews, which are in huge numbers, is a critical task to comprehend [2]. So, in this research, we have worked on this challenge by finding the polarity of a review about a product to estimate the correctness of classification algorithms using several assessment metrics. In addition to this, methods like out-of-bag error and cross-validation were also applied to the best classifier to tune its performance and test it on the desired measures.

This paper is outlined as follows: in section 2, we discussed various literature reviews that have worked towards a similar problem. In section 3, we discussed the various machine learning methods used to carry out a comparative analysis. Section 4 focuses on the data, its features, methodology, and implementation. In sections 5 and 6, we have given the experimental results obtained, and a comparison of the same is also done. In section 7, analysis of the best classifier is carried out based upon the experimental results attained in previous sections. Finally, we concluded the findings of our paper with possible future scope in section 8.

2. Related Work

Several distinct but similar work in this field on a variety of data has been done in the recent past, and they have been considered in this review.

Callen Rain in [3] utilized the current work done in the area of NLP on the reviews on Amazon and used Naïve Bayesian and Decision list classifiers for sentiment analysis and also compared features like bag-of-words and bigrams for their efficacy. Their analysis showed that the Naïve Bayes classifier worked well with over 800 features as well as the highest accuracy was also obtained with it. K. Ghag and K. Shah in [4] did a comparative investigation of sentiment analysis on the detection of the polarity of tweets as positive, negative, and neutral using the lexicon and non-lexicon methods and realized that the sentiment analyzers are centered around the language, with managing negation and language generalization being the major problems. Xing Fang and Justin Zhan in [5] worked on the problem of sentimental polarity categorization on Amazon product reviews with diverse classification algorithms like the Random Forest, Naïve Bayes, and Support Vector Machine, where the performance of each was studied based upon their ROC curves and f1-score metrics. Both the classifiers used by them, viz. Naïve Bayes and SVM were observed to be better than the Random Forest. Muhammad T. Khan et al. in [6] discussed the various sentimental analysis techniques and emphasized the numerous challenges that are faced with natural language processing. Mohan Kamal Hassan et al. in [7] did a sentimental analysis of the laptop product reviews on Amazon.com using Naïve Bayes. The above analysis showed us that it performed optimally with bigrams and stop words as compared to single words with an accuracy of approximately 90% for over 10000+ samples.

Heide Nguyen et al. in [8] used three machine learning and three lexicon-dependent techniques to carry out the sentiment analysis of product reviews on Amazon. The assessment showed that all the three former models outperformed the latter models on all the evaluating metrics: precision, recall, and f1-score. Abhilasha Tyagi and Naresh Sharma in [9] used Logistic Regression with a unigram feature vector to perform sentiment analysis on the data of Twitter by speeding up the classification process. They applied a useful word score heuristic to obtain the scores of frequently used words. Wanliang Tan et al. in [10] used traditional and modern machine learning methods, viz. Naïve Bayes, K-Nearest Neighbour method, Recurrent Neural Network (RNN), Support Vector Machines, etc. to perform sentiment analysis of product reviews on Amazon, and LSTM gave the best results. Momina Shaheen et al. in [11] mined the mobile-phone product reviews from Amazon to predict the ratings as positive and negative, and lastly, they did a comparative analysis of eight classifiers, of which the Random Forest showed the best result with 85% accuracy. Sara A. Aljuhani and Norah S. Alghamdi in [12] carried out a contrast of different algorithmic methods, namely Logistic Regression, Stochastic Gradient Descent, Naïve Bayes, and Convolutional Neural networks (CNN) of mobile phone reviews on Amazon and found that the convolution neural network with word2vec gave the best results with 92.72% accuracy for unbalanced data and 79.60% with balanced data. They also used the Lime technique for assessing the possible logical explanations for the reviews being classified into different polarities.

Jayakumar Sadhasivam and Ramesh B. Kalivaradhan in [13] used an ensemble approach with the currently existing models, viz. Naïve Bayes and SVM, to perform sentiment analysis on Amazon reviews available on the official product site, and then the product is recommended based on the analysis. Hui Zhang in [14] analyzed the Amazon Alexa reviews to study the sentimental aspect of the nature of such reviews by working with Naïve Bayes and Logistic Regression classifiers. The analysis predicted that Logistic Regression had performed slightly better than Naïve Bayes with an accuracy of 87.4% as compared to 87.1% for Naïve Bayes on the unbalanced dataset. The dataset, which was unbalanced in nature, was balanced by the SMOTE technique, which led to advancing the ROC curve (AUC) scores of these two models from 0.5 to 0.8. Emilie Coyne et al. in [15] discussed the performance of three algorithms, namely Multinomial Naïve Bayes, Long Short-term Memory network (LSTM), and Linear Support Vector Machine (LSVM) based upon the sentimental analysis of 60,000 product reviews selected randomly from Amazon.com. The analysis determined that LSTM performed better among them, with an accuracy of 90%. The best results with LSTM were achieved on the remaining 3.94 million reviews with an accuracy of 92%. Vineet Jain and Mayur Kambli in [16] discussed the sentimental analysis on Amazon product reviews with different supervised and unsupervised machine learning techniques and models like Naïve Bayes, Logistic models, etc. were evaluated based upon their bag of words accuracy and TF-IDF scores, where both of these models performed similarly. K. Ashok Kumar et al. in [17] used supervised machine learning methods to perform sentiment analysis of Amazon product reviews, and their model is capable of determining whether the consumer intends to propose the product or not.

3. Machine Learning Methods

To carry out our research, we have done a comparison of five diverse methods on reviews to assign them with different polarities by building a prediction model on the Amazon mobile phone reviews dataset.

A. Multinomial Naïve Bayes

This model is a widely used classifier for the classification problem that has discrete features. It follows a probabilistic approach in which the feature vectors signify the count of frequencies, and certain events are produced by a multinomial. If we have a class y with n features, then the distribution is parametrized by vectors $\theta_y = \theta_{y1}, \theta_{y2}, \ldots, \theta_{yn}$ where θ_y is assessed using relative frequency counting:

$$\hat{} \quad N_{yi} + \alpha\, N_y + \alpha n$$

$$\theta_{yi} = \underline{\qquad\qquad}$$

Where in a sample of class y, θ_{yi} is the probability $P(x_i \mid y)$ of the feature i; N_{yi} is the total occurrences of i in y and for class y, N_y is the cumulative count of all the features, with α being the smoothing priors [18, 19].

B. Support Vector Machine

This model is a robust supervised learning algorithm that is created on a learning framework and is based upon statistics. This model is mainly used for classification problems. If we plot different groups or classes of data in an n-dimensional space then SVM performs classification by finding a hyperplane in this space, that differentiates the class of data. And it draws these hyperparameters by transforming the data using kernels.

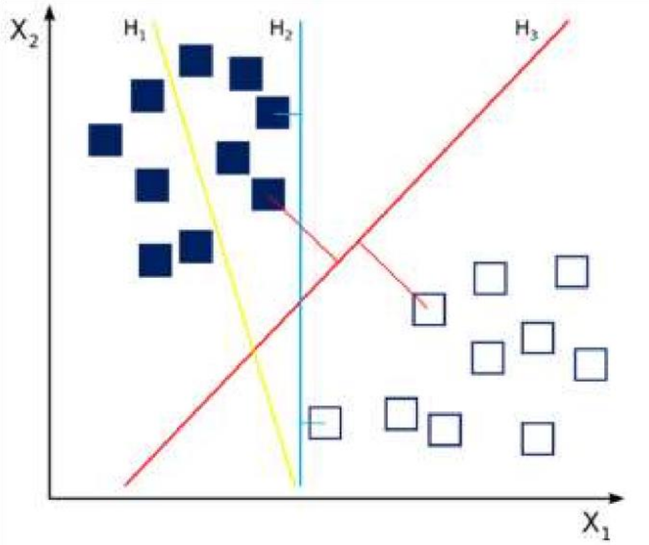


Fig. 1. Classification with the help of SVM.

The hyperparameter having the largest margin or distance from the classes of data is then chosen as the best hyperparameter [20]. In Fig. 1, the squares represent the support vectors with H_1, H_2, and H_3 as the margin classifiers. The H3 plane separates the data with the largest margin and therefore correctly divides the data.

C. *Logistic Regression*

It is a predictive model that is significantly used for predictive analysis and in cases when the target variable is categorical [21]. And for our research, we have specifically used Multinomial Logistic Regression that classifies the review as positive, negative, or neutral by re-running the binary classification for each class multiple times. This method is implemented by choosing a threshold value that helps in differentiating a class of data and thereby helps in the analysis of the reviews.

D. *Decision Trees*

It is one of the most basic and useful predictive models that can be used as a regressor or a classifier and has a tree structure in which each node signifies the test value of a certain attribute, each edge links to the result of a test, and it joins directly to the next node. The terminal nodes are the end nodes of the tree that ultimately predicts the sentimental outcome conveyed through the reviews. These work on the principle of binary recursive partitioning, where the data is split into partitions and then into branches [22].

E. Random Forest

It is a model that can also be used as a regressor or a classifier, but it contains a collection of decision trees as an ensemble. And this is a much more efficient model than the Decision Tree, as a group of decision trees will outperform to give a much better result. For this, it makes use of bootstrapping and bagging and has two pre-requisites: one, there should be an actual signal in the features, and second, the predicted values of each decision tree should have less correlation with each other [23]. If we take an individual tree, then during the splitting of each node, every feature is taken into consideration, and the feature that has a significant number of separations between the observations on the left and the right node is preferred. In the Random Forest, each separate tree can choose from a subgroup of features, and hence this results in a lower correlation across many trees.

4. Data and Methodology

Here, we have unveiled the methodology and procedure utilized to classify the polarity of mobile phone reviews. The dataset is categorized into two groups, viz. training and testing.

The former set is used to understand the classifier, whereas the latter one is used to test and assess the score of our classifier. And Fig. 2 shows the methodology followed.

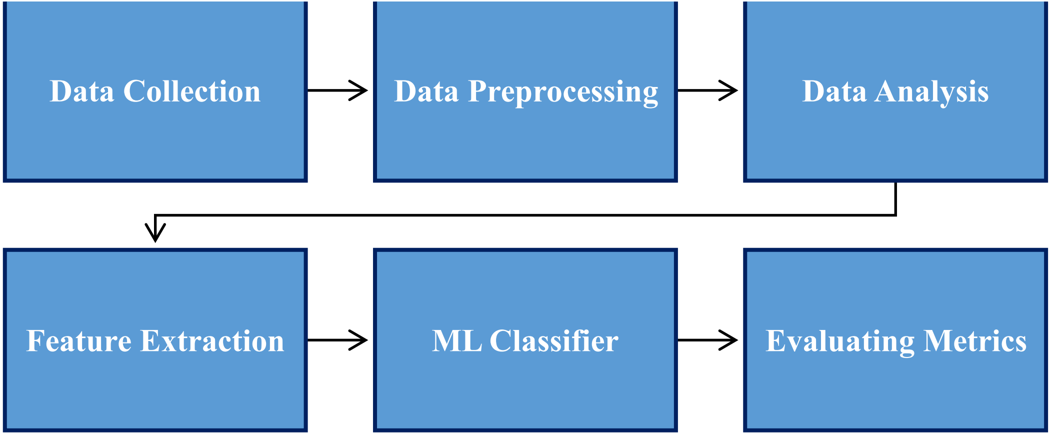


Fig. 2. Approach followed.

a. Data Collection

For this research, we chose the Amazon reviews of unlocked mobile phones, and they are available on Kaggle. This dataset file is available in a comma-separated values (CSV) format. The dataset has more than 400,000 reviews of different variety of unlocked mobile phones, and it primarily consists of 6 columns, namely:

i.*Product Name:* This column contains the name of the product. For example, the Sprint EPIC 4G Galaxy SPH-D7.

ii.*Brand Name:* This column contains the brand of the corresponding product. For example, Samsung.

iii.Price: This column contains the cost of the product. For example, the cost of the Sprint EPIC 4G Galaxy SPH-D7 is $199. iv.*Rating:* This column contains the rating of the corresponding product in a range of 1 to 5.

v.Reviews: This column gives the description of the user experience that he/she has given to the product on Amazon.

vi.*Review Votes:* This column contains the number of people who find these reviews useful. ***b.***

Data Preprocessing

After the data is collected, it is pre-processed for further analysis. This step involves multiple procedures that are carried out to ensure efficient working with data. It involves 6 steps:

i. Tokenization: In this, we separate a piece of text into smaller units which are termed tokens. These tokens can be words, characters, sub-words, phrases, and symbols in which we discard the punctuation marks to allow simpler and efficient analysis.

ii. Removing Stop Words: Here, we discard all stop words, that do not convey significant importance to the structure of the input sentence (review), and therefore helps in increasing the total efficiency of data preprocessing. iii. *Conversion to Lower Case:* In this step, all the upper-case words were converted to lowercase to avoid ambiguity in the data. iv. *Stemming:* Now in this step, we reduce the words into a root, also known as a stem, to allow effective working with the data. Basically, in this step, we remove the unnecessary suffix and thereby increasing the accuracy of the classification model.

v. *Removing Punctuations:* In this step, all the punctuation marks, like a full stop, comma, colon, etc., are removed. vi. *Labeling the Data:* Finally, in this step, we categorize the column '*Rating'* into 3 parts:

positive (labeled as 2), neutral (labeled as 1), and negative (labeled as 0).

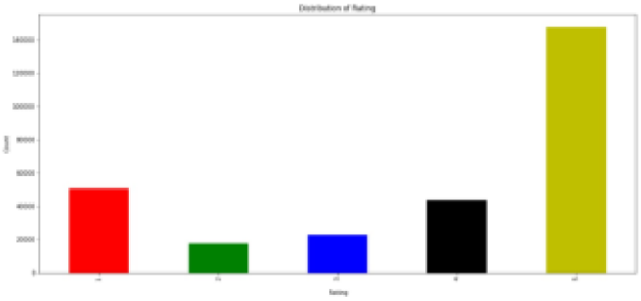


Fig. 3. Distribution of rating from 1–5 with respect to their count.

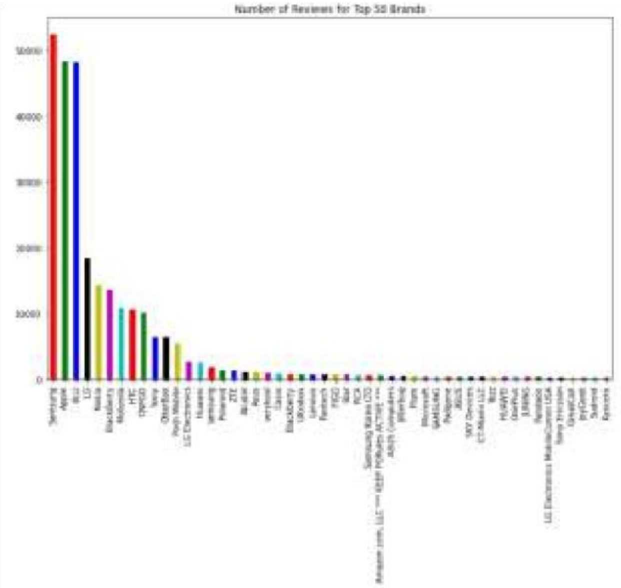


Fig. 4. The number of reviews for top 50 brands.

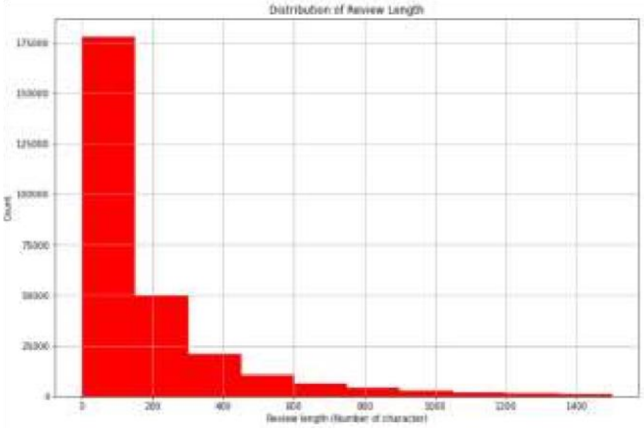


Fig. 5. Distribution of reviews length and their rating.

c. Data Analysis

After the pre-processing, the data is analyzed for further action. And we carried out the analysis in the following ways:

i.*First,* we analyzed the inclination and distribution of the ratings and observed that most of the ratings, which are approximately 140,000+, were given 5 stars for a variety of mobile phones. This rating is based on the scale of 1–5 of Amazon's rating scale. This is depicted in Fig. 3.

ii.*Second,* we analyzed the number of reviews that were given for the top fifty brands and observed that most reviews were given to 'Samsung', followed by other brands, with over 50,000+ reviews given to a variety of 'Samsung' products and approximately 45,000+ reviews were given to the 'Apple' brand. Fig. 4 illustrates this. iii.*Finally,* we analyzed the distribution of the review length, which is the total count of characters in a particular review, and observed that over 175,000+ were less than 200 characters long. This is shown in Fig. 5. ***d. Feature Extraction***

After the analysis, the necessary features are extracted. As for this research, as we are working on a textual dataset, it can not be directly fed into the model. Therefore, they are first converted into numerical form and then are worked upon for further evaluation. This new format summarizes most of the information conveyed through textual data. And this is done using the Term Frequency-Inverse Document Frequency (TF-IDF) method. In this, the words are evaluated on the basis of their relevancy in the whole review [24]. Term Frequency is the frequency counter of the word in the entire corpus, and Inverse Document Frequency measures the informativeness of that word in the whole set of the corpus. Every individual word has its own set of TF and IDF scores, so multiplying these two scores results in a TF*IDF score for that word in the corpus [25]. This score helps in evaluating the rarity of a word, i.e., rarity is directly proportional to the value of this score. The higher the score, the higher is the rarity of that word. And with greater rareness, the word is more relevant and tends to appear in top search results. This helps in intercepting the usage of stop words easily [26].

e. Evaluating Metrics

Finally, after the feature has been extracted, the metrics are used to examine the performance of the models. We use a confusion matrix to describe the performance of each model. This matrix is a table with four different possible values for actual and predicted classes. The confusion matrix is illustrated in Table I.

True Positive (TP) depicts correctly predicted event values, False Positive (FP) depicts incorrectly predicted event values, True Negative (TN) depicts correctly predicted no-event values, and lastly, False Negative (FN) depicts incorrectly predicted no-event values [27, 28]. And this is used for measuring the following parameters:

i. *Precision:* This represents the proportion of predicted positives that are true positives. This metric measures the exactness of the review classified as a positive sentiment [28]. And it is represented by (1).

$$P = \frac{TP}{\mathrm{TP + FP}} \qquad (1)$$

ii. Recall: This is the proportion of real positives to the entire number of probable positive predictions that can be classified properly. It measures the susceptivity of the review classified as negative sentiment [28]. And it is represented by (2).

$$R = \frac{TP}{\mathrm{TP + FN}} \qquad (2)$$

iii. F1-Score: It represents the weighted harmonic mean of both precision and recall [27, 28]. And it is represented either by (3) or (4).

$$F = \frac{2 * P * R}{\mathrm{P + R}} \qquad (3)$$

$$2 * TP(4)\ F = \underline{\qquad\qquad\qquad}$$
$$2 * \mathrm{TP + FP + FN}$$

iv. *Accuracy:* It represents the percentage of true results in the total number of cases that are investigated [28]. And it is represented by (5).

$$\mathrm{TP + TN} \qquad (5)\ \mathrm{TP + TN + FP + FN}$$
$$A = \underline{\qquad\qquad\qquad\qquad}$$

5. Discussion And Results

The experimental findings from five different classifiers, including Multinomial Naïve Bayes, Support Vector Machine, Logistic Regression, Decision Tree, and Random Forest Classifier, are shown here. Tables II-VI depict the outcomes of the evaluating metrics for three types of labeled data, viz. 0 for a negative, 1 for neutral, and 2 for a positive. The overall result showed that the Random Forest Classifier worked better with the dataset and gave an overall accuracy of 92.33%.

TABLE I. CONFUSION MATRIX

		Actual Class	
		Positive	***Negative***
Predicted Class	***Positive***	True Positive	False Positive
	Negative	False Negative	True Negative

TABLE II. METRICS FOR MULTINOMIAL NAÏVE BAYES CLASSIFIER

Label	Precision	Recall	F1-Score	Accuracy
0	0.79	0.80	0.80	85.19 %
1	0.45	0.22	0.29	
2	0.89	0.94	0.92	

TABLE III. METRICS FOR SUPPORT VECTOR MACHINE CLASSIFIER

Label	Precision	Recall	F1-Score	Accuracy
0	0.81	0.86	0.83	87.55 %
1	0.69	0.19	0.30	
2	0.90	0.96	0.93	

TABLE IV. METRICS FOR LOGISTIC REGRESSION CLASSIFIER

Label	Precision	Recall	F1-Score	Accuracy
0	0.81	0.85	0.83	87.26 %
1	0.61	0.18	0.28	
2	0.90	0.96	0.93	

TABLE V. METRICS FOR DECISION TREE CLASSIFIER

Label	Precision	Recall	F1-Score	Accuracy
0	0.84	0.83	0.84	88.59 %
1	0.66	0.57	0.61	
2	0.93	0.94	0.93	

TABLE VI. METRICS FOR RANDOM FOREST CLASSIFIER

Label	Precision	Recall	F1-Score	Accuracy
0	0.92	0.85	0.90	92.33 %
1	0.97	0.52	0.68	
2	0.92	0.99	0.95	

TABLE VII. CLASSIFIERS AND THEIR CORRESPONDING ACCURACY

Classifiers	Accuracy
Multinomial Naïve Bayes	85.19 %
Support Vector Machine	87.55 %
Logistic Regression	87.26 %
Decision Tree	88.59 %
Random Forest	92.33 %

6. Comparison of the Classifiers

We have now compared the outcomes of the five classifiers that were obtained in the previous section. And this comparison is done by considering accuracy. Table VII shows the comparative analysis with accuracy. The results obtained from all the classifiers were analyzed. The Random Forest gives us the best outcomes and outperformed all the other classifiers with the highest accuracy of 92.33%, thus it works well with the given dataset of Amazon mobile reviews, while the Multinominal Naïve Bayes classifier showed the least accuracy of 85.19%. Hence, it is important to examine the reasons for the outperformance of the Random Forest over others in the dataset. As mentioned earlier, a Random Forest can be used as a classifier or a regressor that is majorly an ensemble of many Decision Trees. This factor alone provides it with the following advantages over Decision Trees and other bagging classifiers that have the same hyperparameters as the Random Forest when trained on a large variety of datasets:

i. *Performance:* The prediction score computed by the Random Forest is the prediction score of the majority of trees in the Random Forest for a given target variable, and the majority outweighs the prediction score of an individual tree. Additionally, the overall prediction error is also reduced when we take the average of prediction scores of multiple trees in the Random Forest giving the same numerical value for the target variable. ii. *Robustness:* The probability of overfitting a Random Forest is low as compared to an individual Decision Tree or other classifiers that we have used. This is attributed to randomness in the feature selection while splitting the node. Moreover, when we compare a single Decision Tree with the Random Forest, we generally have a high-variance estimator in hand as the prediction estimated by a single Decision Tree can be greatly impacted if we make a small change in the dataset used for training the model. The Random Forest provides us with a chance to make a low-variance estimator by making an ensemble of many Decision Trees where we will use the sampling technique with the replacement of samples for every tree in the Random Forest that is going to be utilised in aggregation for the overall prediction of the model. iii. *Scalability:* Another important advantage that comes with the Random Forest is its ability to automatically scale the importance of each feature by considering the impurity or error that comes into the prediction of the nodes of the trees. This can help the model to give more relative importance to a feature that introduces less impurity in the overall prediction.

The above inherent advantages of a Random Forest are not the only factors that make us biased towards using this model over others, but we can also improve its performance and training speed by tuning its hyperparameters to either increase its execution speed or the comprehensive prediction accuracy of the model.

7. Experimental Evaluation of Random Forest

The result of the comparative study shows that Random Forest performed optimally on the unbalanced dataset of Amazon mobile phone reviews. These results encouraged us to analyze the performance of the Random Forest using methods like out-of-bag and cross-validation to fine-tune the Random Forest's hyperparameters. The purpose of doing this analysis only on

the Random Forest is inclined towards the aim of finding an optimal classifier that may have higher chances of performing better as compared to other standard classifiers used to carry out sentiment analysis for product manufacturers. The Random Forest is an adjustment provided for the bagged decision trees to form a large number of de-correlated trees that can additionally enhance predictive execution with reasonably less need for hyperparameter tuning [29]. However, simple alteration of bagged trees can result in tree-correlation that in turn restricts the impact of variance reduction. This is conquered by mixing more randomness into the treedeveloping cycle [30]. As through the algorithm, a bootstrap sample is arbitrarily chosen for training and a random sample of features for each split, therefore a more different arrangement of trees is generated that will in general decrease the tree-correlation and raise the predictive power significantly. The variables and limits of the Random Forest are the parameters utilized to split each node throughout training, and Scikit-Learn [31] is not an assertion to provide an ideal solution as it applies a default set of reasonable hyperparameters for all the models. Thus, it is impractical to determine the best hyperparameters beforehand and a greater need arises to rely on an analytical approach.

We now have a trained Random Forest model which is optimal for the Amazon mobile reviews, but in pursuit of a greater degree of optimal performance, we analyzed the performance of this classifier. There are many ways to do this. For example, we can collect more data and then perform feature engineering, and this generally gives the best result in terms of the time contributed to the improved performance. But once all the data sources have been drained and are unknown to the variables of manipulation, then we can tune the hyperparameters of our model. We have tuned the parameters of the Random Forest in primarily two ways.

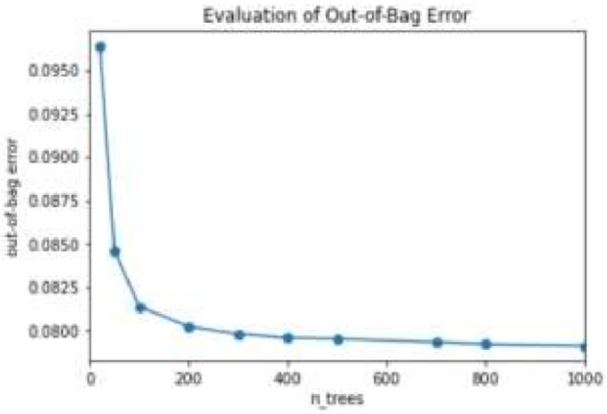


Fig. 6. Distribution of out-of-bag error and n_estimators (n_trees).

a. Tuning of Random Forest by Out-of-bag Error

Out-of-bag error is a technique utilized for estimating prediction errors for the lowest variance results. This error is essentially the average error for each training observation determined by utilizing the predictions from the trees that do not have these training observations in their corresponding bootstrap sample, thus allowing the random forest to be fit and validated while it is being trained [30]. When the samples are prepared, certain data points fail to be a part of a specific sample during training and form the out-of-bag points. In Fig. 6, we have shown a demonstration of how the out-of-bag error can be helpful for choosing a rough appropriate estimation of n_estimators, i.e., the number of trees (n_trees) at which the error balances out, and here it stabilizes near 1000.

b. Evaluation with Out-of-bag Error

The out-of-bag error estimate can be calculated from the out-of-bag score, i.e., the oob_score parameter of the Random Forest that is set to be true. It is represented by (6).

$$oob\_error \ \square\square 1 \ \square \ oob\_score \qquad (6)$$

There are various advantages to using oob_score for the analysis of the Random Forest, like no data leakage, better predictive models, and no overfitting of the model resulting in less variance, which comes at the expense of more time spent on validating the model using oob_score. The overall evaluation metrics revealed that the Random Forest performed slightly better when we selected n_estimators as 1000, and the experimental result is depicted in Table VIII.

TABLE VIII. METRICS FOR RANDOM FOREST CLASSIFIER WITH OOB_SCORE

Label	Precision	Recall	F1-Score	Accuracy
0	0.92	0.88	0.90	92.55 %
1	0.97	0.52	0.68	
2	0.92	0.99	0.96	

c. Cross-validation of Random Forest

Cross-validation is a method that is used in the estimation of metrics that measures the performance of any classifier by first training it on one subgroup of the data and then evaluating its performance based upon certain metrics on another subgroup of the data from the original dataset [32]. Generally, we estimate the training error of the model by evaluating it after training. But this limits us to know the performance of our classifier only on the data on which the training was performed. However, when we test our model on an unknown dataset, our classifier might be overfitted [33]. An overfitted model may give impressive results on the training dataset but can not be applied to new real-world applications. Therefore, the standardization for optimizing the hyperparameters that account for overfitting of the data through cross-validation techniques is as follows:

i. *Holdout Method:* It is one of the simplest cross-validation techniques in which the input data is divided into different sets of data [32]. The Random Forest is first trained, and then it undergoes testing subjected to a different set of performance metrics. The dataset can be split into any standard ratio, like 80:20. The metrics obtained in Table VI are computed through the holdout method, in which "reviews" of the Amazon mobile reviews dataset are mixed up anyhow before it undergoes splitting. Though we have trained the model on a different combination of samples, it can not guarantee that the training set that has been selected demonstrates the whole data.

ii. *Cross-validation with K-fold:* It is a method that is used to enhance the basic holdout method as when the data is limited, removing a part of it for validation may give rise to underfitting, so we can utilise this method in which the data is separated into distinct subgroups of a number k and the simple holdout based idea is rehashed k number of times [32], thus ensuring that the score of the Random Forest is not dependent upon the way we select our training and testing set, resulting in a less biased model compared to other methods. And generally, k is taken as 3, 5, or 10 while using this method.

d. Evaluation of Metrics with Cross-validation

A hyperparameter is a model parameter that is defined before training begins [33]. Various models have numerous diverse hyperparameters that can be set, and we have validated our model using 3-fold cross-validation on the following four parameters for the analysis of the Random Forest:

i. *n_estimators:* This parameter represents the total number of trees in the Random Forest which will be generated when it grows during the training and testing of the data. The standard value of this parameter is 10. We can get better performance if we use higher values of this parameter, but the training time of the model is compromised. The validation curve was created for the n_estimators parameter for a range of values from 20 to 300 and it is depicted in Fig. 7 and 8. The distributions indicate that despite the substantial divergence in training and cross-validation scores, for each of the three cross-validations, the mean training accuracy was more than 98%, while the mean cross-validation accuracy was between 88% and 90% for all n_estimator values. This demonstrates that despite the large number of n_esitmators employed, the Random Forest is moderately accurate.

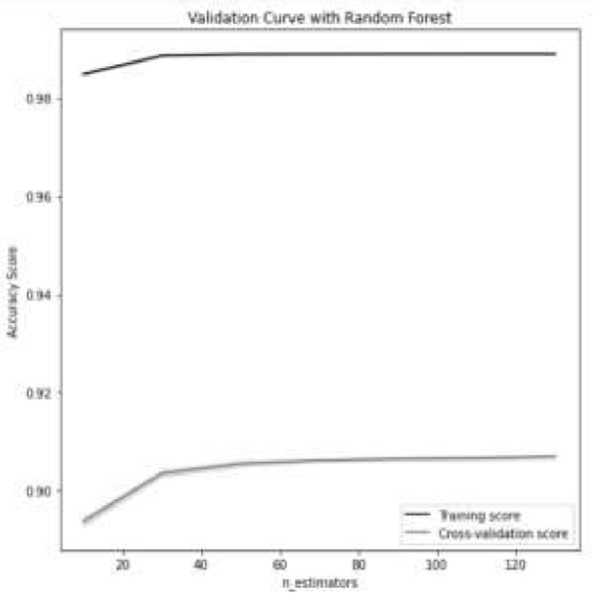


Fig. 7. Distribution of score and n_estimators.

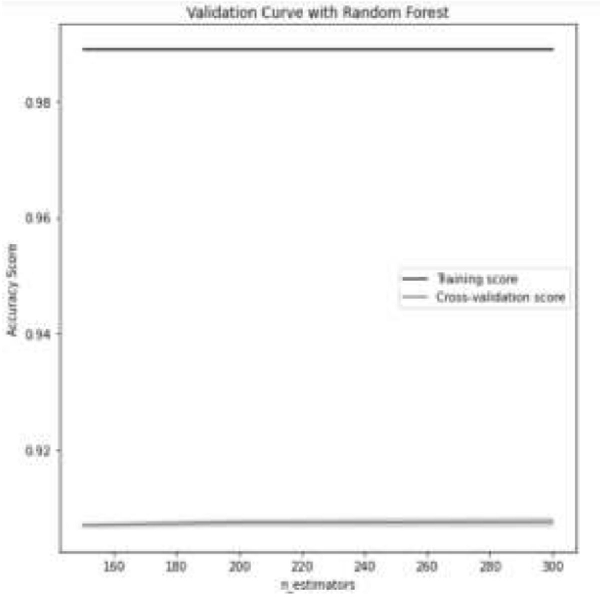


Fig. 8. Distribution of score and n_estimators.

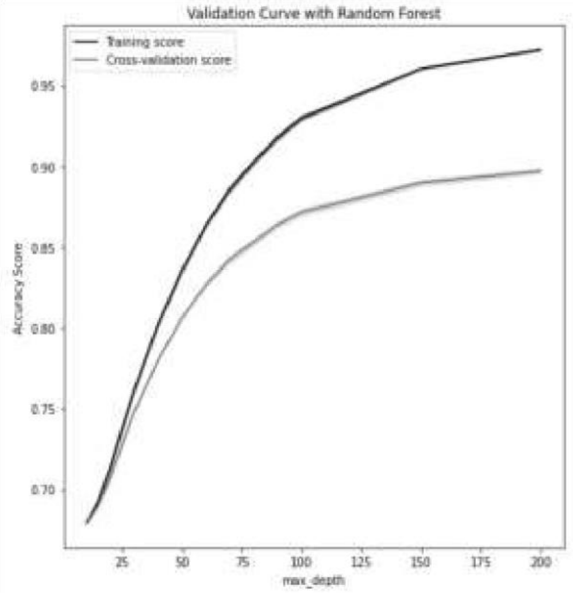


Fig. 9. Distribution of score and max_depth.

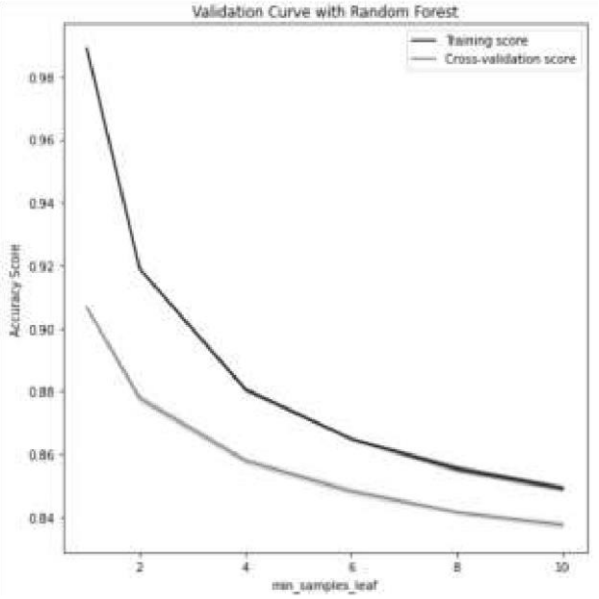


Fig. 10. Distribution of score and min_samples_leaf.

ii. *max_depth:* It determines the greatest profundity of each tree, and its standard value is none. It essentially controls the maximum depth of each tree in a Random Forest. The Random Forest was 3-fold cross-validated on values ranging from 10 to 200 for max_depth. Fig. 9 shows that when max_depth is 75, then the cross-validation score is above 80%, whilst the training score is above 85%. Although we may select a larger value that gives us the maximum accuracy while training the Random Forest, this may however lead to overfitting of the training data and might result in a model that is not suitable for certain purposes. iii. *min_samples_leaf:* It determines how many samples are required at each leaf node, and its standard value is one. The cross-validation curves in Fig. 10 suggest that the standard value of one is the best choice.

iv. *min_samples_split:* It signifies the lowest possible number of samples that are essentially needed to separate an internal terminal node, and its standard value is two. The cross-validation curve in Fig. 11 shows that the standard value of two is the most appropriate value for this parameter. We will have more generic terminal nodes if we choose bigger values for the minimal number of samples that are necessary before an internal node splits, which will affect the overall accuracy.

The suggested values of all the above parameters when tuned into the Random Forest show us that its performance based on the evaluating metrics in Table IX decreased even though we have cross-validated the model at the expense of a large amount of validating time spent for the purpose. Although we have obtained a deeper insight into tuning the hyperparameters, they can be used to carry out more exhausting hyperparameter tuning methods like GridSearchCV. It would be more useful to do feature engineering and use the default parameters of the Random Forest for better performance.

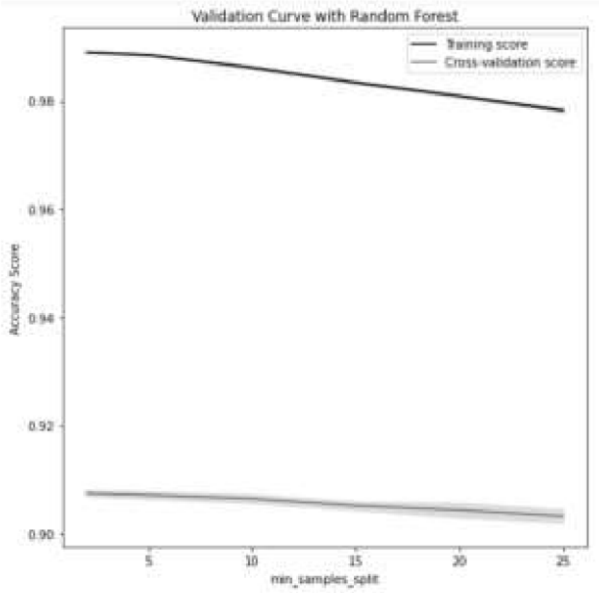


Fig. 11. Distribution of score and min_samples_split.

TABLE IX. METRICS FOR RANDOM FOREST CLASSIFIER WITH THE SUGGESTED VALUES OF HYPERPARAMETERS BY 3-FOLD CROSS-VALIDATION

Label	Precision	Recall	F1-Score	Accuracy
0	0.93	0.68	0.78	86.35 %
1	1.00	0.32	0.48	
2	0.84	0.91	0.91	

8. Conclusion and Future Scope

Reviews are a vital part of an e-commerce platform and are responsible for determining the overall product. With the escalating advancement in technology, the need to understand the sentiments expressed through a review has become highly essential. For our research, we used the TF-IDF feature extraction technique and utilized five classifiers, viz. Multinomial Naïve Bayes, Support Vector Machine, Logistic Regression, Decision Trees, and Random Forest to carry out the sentimental analysis and classification of Amazon phone reviews and then later did their comparison. And for evaluating the results, we used a confusion matrix and applied four evaluating metrics, namely precision, recall, f1-score, and accuracy. The comparative analysis showed that the Random Forest gave the optimal outcomes with an accuracy of 92.33%. Though the Multinomial Naïve Bayes classifier showed the least accuracy, it can still be used with a smaller dataset as it had an accuracy of 85.19%, and it will work well with a lower number of reviews. Decision Trees were also found to be effective, and they can be utilized in certain cases as they had an accuracy of over 88%. In pursuit of getting a classifier that has a greater degree of optimal performance, the Random Forest was further assessed as it gave the best result with the methodology that was followed. And this was done by tuning parameters of our model with methods like out-of-bag error and 3-fold cross-validation. With the former technique, the accuracy was enhanced to 92.55%, but it was reduced to 86.35% with the latter technique. Therefore, a Random Forest tuned with the help of out-of-bag error can be used as a primary method to perform sentimental analysis and classification of the reviews to reach a decision. For the immediate future, we intend to use additional classifiers and a lexicon-dependent technique with different feature extraction techniques, viz. bag-of-words and word2vec, to get a greater outlook and thereby produce an enhanced model with better accuracy. Apart from the abovementioned intended elements, such as the unsupervised learning approach for future work, we also aim to enhance the performance of the best-selected model from our comparative analysis approach that has been carried out so far. Also, we may work on reviews with emoji and a larger dataset at the same time, to generate an efficient version of the current model. And we additionally aim to work with reviews in different languages to have a bigger scope of reaching out to a wider audience.

References

[1]. Theresa Wilson, Janyce Wiebe, and Paul Hoffmann, “Recognizing contextual polarity in phrase-level sentiment analysis”, Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pp. 347–354, 2005. Available at: https://dl.acm.org/doi/10.3115/1220575.1220619.

[2]. Zhu Zhang, “Weighing stars: Aggregating online product reviews for intelligent ecommerce applications”, IEEE Intelligent Systems, vol. 23, no. 5, pp. 42–49, 2008. Available at: 10.1109/MIS.2008.95.

[3]. Callen Rain, “Sentiment Analysis in Amazon Reviews Using Probabilistic Machine Learning”, Swarthmore College, Department of Computer Science, 2013. Available at: https://www.sccs.swarthmore.edu/users/15/crain1/files/NLP_Final_Project.pdf.

[4]. Kranti Ghag and Ketan Shah, “Comparative analysis of the techniques for Sentiment Analysis”, International Conference on Advances in Technology and Engineering, no. 124, pp. 1–7, 2013. Available at: 10.1109/ICAdTE.2013.6524752.

[5]. Xing Fang and Justin Zhan, “Sentiment analysis using product review data”, Journal of Big Data, vol. 2, no. 1, pp. 5, 2015. Available at: https://doi.org/10.1186/s40537-015-0015-2.

[6]. Muhammad T. Khan, M. Durrani, Armughan Ali, Irum Inayat, Shehzad Khalid and Kamran H. Khan, “Sentiment analysis and the complex natural language”, Complex Adapt Syst Model, pp. 1–19, 2016. Available at: https://doi.org/10.1186/s40294-016-0016-9.

[7]. Mohan Kamal Hassan, Sana Prasanth Shakthi, Sasikala Ra, “Sentiment analysis of Amazon reviews using naïve bayes on laptop products with MongoDB and R”, IOP Conference Series Materials Science and Engineering, vol. 263, pp. 1–10, 2017. Available at: 10.1088/1757-899X/263/4/042090.

[8]. Heidi Nguyen, Aravind Veluchamy, Mamadou Diop, Rashed Iqbal, “Comparative Study of Sentiment Analysis with Product Reviews Using Machine Learning and Lexicon-Based Approaches,” SMU Data Science Review, vol. 1, no. 4, Article 7, 2018. Available at: https://scholar.smu.edu/datasciencereview/vol1/iss4/7/.

[9]. Abhilasha Tyagi, Naresh Sharma, “Sentiment Analysis using Logistic Regression and Effective Word Score Heuristic,” International Journal of Engineering and Technology(UAE), vol. 7, no. 2.24, pp. 20–23, 2018. Available at: 10.14419/ijet.v7i2.24.11991.

[10]. Wanliang Tan, Xinyu Wang, and Xinyu Xu, “Sentiment Analysis for Amazon Reviews,” International Conference on Human and AI interaction, 2018. Available at: http://cs229.stanford.edu/proj2018/report/122.pdf.

[11]. Momina Shaheen, Shahid M. Awan, Nisar Hussain, Zaheer A. Gondal, "Sentiment Analysis on Mobile Phone Reviews Using Supervised Learning Techniques,” International Journal of Modern Education and Computer Science(IJMECS), vol. 11, no. 7, pp. 32–43, 2019. Available at: http://www.mecs-press.org/ijmecs/ijmecs-v11-n7/IJMECS-V11-N7-4.pdf.

[12]. Sara A. Aljuhani, Norah S. Alghamdi, “A Comparison of Sentiment Analysis Methods on Amazon Reviews of Mobile Phones,” International Journal of Advanced Computer Science and Applications, vol. 10, no. 6, pp. 608–617, 2019. Available at: 10.14569/IJACSA.2019.0100678.

[13]. Jayakumar Sadhasivam, Ramesh B. Kalivaradhan, “Sentiment Analysis of Amazon Products Using Ensemble Machine Learning Algorithm,” International Journal of Mathematical, Engineering and Management Sciences, vol. 4, no. 2, pp. 508–520, 2019. Available at: 10.33889/IJMEMS.2019.4.2-041.

[14]. Hui Zhang, “Sentiment Analysis on Amazon reviews,” pp. 1–13, 2019. Available at: 10.13140/RG.2.2.31090.53447.

[15]. Emilie Coyne, Jim Smit, Levent Güner, “Sentiment analysis for Amazon.com reviews,” pp. 1–9, 2019. Available at: 10.13140/RG.2.2.13939.37920.

[16]. Vineet Jain and Mayur Kambli, ”Amazon Product Reviews: Sentiment Analysis”, 2020. Available at: https://www.researchgate.net/publication/344677952_Amazon_Product_Reviews_Sentiment_Analysis.

[17]. K. Ashok Kumar, C. Jagadeesh, Pravin Kshirsagar, Swagat. M. Marve, “Sentiment Analysis of Amazon Product Reviews using Machine Learning,” Test Engineering and Management, vol. 82, pp. 5245–5254, 2020. Available at: http://www.testmagzine.biz/index.php/testmagzine/article/view/1670/1505.

[18]. Shuo Xu, Yan Li and Wang Zheng, “Bayesian Multinomial Naïve Bayes Classifier to Text Classification,” International Conference on Multimedia and Ubiquitous Engineering International Conference on Future Information Technology, pp. 347–352, 2017. Available at: 10.1007/978-981-10-5041-1_57.

[19]. Konstantinas Korovkinas, Gintautas Garšva, “Selection of Intelligent Algorithms for Sentiment Classification Method Creation,” International Conference on Information Technologies, pp. 152–157, 2018. Available at: http://ceur-ws.org/Vol-2145/p26.pdf.

[20]. Richard A. Berk, “Support Vector Machines, In: Statistical Learning from a Regression Perspective,” Springer Texts in Statistics, Springer, Cham, 2016. Available at: https://doi.org/10.1007/978-3-319-44048-4_7.

[21]. Chao-Ying Joanne Peng, Kuk Lida Lee, and Gary M. Ingersoll, “An introduction to logistic regression analysis and reporting,” The Journal of Educational Research, vol. 96, no. 1, pp. 3–14, 2002. Available at: 10.1080/00220670209598786.

[22]. Edgar C. Merkle and Victoria A. Shaffer, “Binary recursive partitioning: Background, methods, and application to psychology,” The British journal of mathematical and statistical psychology, vol. 64, pp. 161–81, 2011. Available at: 10.1348/000711010X503129.

[23]. Jehad Ali, Rehanullah Khan, Nasir Ahmad, Imran Maqsood, “Random Forests and Decision Trees,” International Journal of Computer Science Issues, vol. 9, pp. 272–278, 2012. Available at: https://ijcsi.org/papers/IJCSI-9-5-3-272-278.pdf.

[24]. Shahzad Qaiser and Ramsha Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," International Journal of Computer Applications, vol. 181, no. 1, pp. 25–29, 2018. Available at: 10.5120/ijca2018917395.

[25]. Q. Liu, J. Wang, D. Zhang, Y. Yang and N. Wang, "Text Features Extraction based on TF-IDF Associating Semantic," IEEE 4th International Conference on Computer and Communications, pp. 2338–2343, 2018. Available at: 10.1109/CompComm.2018.8780663.

[26]. Ravinder Ahuja, Aakarsha Chuga, Shruti Kohlia, Shaurya Gupta and Pratyush Ahuja, "The Impact of Features Extraction on the Sentiment Analysis," International Conference on Pervasive Computing Advances and Applications, vol. 152, pp. 341–348, 2019. Available at: 10.1016/j.procs.2019.05.008.

[27]. Cyril Goutte and Eric Gaussier, "A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation," Lecture Notes in Computer Science, vol. 3408, pp. 345–359, 2005. Available at: https://doi.org/10.1007/978-3-540-31865-1_25.

[28]. Mohammad Hossin and Sulaiman M.N, "A review on evaluation metrics for data classification evaluations," International Journal of Data Mining and Knowledge Management Process, vol. 5, no. 2, pp. 1–11, 2015. Available at: 10.5121/ijdkp.2015.5201.

[29]. Leo Breiman, "Random Forests," Machine Learning, Springer, vol. 45, no. 1, pp. 5–32, 2001. Available at: 10.1023/A:1010933404324.

[30]. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, "Random Forests, In: The Elements of Statistical Learning," Springer Series in Statistics New York, NY, USA, vol. 1, pp. 587–604, 2009. Available at: https://doi.org/10.1007/978-0-387-84858-7_15.

[31]. Fabian Pedregosa et al., "Scikit-learn: Machine Learning in Python," JMLR, vol. 12, no. 85, pp. 2825–2830, 2011. Available at: https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf.

[32]. Daniel Berrar, "Cross-Validation in Encyclopedia of Bioinformatics and Computational Biology," Elsevier, pp. 542–545, 2018. Available at: 10.1016/B978-0-12-809633-8.20349-X.

[33]. Philipp Probst, Marvin Wright and Anne-Laure Boulesteix, "Hyperparameters and Tuning Strategies for Random Forest", WIREs Data Mining and Knowledge Discovery, vol. 9, no. 3, 2019. Available at: https://doi.org/10.1002/widm.1301.

A Secure Image Steganography using X86 Assembly LSB

Avi Gupta[1], Himanshu Shukla[2], Meenu Gupta[3]
[1,2]Student, Department of Computer Science and Engineering, Chandigarh University Punjab, India - 140413
[3]Associate Professor, Department of Computer Science and Engineering, Chandigarh University Punjab, India - 140413
17bcs1603@cuchd.in, 19bcs1641@cuchd.in, meenu.e9406@cumail.in

Abstract: Steganography has become a very important research field in recent years including many programs. It is scientific to embed details in a cover photo i.e., Text, video, and image (paid load) without creating too many changes to the cover image. Today's a secure image steganography is used to represents the daunting task of transferring embedded data to an invisible realm. This work discusses about a picture of steganography that includes Discrete Cosine Transform (DCT), Least Significant Bit (LSB), and compression techniques for green images to improve the security of the paid uploads. At initial stage the LSB was used to embed pre-loaded pieces on the cover image to get a stegno image. A DCT technique is used to convert stegno image from a local domain to a standard domain. Further, this work discusses about the transmission of secure images with MSE and low BER without use of any password and compared to previous functions.

Keywords: Cybersecurity. Assemblyx86 language. Data privacy. Reverse Engineering. Radare2

Introduction

In present era, where every day new technology came with different features which is a collection of new data. When technology changes then it come up with major challenges where security is the major concern (i.e., cause of loss of data) [2, 9]. The information needs to be kept secure and secure to be accessed only by specific person and any other user can not access that data. Sharing of data is also increasing due to huge amount of information is transferred in the form of thousands of messages and data are sent online per day. Data protection is a top priority for the sender [1]. The data protection requires to send data privately (i.e., message) where sender and receiver can only understand it by sharing a secret code [5]. The cryptographic key is known only to authorized people who can decrypt the received message. There is a limitation of encrypting the text Messages which shows a notification of hidden text. That can be a cause of knowing about the secret message is send by someone [17].

Overcoming this limit, the steganography process was introduced [6]. As compared to cryptography, the method of steganography is much better because in this the data was hidden by the image [9] and then the images are uploaded online [8]. This methodology also helps to the average person who does not know about the presence/ absence of data in the image. A data from an image can be removed by an authenticate person who can access the key to determine the details [11, 22]. Due to this reason a security and reliability of transfer data also improved with the because no one else can change the data sent. The most widely used fields for steganography is shown in figure 1.

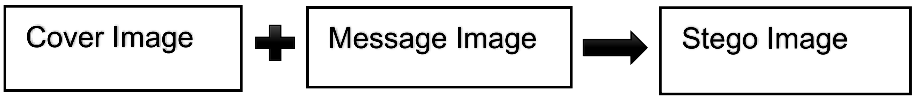


Fig 1. A Typical Steganography Technique [4]

Literature Review

The steganography is a growing field in the term of security and privacy of image data. Steganalysis is a breakdown of steganography and it is the way of finding hidden information [14]. Mainly the Steganalysis is used to break the steganography and the acquisition of stegno images. All Steganalysis algorithms are based on steganographic algorithms that show mathematical differences between the cover and the stegno image as shown in figure 2. In [3], the authors discussed about the examination of medical records of a patient's sensitive information reveals much about the power of the imaging-based treatment system. This method provides a safe and secure way to protect digital medical imaging [16]. The authors used Integer Wavelet Transform (i.e., a steganography technique) to secure the MRI image on a single vessel image. he patient's diagnostic image was taken as a secret image and Arnold's modification was used to obtain an annoying secret image. A secret image was added to the container image and with the help of Inverse IWT an image is captures. It has been observed that quality standards are improved with an acceptable PSNR compared to existing algorithms [10].

Stego Image
Cover Image
Message Image

Fig 2. A Typical Steganography Technique

Spatial Domain Strategies

In local domain techniques, network object pixels, such as photo and video objects, are used directly and modified to hide private data within them [1] [15, 17]. The following strategies fall under the local domain as shown in figure 3:

- Non-Signal Bit (LSB): It is a simple steganography strategy. As with all steganographic methods, it encloses information on a cover, so that it cannot be observed by a casual observer [9]. This method works by inserting additional information into a given pixel with information from the data in the image [4].

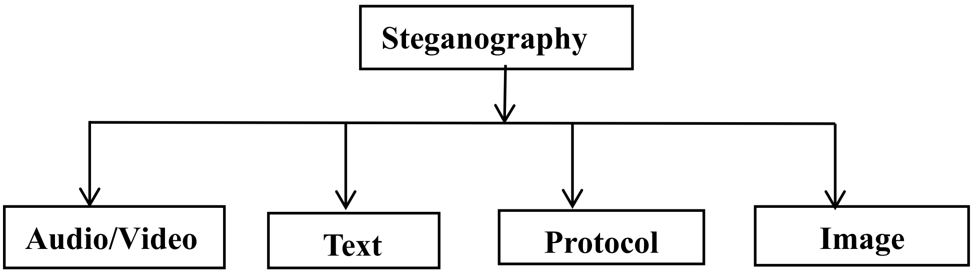


Fig 3. Categories of steganography

- Gray Conversion: In this to represent a binary data a pixel-level values of an image are converted according to the mathematical function. Every pixel has a different gray level value [3].
- Pixel Value Differencing (PVD): In this, it uses the difference of two consecutive pixels in a block to find out the number of secret bits embedded in an image [12]. After this, it

constructs a quantization table to determine the consecutive pixels values with its difference. In addition to this it offers the possibility to transfer a large number of uploads, while maintaining the similarity of the image element after the data embedding [2, 6].

Transform Domain Techniques

Changing domain strategies, a network company item first converts from a local domain to a domain, and then uses its own waves to hide private information. These methods are low-cost but robust in the fight against statistical attacks [8, 9]

- **Discrete Wavelet Transform (DWT):** DWT conversion states that achieving a wavelet transformation that uses a translation that follows defined rules and a separate set of wavelet scales [11].

- **Discrete Fourier Transform (DFT):** This modification is considered the most important modification used to carry out Fourier analysis in many operating systems. Samples can be the number of pixels in sequence or in the raster image column in the image processing [5].

- **Discrete Cosine Transform (DCT):** This modification introduces a consistent sequence of data points in the sense of the amount of flexible cosine activity on multiple frequencies. DCTs are important in various applications in engineering and science such as [13], lost audio files such as MP3 files, and images such as JPEG files wherever very small objects are banned. In fact, the use of cosine instead of sine functions is important in stress, because a small amount of cosine activity is needed to measure the normal signal. The figure 4 shows the order of pixels.

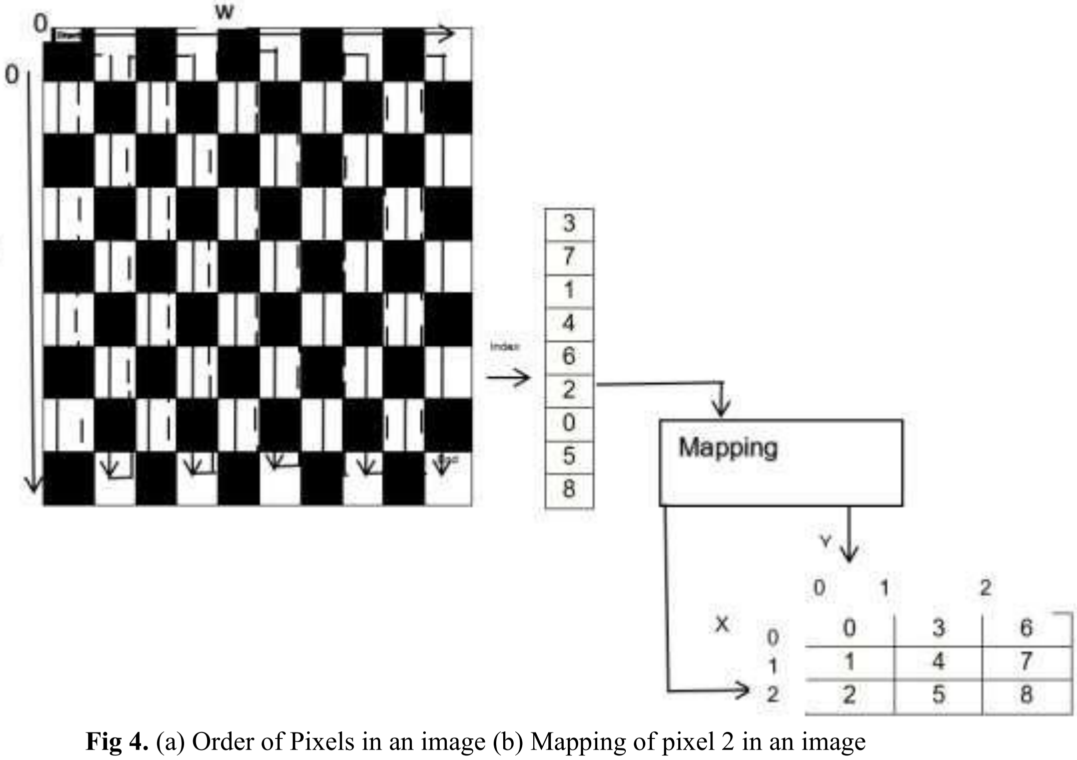


Fig 4. (a) Order of Pixels in an image (b) Mapping of pixel 2 in an image

Materials and methods

Learning objective in this work is focus on data security issues when a data sends to the network using steganographic techniques. The key elements of the work are shown in figure 5.

- The need of steganography program because a high message carried by stegnomedia is not making a sense to people.
- To avoid the suspicion in the existence of a hidden message, a steganography is used.

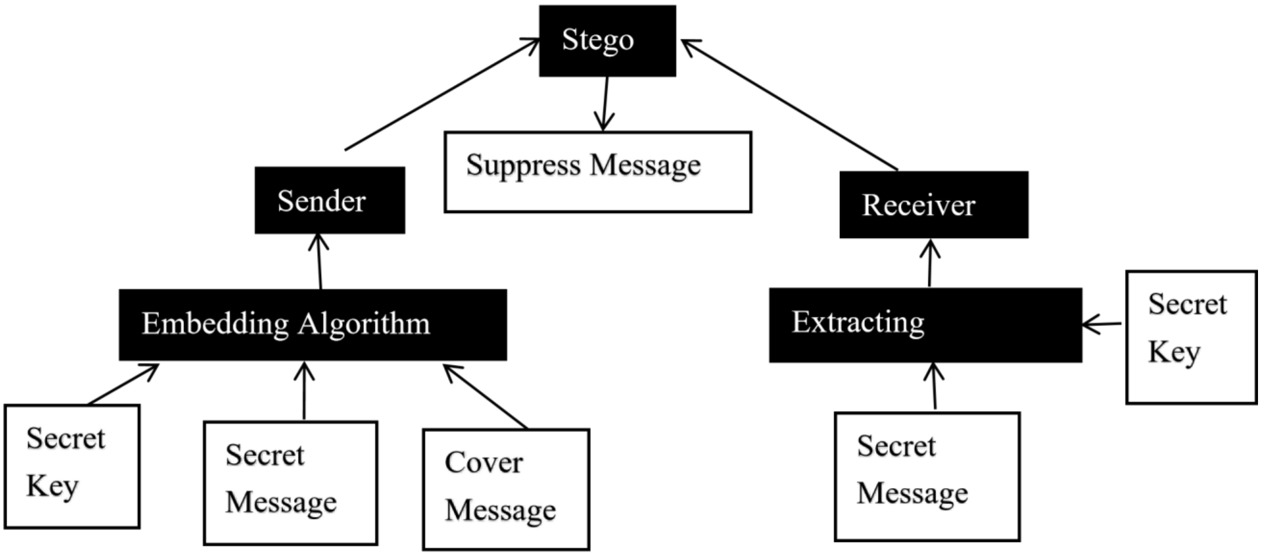


Fig 5. Steganography process used

Methodology used

In general, when a data is inserted into an image then an image may lose its resolution. This proposed work is securing the resolution of image as well as size while inserting the data in it. A speed of entering the data into an image is also higher as the image is protected and a data sent to the destination is safe [7, 15].

An AES (Advanced Encryption standard) is used to encrypt the messages that makes it difficult for unauthorized persons to extract the original message. There is one drawback of this method as it uses DWT and LSB which directly have an impact over its performance and very easy to get the original message. In [13], the authors proposed an algorithm in a digital image based on Least Significant Bit. They introduced a new steganographic approach to the local domain to add more details to the cover image using small changes to cover the pixels of the image. Their approach focuses on the LSB embedding process. They used LSB-2 to increase the secrecy of encryption. It provides additional security for bits of private messages because Stegno-Key is used to rearrange and allow pieces of the private message before removing them from the cover [16]. A figure 6 shows the methodology used in steganography.

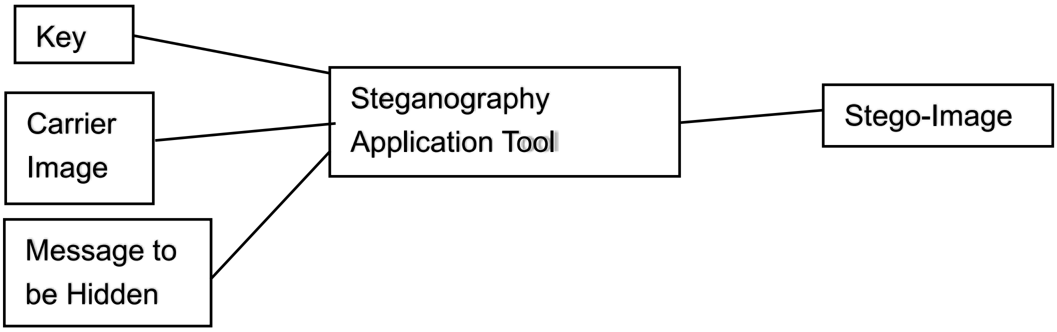


Fig. 6 Steganography methodology

Proposed Algorithm

Inserting the minimum key bits (LSB) is an easy way to embed details in an image file. Simple steganographic techniques incorporate the message pieces directly into the most important cover of the cover image in the sequence. The most important small change does not lead to an understanding person because the magnitude of the change is small [5, 6]. The figure 7 shows the process of LSP substitution in color images.

1: Read a short message set.
2: Read the pixels of the cover image.
3: Read the cover image of LSB-1.
4: Read LSB-2 cover photo set.

PSNR is defined as

$$\mathbf{R = 20log_{10}}\left(\frac{\mathbf{MAX_1}}{\sqrt{\mathbf{MSE}}}\right) \qquad (1)$$

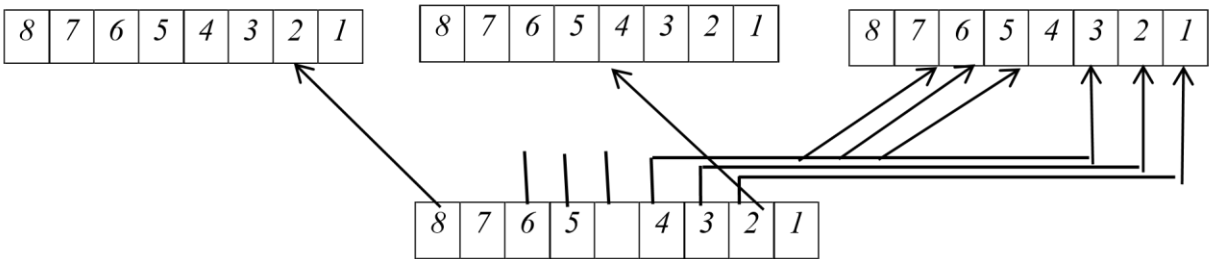


Fig7. The process of LSB substitution in a color image

Security is an important issue when communicating information over the Internet because any unauthorized person may hack the information and make it useless or obtain information that is not intended for them [23]- [26]. This problem often leads to challenges in invalidating, evaluating, and reproducing reported strategies in a consistent manner. It is our view that the research community of steganography / Steganalysis will benefit from the availability of the same database, thereby improving transparency and academic integrity. In this study, we considered four factors: image detection, pre-processing, steganographic techniques, and the degree of embedding in the creation of a steganography image database.

Experimental analysis

In this study, an analysis was performed considering the strength of the image by the process of LSB steganography to hide the image within another image. Three sensible functions are considered as low coverage methods and image quality is analyzed. In this work, a method is proposed using the pseudo rate LSB steganography. In this paper, we introduce two LSB algorithms based on quantum images, with at least two advantages: (1) blindness at all. The extraction process does not require an original cover or initial message. (2) The whole process can be accomplished by quantum computers and does not require the help of old computers or by people. Tests and simulation-based test results show that inconsistency is good, and the balance between volume and intensity can be adjusted according to the requirements of the applications. Figure 8 shows the conversion of MSB to LSB and its results shown in table 1.

Table 1: Results of sequential-LSB

Payload Size	Embedded Data (Bytes)	MSE	PSNR(dB)	NK	AD	SC	MD	LMSE(*10-6)	NAE
1	1024	0.031	63.2094	1	-	1	7	0.588	0.000
		1			0.006				1
2	2048	-0.062	60.206	1	-0.0014	1	7	0.6453	0.0001
4	4096	0.1259	57.1313	1	-0.0029	1	7	0.7134	0.0003
8	8192	0.2534	54.092	1	-0.0021	1	7	0.6633	0.0005
16	16384	0.501	51.1321	1.0001	-0.0104	0.9999	7	0.6166	0.0011
32	32768	1.0004	48.129	1.0001	-0.022	0.9998	7	1.1311	0.0021
64	65536	2.0011	45.1182	1.0002	-0.0426	0.9995	7	0.5073	0.0043
128	131072	3.996	42.1145	1.0005	-0.0846	0.9988	7	1.1311	0.0086
256	-	-	-	-	-	-	-	-	-

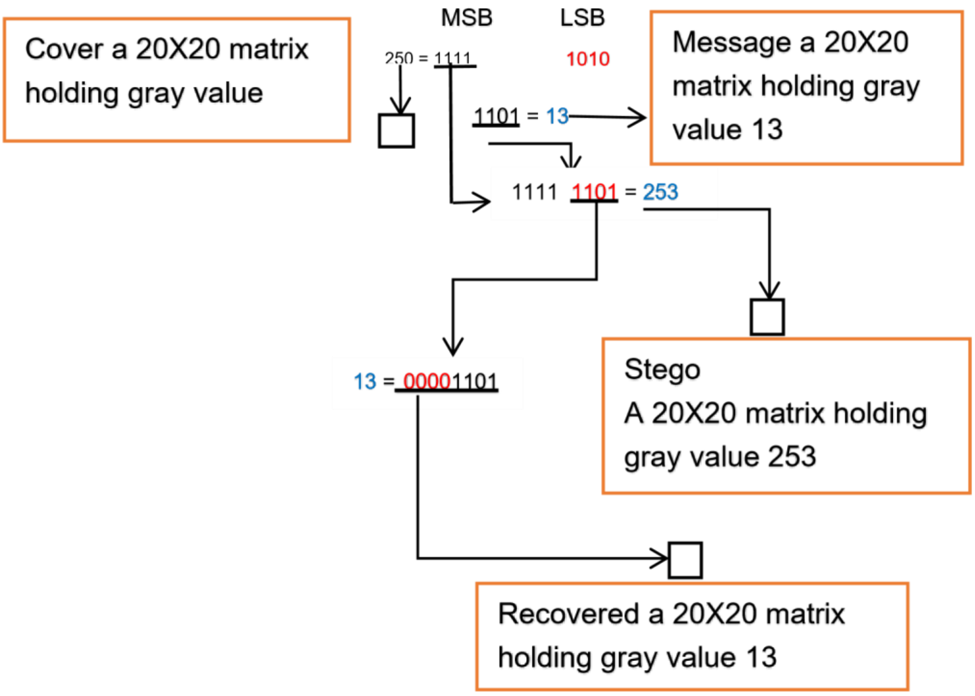

MSB
LSB
Cover a 20X20 matrix holding gray value
250 = 1111
1010
Message a 20X20 matrix holding gray value 13
1101 = 13
1111 1101 = 253
Stego A 20X20 matrix holding gray value 253
13 = 00001101
Recovered a 20X20 matrix holding gray value 13

Fig 8. MSB to LSB Conversion

Conclusion and future work

The image quality is always matter for any work. The main aim of this study is to improve the image quality and do compression of text. In future, this work can be implemented by using different image formats such as .tif, bmp, peg, etc. Securing the most important bit is providing a good security but by exchanging carriers using different encryption keys a quality of an image can be improved. The work shows that a steganography is used to hide the message written over an image. One case is considered in this work where one image is hidden over another image that is used to hide the data. A retrieved image after that embedded that is called as stegno image. Different methods such as DFT, LSB, etc. are used in steganography for suring the data but every method has its advantage and disadvantages. In this work, advanced LSB methodology is used that process the color images by embedding data into three RGB image planets to enhance the image quality and gains high embedding capabilities. The PSNR value of the proposed procedure is better than previous steganography methods.

References

[1] Trivedi, M. C., Sharma, S., & Yadav, V. K. (2016, March). Analysis of several image steganography techniques in spatial domain: A survey. In *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies* (pp. 1-7).

[2] Petitcolas, F. A., Anderson, R. J., & Kuhn, M. G. (1999). Information hiding-a survey. *Proceedings of the IEEE*, *87*(7), 1062-1078.

[3] Mazurczyk, W., & Caviglione, L. (2014). Steganography in modern smartphones and mitigation techniques. *IEEE Communications Surveys & Tutorials*, *17*(1), 334-357.

[4] Singla, D., & Juneja, M. (2014, March). An analysis of edge based image steganography techniques in spatial domain. In *2014 Recent Advances in Engineering and Computational Sciences (RAECS)* (pp. 1-5). IEEE.

[5] Akhtar, N. (2016). An LSB substitution with bit inversion steganography method. In *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics* (pp. 515-521). Springer, New Delhi.

[6] Chen, Y. Z., Han, Z., Li, S. P., Lu, C. H., & Yao, X. H. (2010, October). An adaptive steganography algorithm based on block sensitivity vectors using HVS features. In *2010 3rd International Congress on Image and Signal Processing* (Vol. 3, pp. 1151-1155). IEEE.

[7] Chan, C. K., & Cheng, L. M. (2004). Hiding data in images by simple LSB substitution. *Pattern recognition*, *37*(3), 469-474.

[8] Wu, D. C., & Tsai, W. H. (2003). A steganographic method for images by pixelvalue differencing. *Pattern recognition letters*, *24*(9-10), 1613-1626.

[9] Wu, H. C., Wu, N. I., Tsai, C. S., & Hwang, M. S. (2005). Image steganographic scheme based on pixel-value differencing and LSB replacement methods. *IEE Proceedings-Vision, Image and Signal Processing*, *152*(5), 611-615.

[10] Anand, J. V., & Dharaneetharan, G. D. (2011, February). New approach in steganography by integrating different LSB algorithms and applying randomization concept to enhance security. In *Proceedings of the 2011 International Conference on Communication, Computing & Security* (pp. 474476).

[11] Kukapalli, V. R., Rao, T., & Reddy, S. (2014). Image Steganography byEnhanced Pixel Indicator Method Using Most Significant Bit (MSB) Compare. *International Journal of puter Trends and Technology (IJCTT)–15*, *3*, 97-101.

[12] Dighe, D., & Kapale, N. D. (2013). Random Insertion Using Data Parity Steganography Technique. *Int. J. Eng. Sci. Innov Technol (IJESIT)*, *2*(2), 364-368.

[13] Bashardoost, M., Sulong, G. B., & Gerami, P. (2013). Enhanced LSB image Steganography method by using knight Tour algorithm, Vigenere Encryption and LZW compression. *International Journal of Computer Science Issues (IJCSI)*, *10*(2 Part 1), 221.

[14] Dadgostar, H., & Afsari, F. (2016). Image steganography based on intervalvalued intuitionistic fuzzy edge detection and modified LSB. *Journal of information security and applications*, *30*, 94-104.

[15] Fridrich, J., Goljan, M., & Du, R. (2001). Detecting LSB steganography in color, and gray-scale images. *IEEE multimedia*, *8*(4), 22-28.

[16] Peterson, W. W., & Brown, D. T. (1961). Cyclic codes for error detection. *Proceedings of the IRE*, *49*(1), 228-235.

[17] Li, B., Shen, H., & Tse, D. (2012). An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check. *IEEE communications letters*, *16*(12), 2044-2047.

[18] Morkel, T., Eloff, J. H., & Olivier, M. S. (2005, June). An overview of image steganography. In *ISSA* (Vol. 1, No. 2, pp. 1-11).

[19] Libre and Portable Reverse Engineering Framework. Available at - https://rada.re/

[20] Jorgensen, E. (2019). x86-64 Assembly Language Programming with Ubuntu.

[21] Hyde, R. (2010). The art of assembly language. No Starch Press.

[22] Singh, A. K., Singh, J., & Singh, H. V. (2015). Steganography in images using lsb technique. International Journal of Latest Trends in Engineering and Technology (IJLTET), 5(1), 426-430.

[23] Zhang, Q., Li, Y., Al-Turjman, F. *et al.* Transient ischemic attack analysis through non-contact approaches. *Hum. Cent. Comput. Inf. Sci.* **10,** 16 (2020). https://doi.org/10.1186/s13673-020-00223-z

[24] Bhardwaj, A., Al-Turjman, F., Sapra, V., Kumar, M., & Stephan, T. (2021). Privacy-aware detection framework to mitigate new-age phishing attacks. *Computers & Electrical Engineering*, *96*, 107546.

[25] Al-Turjman, F., & Bakkiamdavid, D. (2021). A Proxy-Authorized Public Auditing Scheme for Cyber-Medical Systems Using AI-IoT. *IEEE Transactions on Industrial Informatics*.

[26] Nagasubramanian, G., kumar Sakthivel, R., Al-Turjman, F., & Senior Member, I. E. E. E. (2021). Secure and Consistent Job Administration Using Encrypted Data Access Policies in Cloud Systems. *Computers & Electrical Engineering*, *96*, 107520.