### A PROJECT REPORT ON

**“Fake News Detection with Machine Learning”**

Submitted for partial fulfillment of the award of the degree

**BACHELOR OF TECHNOLOGY**

**(Information Technology)**

**BY**

Rishabh Jha (MITU17BTIT0015)

Prashansa Bansal (MITU17BTIT0005)

Sahil Gavthankar (MITU17BTIT0011)

Vivek Rote (MITU17BTIT0044)

**Under the guidance of**

Prof. Rohini T. Bhosale



**Department of Information Technology**

### MIT School of Engineering

### MIT Art, Design and Technology University, Pune

### MAEER’s Rajbaug Campus, Loni-Kalbhor, Pune 412201

**2017 - 2021**



**MIT SCHOOL OF ENGINEERING**

DEPARTMENT OF INFORMATION TECHNOLOGY

MAEER’s Rajbaug Campus, Loni-Kalbhor, Pune - 412201

**CERTIFICATE**

This is to certify that the project report entitled

### “Fake News Detection with Machine Learning”

Submitted by

Rishabh Jha (MITU17BTIT0015)

Prashansa Bansal (MITU17BTIT0005)

Sahil Gavthankar (MITU17BTIT0011)

Vivek Rote (MITU17BTIT0044)

is a bonafide work carried out by students under the supervision of Prof. Rohini T. Bhosale and it is submitted towards the fulfillment of the requirement of MIT-ADT University, Pune for the award of the degree of Bachelor of Technology (Computer Science & Engineering )

(Prof.Rohini T. Bhosale) (Prof. Rishikesh Yeolekar) ( Prof. Dr. Rekha Sugandhi)

Guide Project Coordinator Professor & Head of Department

Prof. Dr. Kishore Ravande

Principal

Seal/Stamp of the College

Place :

Date:

**DECLARATION**

We, the team members

| Name | Enrollment No | Roll No |
| --- | --- | --- |
| Rishabh Jha  Prashansa Bansal  Sahil Gavthankar  Vivek Rote | MITU17BTIT0015  MITU17BTIT0005  MITU17BTIT0011  MITU17BTIT0044 | 2175050  2175043  2175054  2175065 |
|  |  |  |

Hereby declare that the project work incorporated in the present project entitled “**Fake News Analysis and Detection**” is original work. This work (in part or in full) has not been submitted to any University for the award or a Degree or a Diploma. We have properly acknowledged the material collected from secondary sources wherever required. We solely own the responsibility for the originality of the entire content.

Date:

**Name & Signature of the Team Members**

| Rishabh Jha  Prashansa Bansal  Sahil Gavthankar  Vivek Rote |
| --- |

**Name & Signature of the Guide**

**Prof. Rohini T. Bhosale**

Seal/Stamp of the College

Place:

Date :



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**MIT SCHOOL OF ENGINEERING,**

**RAJBAUG, LONI KALBHOR,**

**PUNE – 412201**

**EXAMINER’S APPROVAL CERTIFICATE**

The project report entitled “**Fake News Analysis and Detection**” submitted by **Rishabh Jha(MITU17BTIT0015),PrashansaBansal(MITU17BTIT0005),SahilGavanthakar(MITU17BTIT0011),Vivek Rote(MITU7BTIT0044)** in partial fulfilment for the award of the degree of “Bachelor of Technology (Information Technology)” during the academic year 2020-21, of MIT-ADT University, MIT School of Engineering, Pune, is hereby approved.

**Examiners:**

1.

2.

**ACKNOWLEDGEMENT**

I express my profound thanks to my Guide **Prof. Rohini T. Bhosale** for her expert guidance, encouragement, and inspiration during this project work.

I sincerely thank **Prof. Dr. Rekha Sugandhi**, Head, Department of Information Technology, MIT School of Engineering, MIT-ADT University, Pune, for providing the necessary facilities in completing the project.

I am grateful to **Prof. Dr. Kishore Ravande**, Principal, MIT School of Engineering, MIT-ADT University, Pune, for providing the facilities to carry out my project work.

I also thank all the faculty members in the Department for their support and advice.

**Name, Enrollment No.**

Rishabh Jha (MITU17BTIT0015)

Prashansa Bansal (MITU17BTIT0005)

Sahil Gavanthakar (MITU17BTIT001)

Vivek Rote (MITU7BTIT0044)

**Abstract**

This report gives an insight about the Fake News Detection in articles. Fake news is the major outbreak nowadays in articles, news. It will use to generally remove fake news from the article by using various algorithms.

The model will focus on identifying fake news sources, based on multiple articles originating from a source. Once a source is labeled as a producer of fake news, we can predict with high confidence that any future articles from that source will also be fake news. Focusing on sources widens our article misclassification tolerance, because we will have multiple data points coming from each source.

**KEYWORDS:** Information Diffusion, Online Social Network, Community Detection, Topic

Modeling, Trend Detection, Influential Nodes

| **CONTENTS** | | | |
| --- | --- | --- | --- |
|  | Certificate | | I |
|  | Declaration | | II |
|  | Acknowledgment | | III |
|  | Abstract | | IV |
|  | Contents | | V |
|  | List of Figures | | VII |
| I | INTRODUCTION | |  |
|  | 1.1 | Introduction | 1 |
|  | 1.2 | Problem Statement | 4 |
|  | 1.3 | Objectives | 11 |
|  | 1.4  1.5 | Modules  Scope | 12 |
|  | 1.7 | Summary | 14 |
| II | CONCEPTS AND METHODS\* | |  |
|  | 2.1 | Dataset | 16 |
|  | 2.2 | Method /Algorithms / Models ……..  ……. | 17 |
| III | LITERATURE SURVEY \* | | 18 |
|  | 3.5 | Research Gap |  |
|  | 3.6 | Problem Definition | 39 |
| IV | 4.1  4.2 | PROJECT PLAN  Project Schedule  Project Cost Estimate | 40  41 |
| VII | 7.1  7.2 | Software Testing\*  Types of testing used  Test cases and results | 77 |
| VII | CONCLUSION AND FUTURE WORK | | 80 |
|  | BIBLIOGRAPHY | | 83 |
|  | Annexure A: Publications Details (first page of a paper publication)  Annexure B: Laboratory assignments on Project Cost estimation, Mathematical modeling, testing. | | 93 |

**CHAPTER 1**

**1.1 INTRODUCTION**

Content Refining is one of the biggest problems faced by Social Media Companies as well as a news organization. While this job was done by humans (Content Moderators) but due to the fact that there are irregularities in human behavior and it takes a huge toll on mental health therefore nowadays this work is done through AI using Machine Learning.

**1.2 The Problem**

Develop a machine learning program to identify when a news source may be producing fake news.

**1.2 Hypothesis**

Our main hypothesis was to create a model that would detect fake news as it is a primary source of spreading hate and panic among masses.

**1.3 Objective**

1. To create a tool that will detect and remove fake news articles.
2. The primary goal was to detect fake news by scanning the string but in the further part of the project we want our model to be able to detect fake photos and video resources.
3. Getting the result is an important step but we also want to work on performance, so that our tool takes less time but does the same task.Accuracy has been the main focus.

**1.4 Modules**

* Data cleaning
* Tokenization
* Data Visualization
* NLP
* Model Fitting

**CHAPTER 2**

**2.1 DATASET**

The data source used for this project is LIAR dataset which contains 3 files with .tsv format for test, train and validation. Below is some description about the data files used for this project.The original dataset contained 13 variables/columns for train, test and validation sets as follows:

* Column 1: the ID of the statement ([ID].json).
* Column 2: the label. (Label class contains: True, Mostly-true, Half-true, Barely-true, FALSE, Pants-fire)
* Column 3: the statement.
* Column 4: the subject(s).
* Column 5: the speaker.
* Column 6: the speaker's job title.
* Column 7: the state info.
* Column 8: the party affiliation.
* Column 9-13: the total credit history count, including the current statement.
* 9: barely true counts.
* 10: false counts.
* 11: half true counts.
* 12: mostly true counts.
* 13: pants on fire counts.
* Column 14: the context (venue / location of the speech or statement).

To make things simple we have chosen only 2 variables from this original dataset for this classification. The other variables can be added later to add some more complexity and enhance the features.

Below are the columns used to create 3 datasets that have been in used in this project

* Column 1: Statement (News headline or text).
* Column 2: Label (Label class contains: True, False)

You will see that the newly created dataset has only 2 classes as compared to 6 from original classes. Below is a method used for reducing the number of classes.

* Original -- New
* True -- True
* Mostly-true -- True
* Half-true -- True
* Barely-true -- False
* False -- False
* Pants-fire -- False

**2.2 Models**

**2.2.1 Naive Bayes**

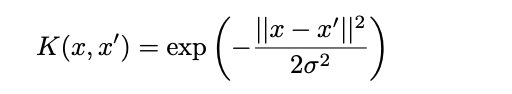
In order to get a baseline accuracy rate for our data, we implemented a Naive Bayes classifier. Specifically, we used the scikit-learn implementation of Gaussian Naive Bayes. This is one of the simplest approaches to classification, in which a probabilistic approach is used, with the assumption that all features are conditionally independent given the class label. As with the other models, we used the Doc2Vec embeddings described above. The Naive Bayes Rule is based on the Bayes’ theorem

P(c|x) = P(x|c)P(c) P(x)

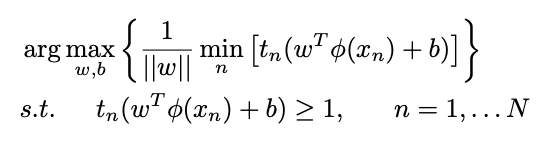
Parameter estimation for naive Bayes models uses the method of maximum likelihood. The advantage here is that it requires only a small amount of training data to estimate the parameters.

**2.3 Support Vector Machine**

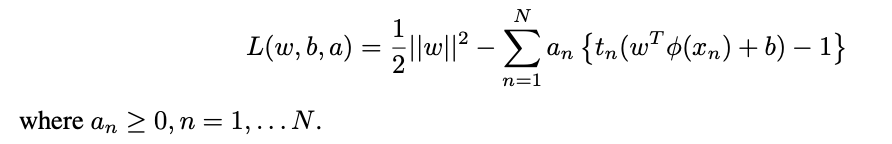
The original Support Vector Machine (SVM) was proposed by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. But that model can only do linear classification so it doesn’t suit most of the practical problems. Later in 1992, Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik introduced the kernel trick which enables the SVM for non-linear classification. That makes the SVM much more powerful. We use the Radial Basis Function kernel in our project. The reason we use this kernel is that two Doc2Vec feature vectors will be close to each other if their corresponding documents are similar, so the distance computed by the kernel function should still represent the original distance. Since the Radial Basis Function is



It correctly represents the relationship we desire and it is a common kernel for SVM. We use the theory introduced in [6] to implement the SVM. The main idea of the SVM is to separate different classes of data by the widest “street”. This goal can be represented as the optimization problem.



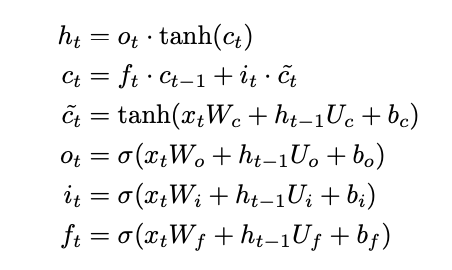
Then we use the Lagrangian function to get rid of the constraints.



Finally we solve this optimization problem using the convex optimization tools provided by Python package CVXOPT.

**2.4 LSTM**

The Long-Short Term Memory (LSTM) unit was proposed by Hochreiter and Schmidhuber[8]. It is good at classifying serialized objects because it will selectively memorize the previous input and 3 use that, together with the current input, to make prediction. The news content (text) in our problem is inherently serialized. The order of the words carries the important information of the sentence. So the LSTM model suits for our problem. Since the order of the words is important for the LSTM unit, we cannot use the Doc2Vec for preprocessing because it will transfer the entire document into one vector and lose the order information. To prevent that, we use the word embedding instead. We first clean the text data by removing all characters which are not letters nor numbers. Then we count the frequency of each word appeared in our training dataset to find 5000 most common words and give each one an unique integer ID. For example, the most common word will have ID 0, and the second most common one will have 1, etc. After that we replace each common word with its assigned ID and delete all uncommon words. Notice that the 5000 most common words cover the most of the text, as shown in Figure 1, so we only lose little information but transfer the string to a list of integers. Since the LSTM unit requires a fixed input vector length, we truncate the list longer than 500 numbers because more than half of the news is longer than 500 words as shown in Figure 2. Then for those list shorter than 500 words, we pad 0’s at the beginning of the list. We also delete the data with only a few words since they don’t carry enough information for training. By doing this, we transfer the original text string to a fixed length integer vector while preserving the words order information. Finally we use word embedding to transfer each word ID to a 32-dimension vector. The word embedding will train each word vector based on word similarity. If two words frequently appear together in the text, they are thought to be more similar and the distance of their corresponding vectors is small. The pre-processing transfers each news in raw text into a fixed size matrix. Then we feed the processed training data into the LSTM unit to train the model. The LSTM is still a neural network. But different from the fully connected neural network, it has cycle in the neuron connections. So the previous state (or memory) of the LSTM unit ct will play a role in new prediction ht.



**CHAPTER 3**

**LITERATURE SURVEY**

# Fake News Detection Using Machine Learning Ensemble Methods: Fake news is detected using textual features and metadata for training various ML models. The author focused mainly on using convolutional neural network (CNN). A convolutional layer is used to capture the dependency between the metadata vectors, followed by a bidirectional LSTM layer. The max-pooled text representations were concatenated with the metadata representation from the bidirectional LSTM, which was fed to fully connected layer with a softmax activation function to generate the final prediction.

# Fake News Detection on Social Media: A Data Mining Perspective :

# With the increasing popularity of social media, more and more people consume news from social media instead of traditional news media. However, social media has also been used to spread fake news, which has strong negative impacts on individual users and broader society. In this article, we explored the fake news problem by reviewing existing literature in two phases: characterization and detection. In the characterization phase, we introduced the basic concepts and principles of fake news in both traditional media and social media. In the detection phase, we reviewed existing fake news detection approaches from a data mining perspective, including feature extraction and model construction.

# Emerging Ubiquitous Systems and Pervasive Networks Detecting Fake News in Social Media Networks : Fake news and Clickbaits interfere with the ability of a user to discern useful information from the Internet services especially when news becomes critical for decision making. Considering the changing landscape of the modern business world, the issue of fake news has become more than just a marketing problem as it warrants serious efforts from security researchers. It is imperative that any attempts to manipulate or troll the Internet through fake news or Clickbaits are countered with absolute effectiveness. We proposed a simple but effective approach to allow users install a simple tool into their personal browser and use it to detect and filter out potential Clickbaits.

# Effective Fake News Detection with Deep Diffusive Neural Network :

In this paper, we have studied the fake news article, creator and subject detection problem. Based on the news augmented heterogeneous social network, a set of explicit and latent features can be extracted from the textual information of news articles, creators and subjects respectively. Furthermore, based on the connections among news articles, creators and news subjects, a deep diffusive network model has been proposed for incorporate the network structure information into model learning. In this paper, we also introduce a new diffusive unit model, namely GDU. Model GDU accepts multiple inputs from different sources simultaneously, and can effectively fuse these input for output generation with content “forget” and “adjust” gates.

# Automatic Fake News Detection: Are Models Learning to Reason? :

We investigate if fact checking models for fake news detection are learning to process claim and evidence jointly in a way resembling reasoning. Across models of varying complexity and evaluated on multiple datasets, we find that the best performance can most often be obtained using only the evidence. This highlights that models using both claim and evidence are inherently not learning to reason, and points to a potential problem in how evidence is currently obtained in existing approaches for automatic fake news detection.

**CHAPTER 4**

**PROJECT PLAN**

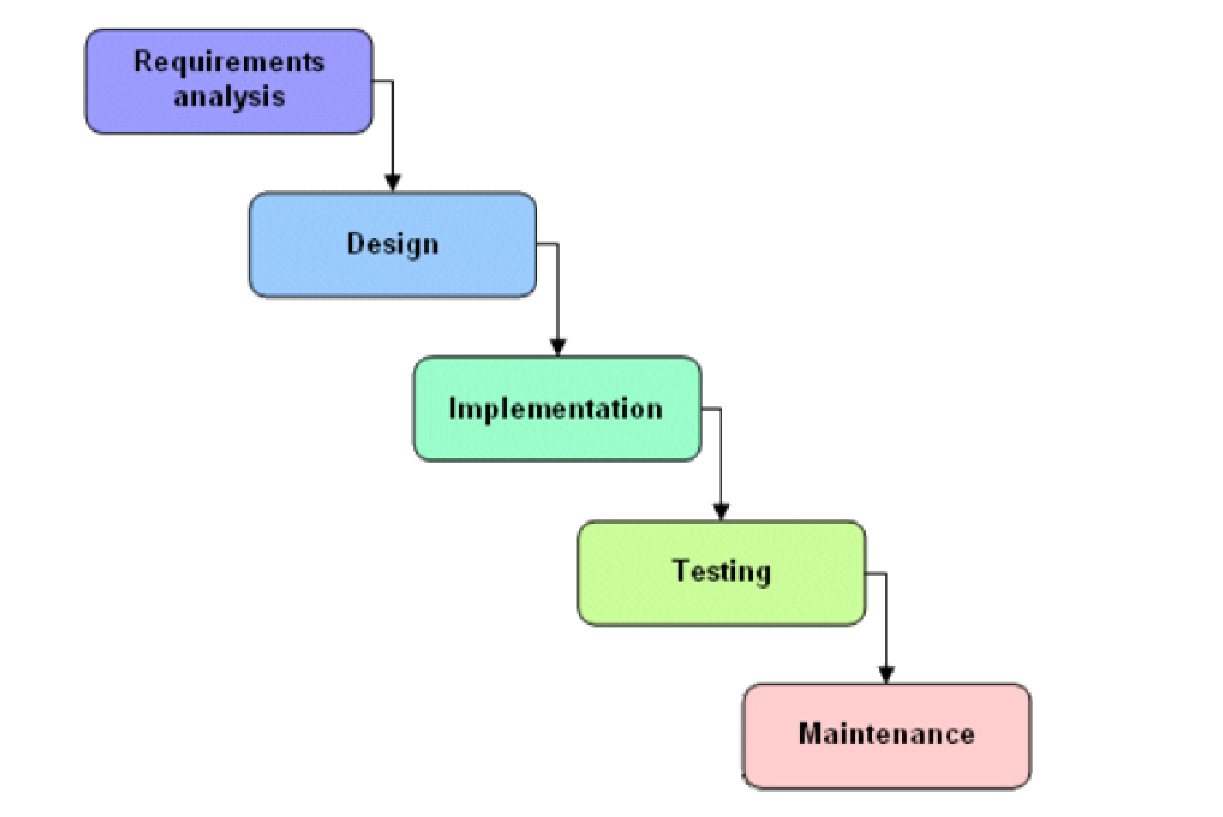
****

Figure 4.1: Project Plan

We approached the system development using the waterfall model depicted in the Based on this model, the required estimates have been stated in Annexure. In order to map our estimates with the steps in a waterfall model, we considered each phase separately and then stated the required estimates.

| **Start Date** | **End Date** | **Topic** | **Time Required**  **(Days)** | **% of Completion** |
| --- | --- | --- | --- | --- |
| 26/03/21 | 01/04/21 | Planning Phase | 7 days | 100% |
| 02/04/21 | 11/04/21 | Recommendation Algorithm | 10 days | 100% |
| 12/04/21 | 18/04/21 | Literature Survey | 7 days | 100% |
| 19/04/21 | 23/04/21 | Sprint 1: Dataset Selection &Environment Setup | 5 days | 100% |
| 24/04/21 | 28/04/21 | Sprint 2: Data Cleaning,Data Manipulation,Data Visualization , EDA & Tokenization | 5 days | 100% |
| 29/04/21 | 05/05/21 | Sprint 3:Training and Testing Data & Fitting Model | 7 days | 100% |
| 16/05/21 | 30/05/21 | Sprint 4:Model Testing & Accuracy testing | 10 days | 100% |
| 31/05/21 | 09/06/21 | Sprint 5:Including modules for image fake news | 15 days | 100% |
| 10/06/21 | 12/06/21 | Sprint 6:Web APP integration | 10 days | 100% |
| 13/06/21 | 17/06/21 | Research Paper | 3 days | 100% |
| 18/06/21 | 20/06/21 | Final report | 3 days | 100% |

**Risk Analysis**

The risks for the Project can be analyzed within the constraints of time and Quality

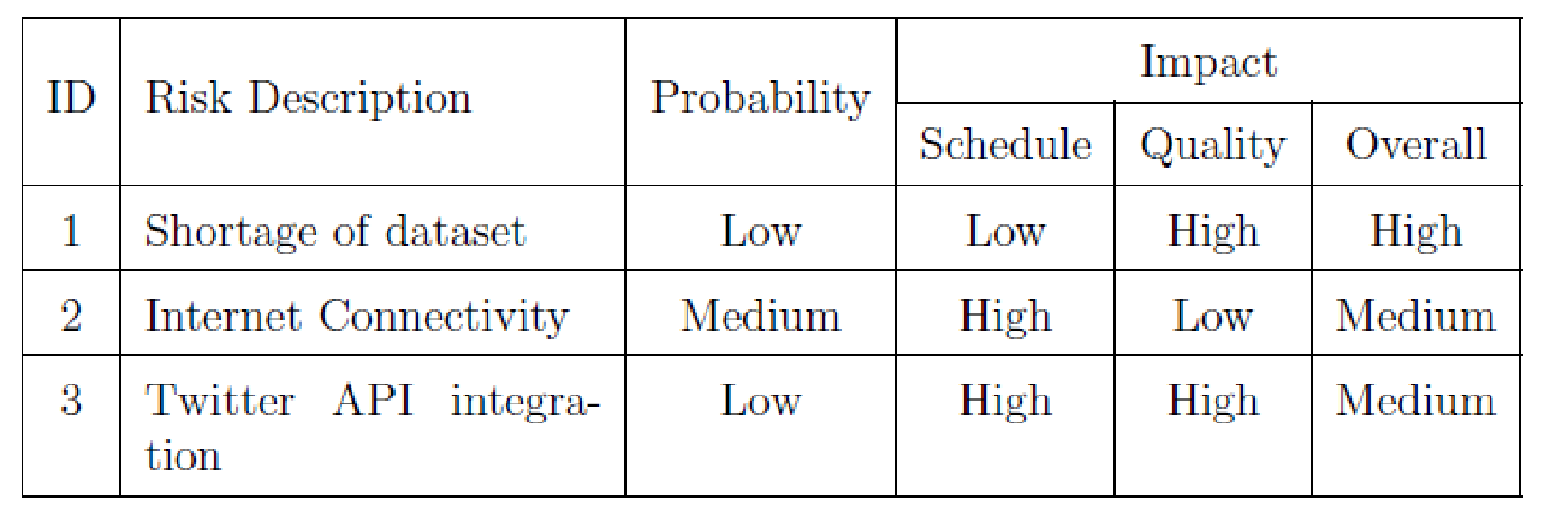


Table 4.1 : Risk table

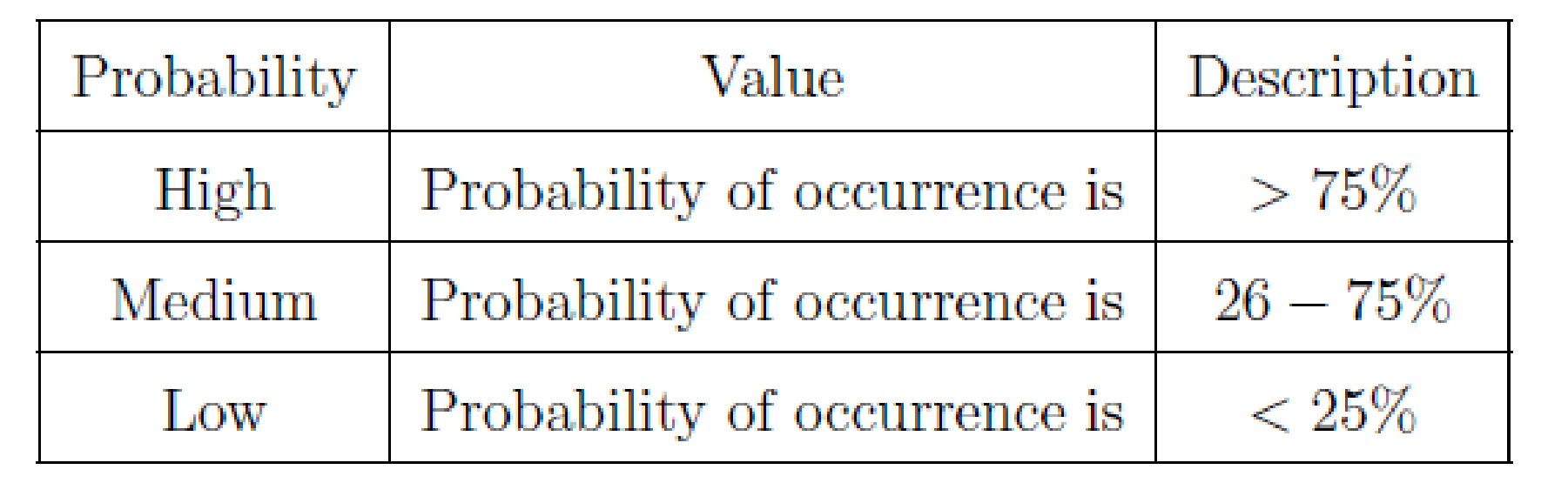


Table 4.2 : Risk probability definitions

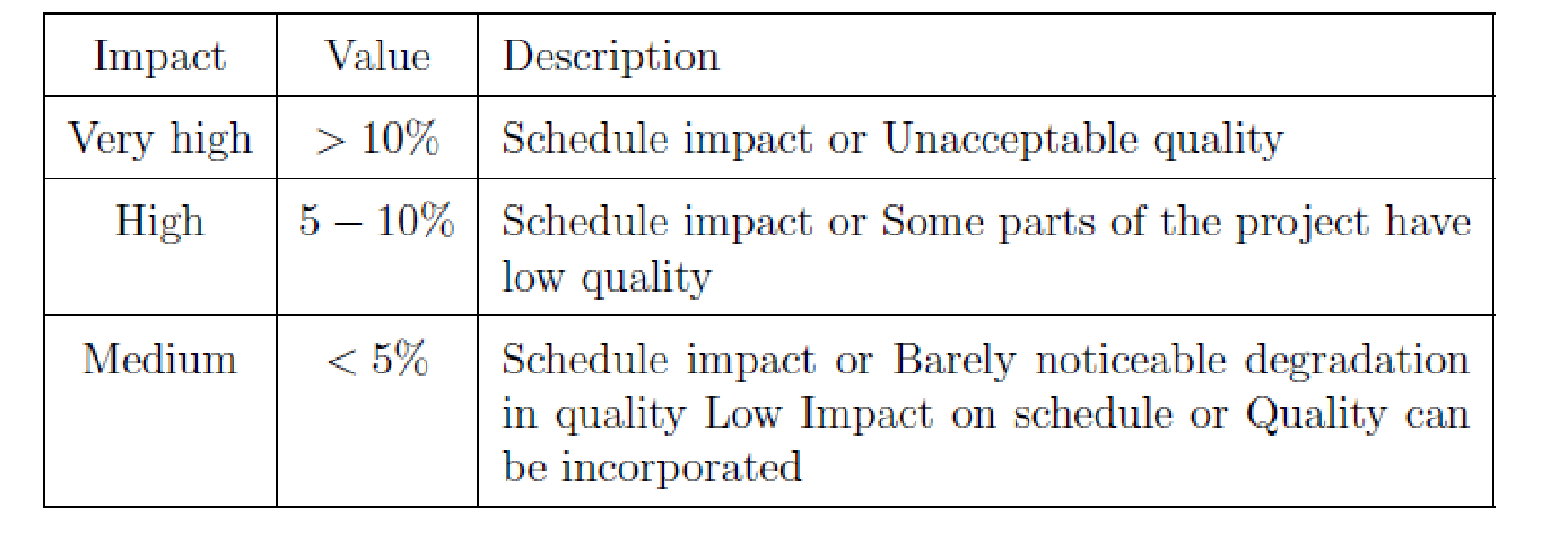
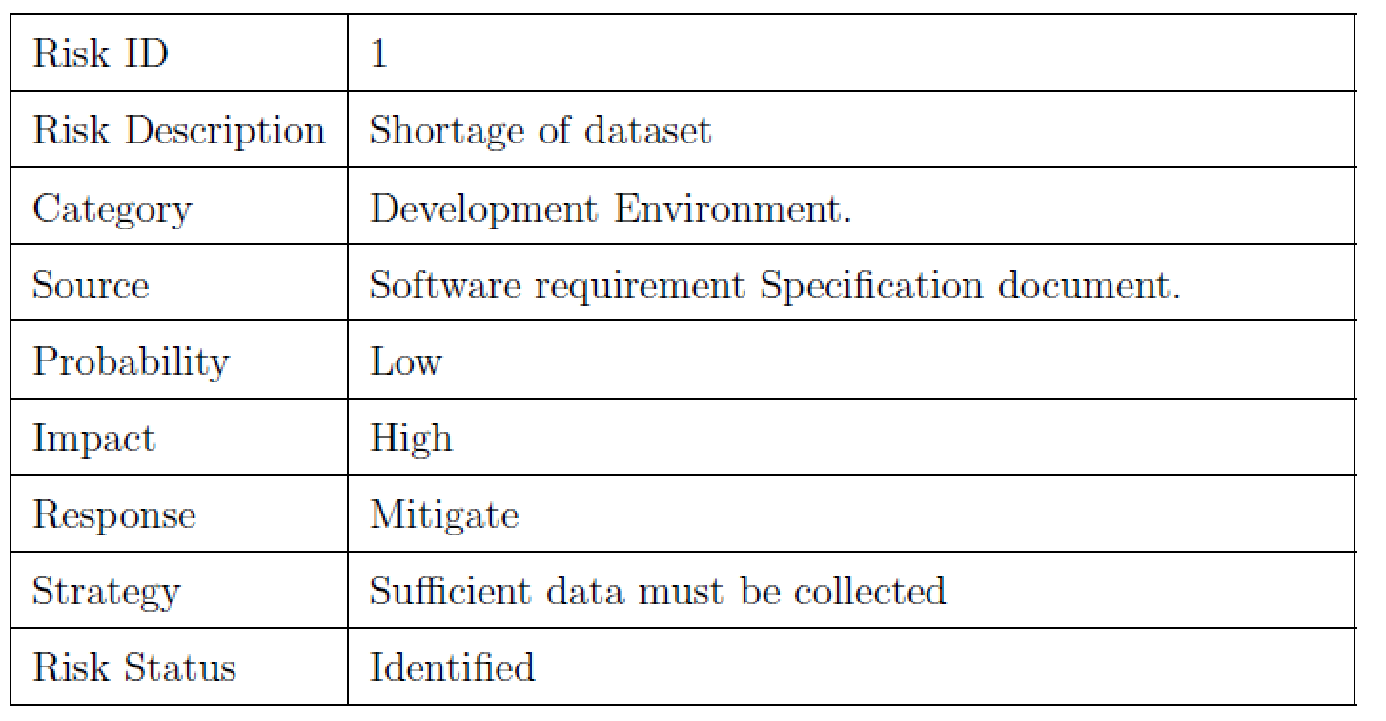
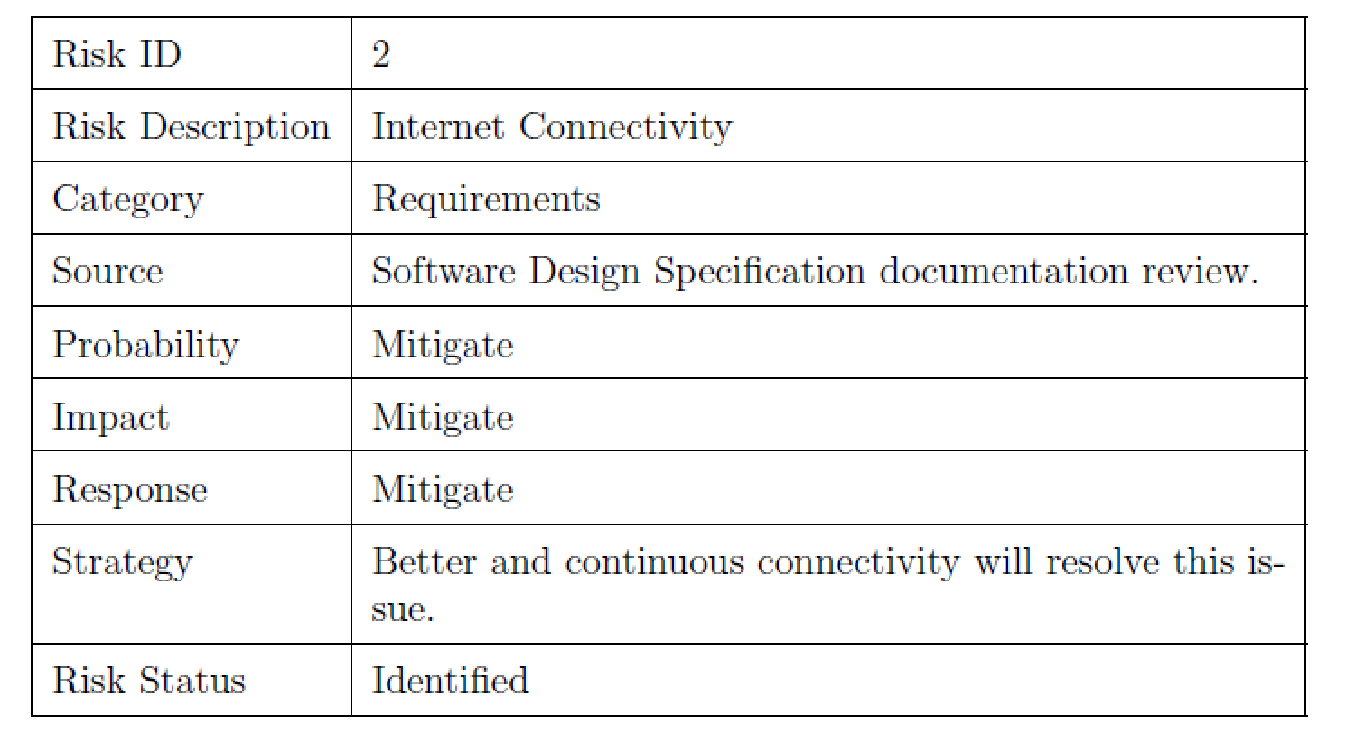


Table 4.3 : Risk impact definition

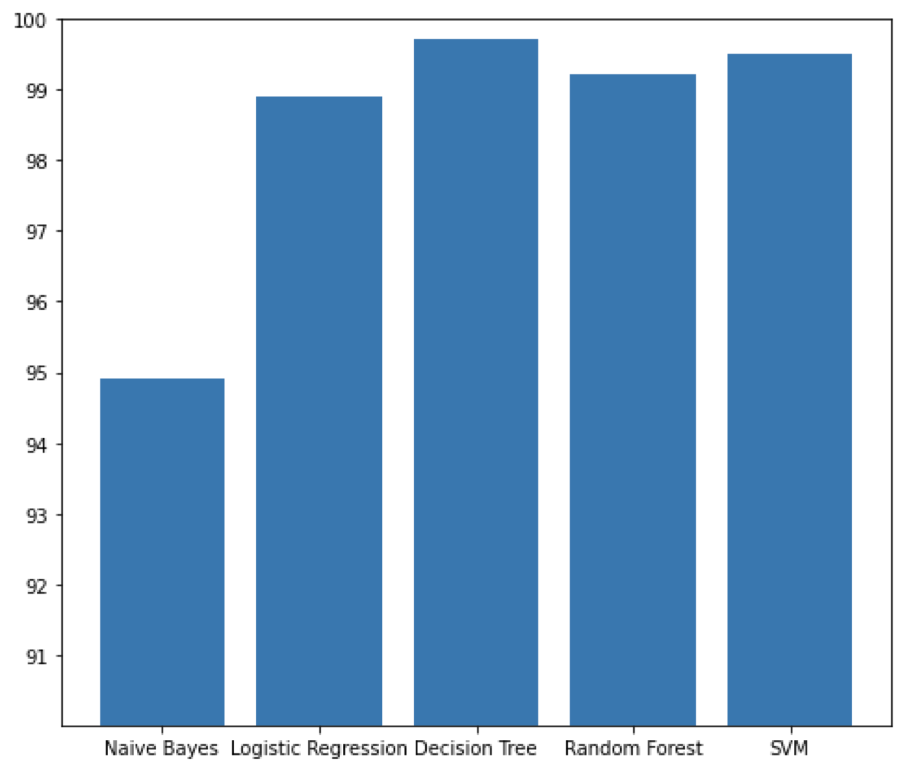
Following are the details for each risk.





**CHAPTER 5**

**RESULT ANALYSIS**



Above graph summarizes the accuracy achieved by each algorithm on the final dataset. It is evident that the maximum accuracy achieved on Decision Tree which is 99.73%. The next highest accuracy is achieved on Support Vector Machine (SVM) which is 99.52%. The next highest accuracy is achieved on Random Forest of 99.22%. The next highest accuracy is achieved on Logistic Regression which is 98.91%. The least accuracy is achieved on Naïve Bayes which is 94.91%. Below Table Represents the name of the classifier and accuracy achieved by classifier.

| **CLASSIFIER** | **ACCURACY** |
| --- | --- |
| **Naïve Bayes** | **94.91%** |
| **Support Vector Machine (SVM)** | **99.52%** |
| **Random Forest** | **99.22%** |
| **Logistic Regression** | **98.91%** |
| **Decision Tree** | **99.91%** |

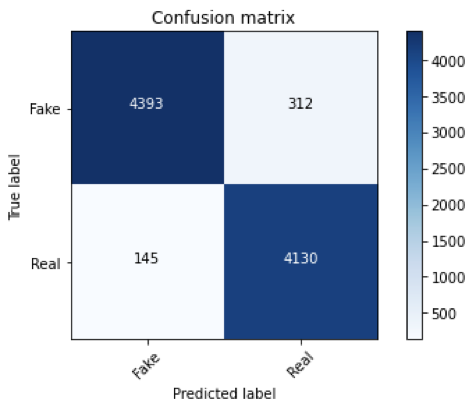
**5.1 Confusion Matrix**

CODE:

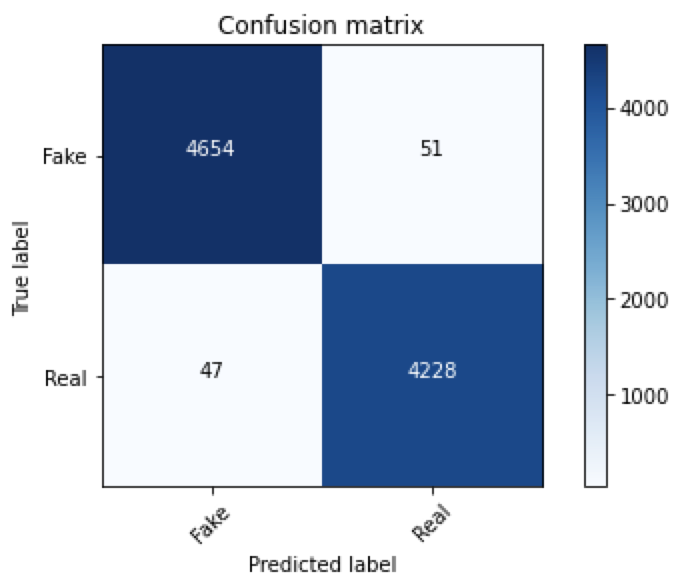
cm = metrics.confusion\_matrix(y\_test, prediction)

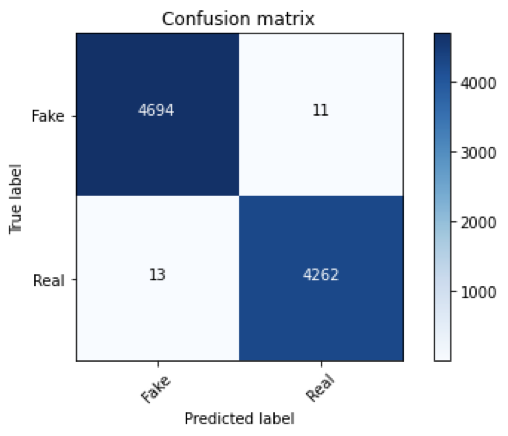
plot\_confusion\_matrix(cm, classes=['Fake', 'Real'])

**5.1.1. Naïve Bayes**

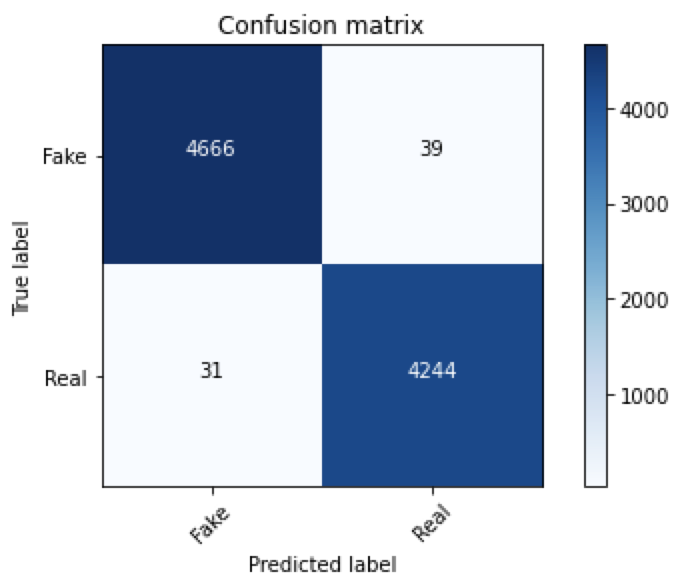


**5.1.2. Logistic Regression**

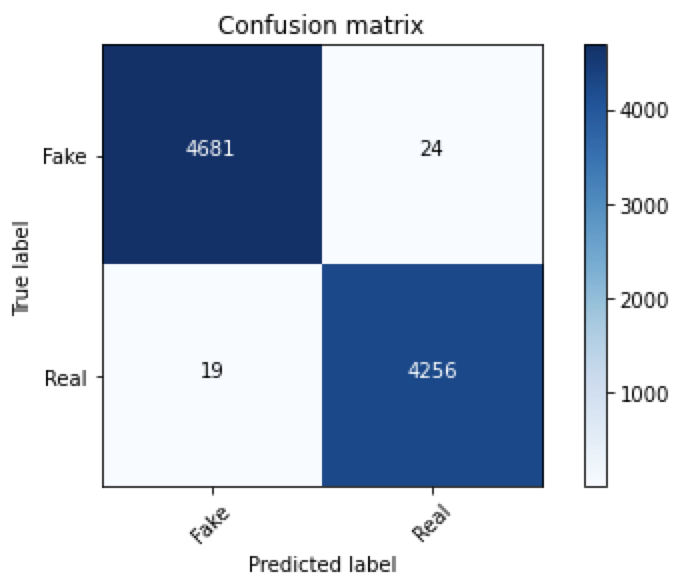
**5.1.3. Decision Tree**



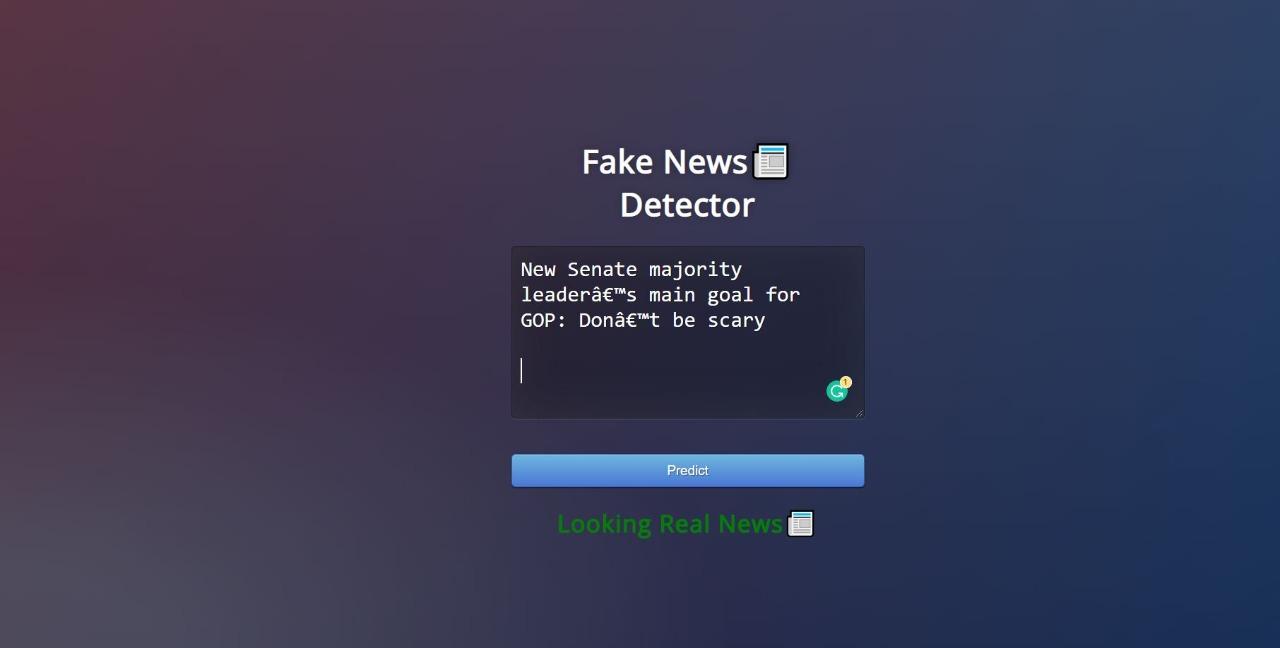
**5.1.4. Random Forest**

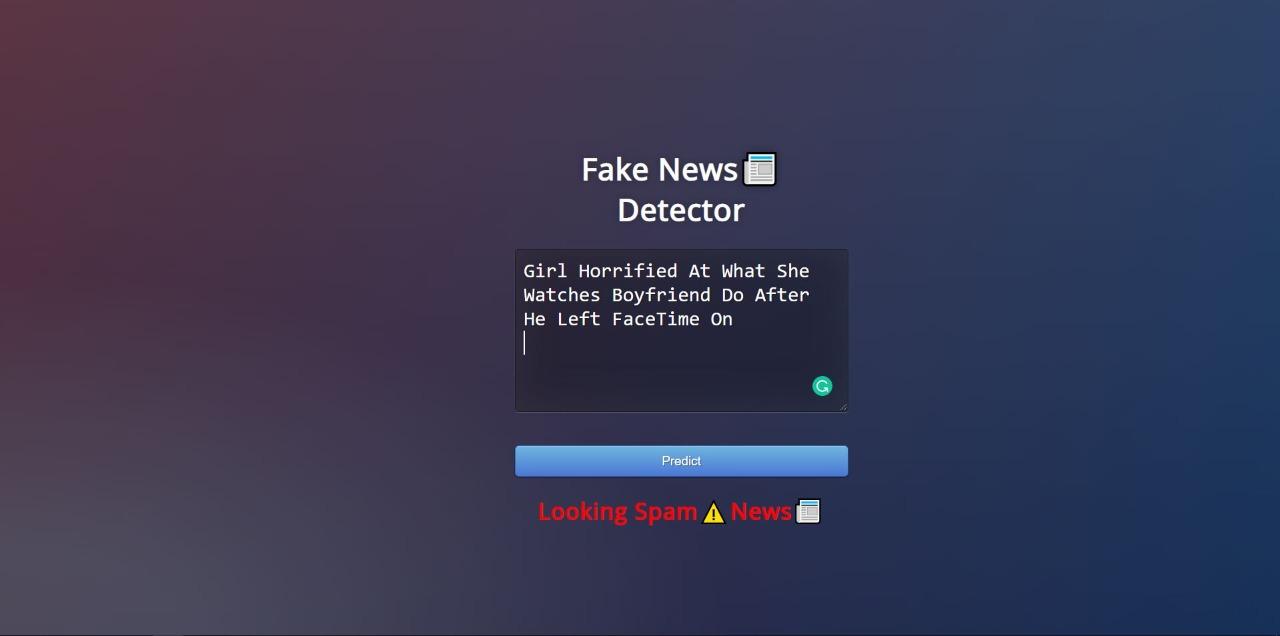


**5.1.5. Support Vector Machine (SVM)**

****

**Application Screenshots:**

****

****

**CONCLUSION AND FUTURE WORK**

The task of classifying news manually requires in-depth knowledge of the domain and expertise to identify anomalies in the text. In this research, we discussed the problem of classifying fake news articles using machine learning models and ensemble techniques. The data we used in our work is collected from the KAGGLE and contains news articles from various domains to cover most of the news rather than specifically classifying political news. The primary aim of the research is to identify patterns in text that differentiate fake articles from true news.

The learning models were trained and parameter-tuned to obtain optimal accuracy. Some models have achieved comparatively higher accuracy than others. We used multiple performance metrics to compare the results for each algorithm. The ensemble learners have shown an overall better score on all performance metrics as compared to the individual learners.

Fake news detection has many open issues that require the attention of researchers. For instance, in order to reduce the spread of fake news, identifying key elements involved in the spread of news is an important step. Graph theory and machine learning techniques can be employed to identify the key sources involved in the spread of fake news. Likewise, real time fake news identification in videos can be another possible future direction.

Finally, this application is only one that would be necessary in a larger toolbox that could function as a highly accurate fake news classifier. Other tools that would need to be built may include a fact detector and a stance detector. In order to combine all of these “routines,” there would need to be some type of model that combines all of the tools and learns how to weight each of them in its final decision.

**BIBLIOGRAPHY**

A. Guille and H. Hacid. (2012), A predictive model for the temporal dynamics of information diffusion in online social networks, In WWW ’12 Companion, pages 1145–1152.

A. Pal and S. Counts. (2011), Identifying topical authorities in microblogs, In WSDM ’11, pages 45–54.

Alex Bavelas. (1950), Communication patterns in task-oriented groups, J. Acoust. Soc. Am, 22(6):725–730, Applications (pp. 115-146): Springer.

Azadeh Nematzadeh, Emilio Ferrara, Alessandro Flammini,Yong-Yeol Ahn.(2014), Optimal Network Modularity for Information Diffusion, Phys. Rev. Lett. 113, 088701 – Published 18 August 2014; Erratum [Phys. Rev. Lett. 113, 259901.](https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.113.259901)

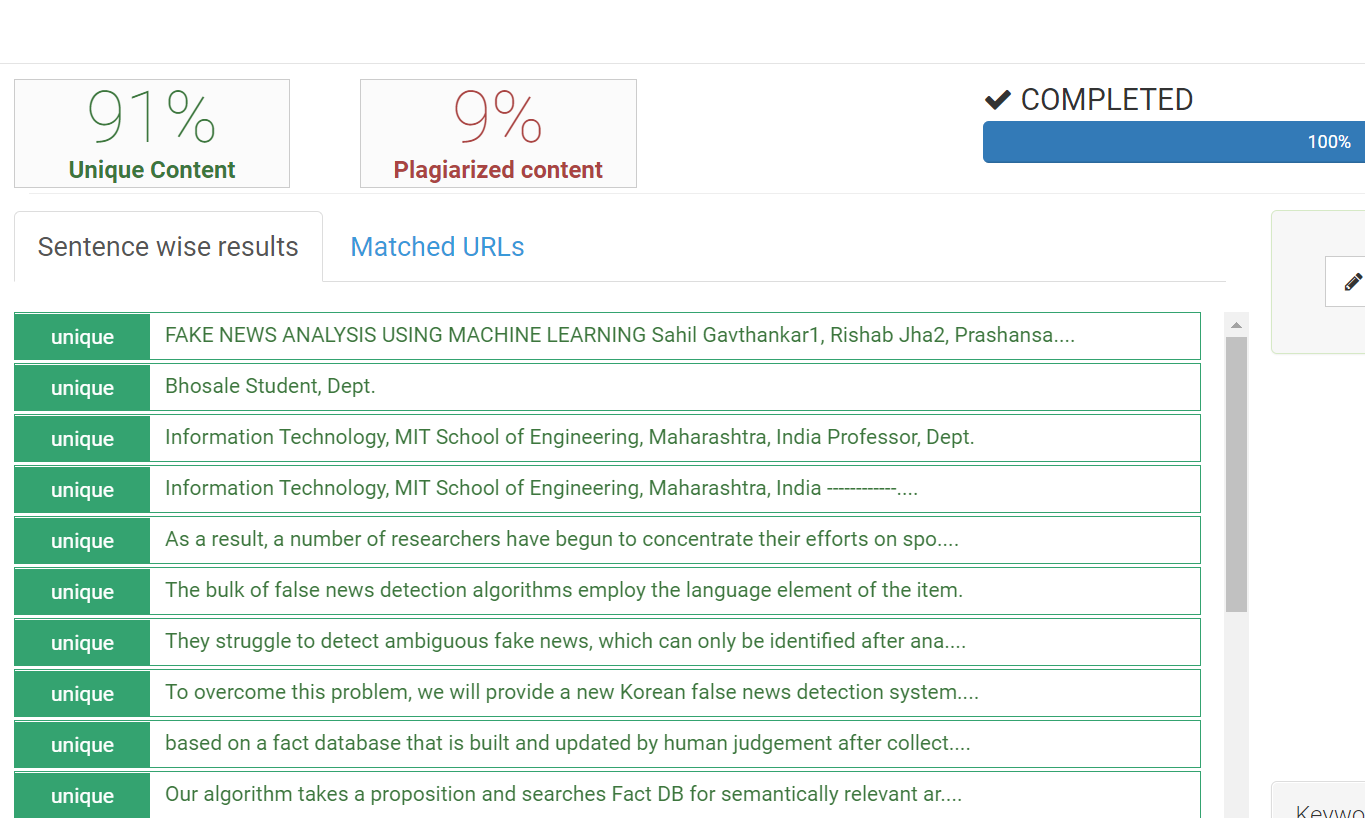
B. W. Kernighan and S. Lin. (1970), An efficient heuristic procedure for partitioning graphs, In the Bell System Technical Journal, vol. 49, no. 2, pp. 291-307, Feb.  
*[doi:10.1002/j.1538-7305.1970.tb01770.x]*

James C. Bezdek. (1981), Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers, Norwell, MA, USA.

Bharathi, S., Kempe, D. & Salek, M. (2007), Competitive influence maximization in social networks. Internet and Network Economics, pp.306–311.

Bonchi, F., (2011), Influence Propagation in Social Networks: A Data Mining Perspective. IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, 2(1), pp.2–2.

**PLAGIARISM REPORT**

****