Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Project Report

On

"Real News Recommendation System"

[Code No.: COMP 308]

(For partial fulfillment of 3rd Year/2nd Semester in Computer Engineering)

Submitted by:

Prajwol Lamichhane (26)

Anukul Parajuli (34)

Abhay Raut (43)

Subarna Subedi (55)

Submitted to:

Dr. Gajendra Sharma

Associate Professor

Department of Computer Science and Engineering

Submission Date: July 23, 2019

Bonafide Certificate

This project work on

“Real News Recommendation System”

is the bonafide work of

“

Prajwol Lamichhane

Anukul Parajuli

Abhay Raut

and

Subarna Subedi

”

who carried out the project work under my supervision.

Project Supervisor

_____

Name: Ms. Rajani Chulyadyo, Ph.D.

Academic Designation: Lecturer

Abstract

In the digital world of enormous flow of data, we could find ourselves at various crossroads that lead to different platforms containing the needed data. Between the options of saving time or checking all platforms, our project merges the crossroads to become one. The project “Real News Recommendation System” utilizes the click behaviour of users and recommends them the best news articles by analysing if the news is fake or not. We felt that a project like “Real News Recommendation System” was needed for the technological aspects relating closely to our nation and the emergence of its numerous news portals. We have created the project as a web application using Python and Django along with the concepts of NLP (Natural Language Processing). The project utilizes the concept of recommendation of online news understanding the importance of relevant news recommendation for the users who read news every day. We hope that our project can impact all Nepali news readers and give them an essence of multi-dimensionality in a concise and time saving manner.

Keywords

ML: Machine Learning

NLP: Natural Language Processing

Table of Contents

List of Figures

Chapter 1 : Introduction

Our project “Real News Recommendation System”, as the meaning rings in the name itself, recommends relevant news articles to the users. More technically, the project recommends news articles (in Nepali language) from different sources or news portals to give a concise and relevant news to the users. The project also consists of fake news detection system as one of its two main modules. The project considers all the beautiful intricacies in the Nepali language and can manipulate the very sophistication it holds and use the same into the process of recommendation. The project is almost entirely based on the concepts from Natural Language Processing to accomplish its goals.

1.1. Background

The project focuses on the process of recommendation in its core. There are two features in our AI-powered recommendation: fake news detection and relevant news recommendation. We went with both these features to distinguish ourselves from the preexisting news portals of Nepal that either do not deliver true news or has no recommendation system. Although the fake news detection algorithm is generic for all kinds of news, we realized the abundance of Nepali news online and hence decided to pursue Nepali fake news articles.

The project required a deep analysis of the all selected news articles in order to execute the dynamic recommendation of the news articles that are relevant to the users. Hence, to make the recommendation accurate, the category of each news article and click behavior of user had to be understood thoroughly so that the project could fulfill its feature of multi-dimensionality. Some other projects that relate to the news recommendation concept had incorporated collaborative filtering and worked with that data, however, the scarcity of user’s personal data led us to use the algorithm that recommended news based on user’s click action in a session.

1.2. Objectives

The major objectives of our project are:

1. To make the task of recommendation and fake news detection automatic.
2. To serve the people whose native language is Nepali.
3. To bridge the technological gap between our mother tongue and AI.

The minor objectives of our project are:

1. To learn the approach of NLP (Natural Language Processing) in AI.
2. To understand machine learning algorithms for classification and supervised regression problem.

1.3. Motivation and Significance

1.3.1 Motivation

Reading articles and blog accounts for a significant amount of time among various online activities. This is because there are a lot of online news portals featuring thousands of articles attracting user activity. This has made it difficult for the users to select suitable articles and blogs of their interest. It may contain many redundant and irrelevant information that substantially take up one's time. But the availability of these thousands of articles has opened vast areas of research for the data science community. So, we have come up with an application that extracts required Nepali articles from the internet and recommends them to the users according to their needs and personal preference. These recommended news articles are legitimate and the ones that are of interest to the user.

1.3.2 Significance

Natural Language Processing is a vast field of research which provides insightful information on how machines deal with human languages. This project can be used as a major starting point for building complex systems that utilizes personal data of users so that they can be recommended better news. The articles scraped

from online news portals can be helpful tools in analyzation and classification of relevant news articles. After the successful analysis of relevant news articles, we were able to recommend them using a simple algorithm and present them to the user. Our project also aids in the comparison of the news articles from different news portals on the same to get a wider perspective over the topic.

1.4 Features

1. Recommending multiple articles to users based on other users with similar kinds of activity on the website.
2. Effective transition probability calculation to filter articles relevant to the users.
3. Using multiple articles recommendation that allows for faster information consumption.
4. Using fake news detection algorithm on Nepalese news so that only the legitimate news is recommended to the users.

Chapter 2 : Related Works

NLP is now being constantly used in the fields to accomplish tasks that were dubbed impossible in the past. It has made advances towards processing and analyzing natural languages written and spoken by humans. In order to solve the problem of minimizing the lengthy articles written in natural languages, many applications related to article news recommendation and fake news detection were made. Some of the popular examples include:

1. **New York Times news recommendation engine** – The New York Times publishes over 300 articles, blog posts and interactive stories a day. Refining the path their readers take through the content – personalizing the placement of articles on their apps and website can help the readers find information relevant to them, such as the right news at the right time, personalized content supplements to major events and stories in their preferred multimedia format. This website uses content based filtering and collaborative filtering for recommending news articles to their users.
2. **Fakenewsai.com** – This website is an AI powered website that detects fake news using neural networks. They analyze websites to see if they are similar to known fake news sites using a neural network. The same technology is used to power other artificial intelligence applications, like Siri and self-driving cars. The website provides a text field to input the URL of the news that users suspect is a fake news. Upon processing the data of the article in their own server, the result is shown on the website i.e. it shows whether the news article on the provided page is fake or not.

Chapter 3 : Design and Implementation

3.1. System Requirement Specification

3.1.1. Software Specification

3.1.1.1. Front-End Tools:

- Programming languages: HTML, CSS, Jinja
- Operating Systems: Windows 10, MAC OS, Linux

3.1.1.2. Back-End Tools:

- Django 2.1
- SQlite3
- Programming Language: Python 3

Python packages used are:

1. Scrapy:

   Scrapy is a Python library for pulling data out of HTML and XML files. It works with your favourite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.

2. Requests:

   Requests is a Python HTTP library, released under the Apache2 License. The goal of the project is to make HTTP requests simpler and more human-friendlier.

3. NLTK (Natural Language Processing):

   NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and

semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

3.1.2. Hardware Specification:

Minimum system requirements for the application are:

- A computer that supports a web browser
- 512 MB of RAM

3.2. System Diagrams

3.2.1 Block Diagram

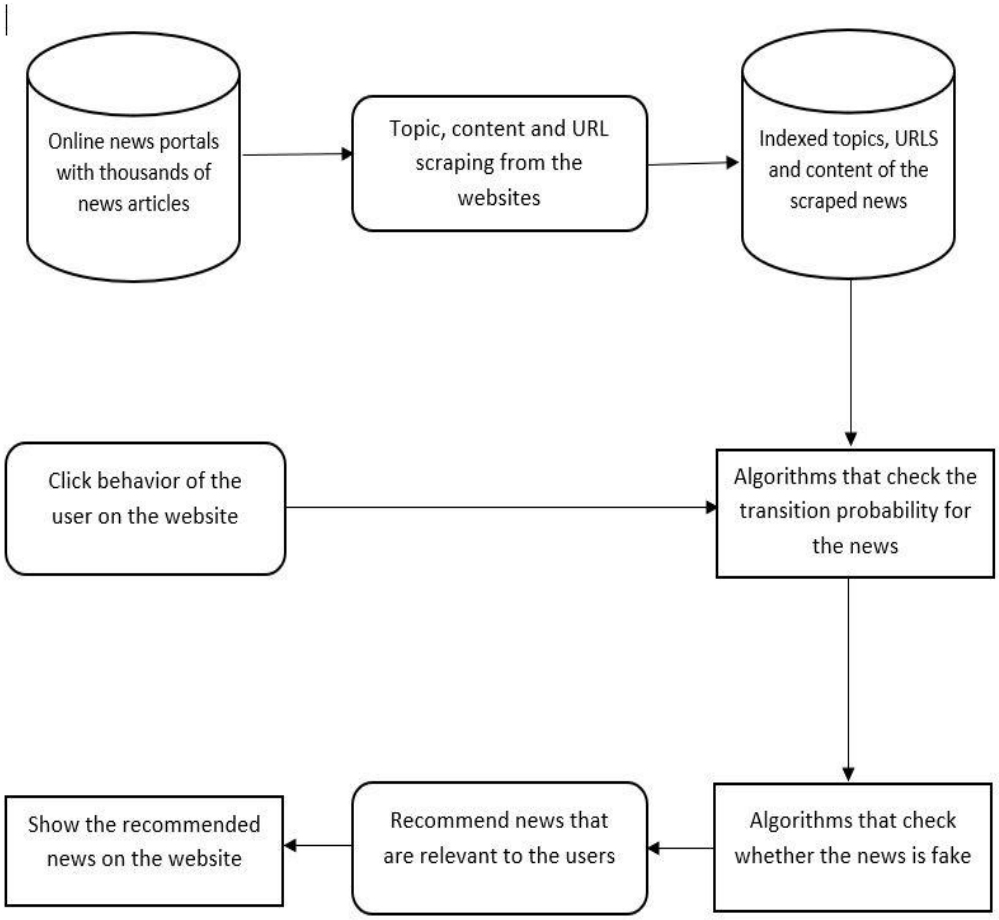


Figure 3-1 : Block Diagram

3.2.2 Use Case Diagram

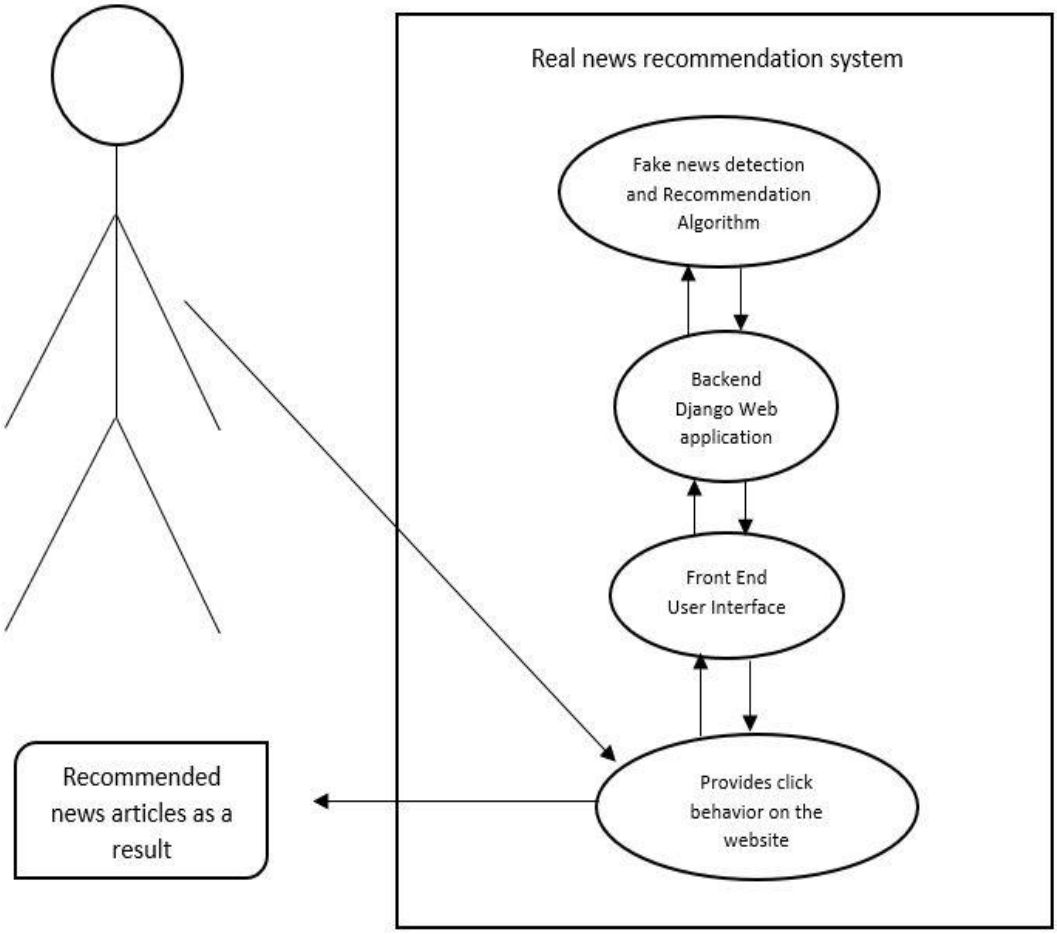


Figure 3-2 : Use Case Diagram

3.2.3 Flow Chart

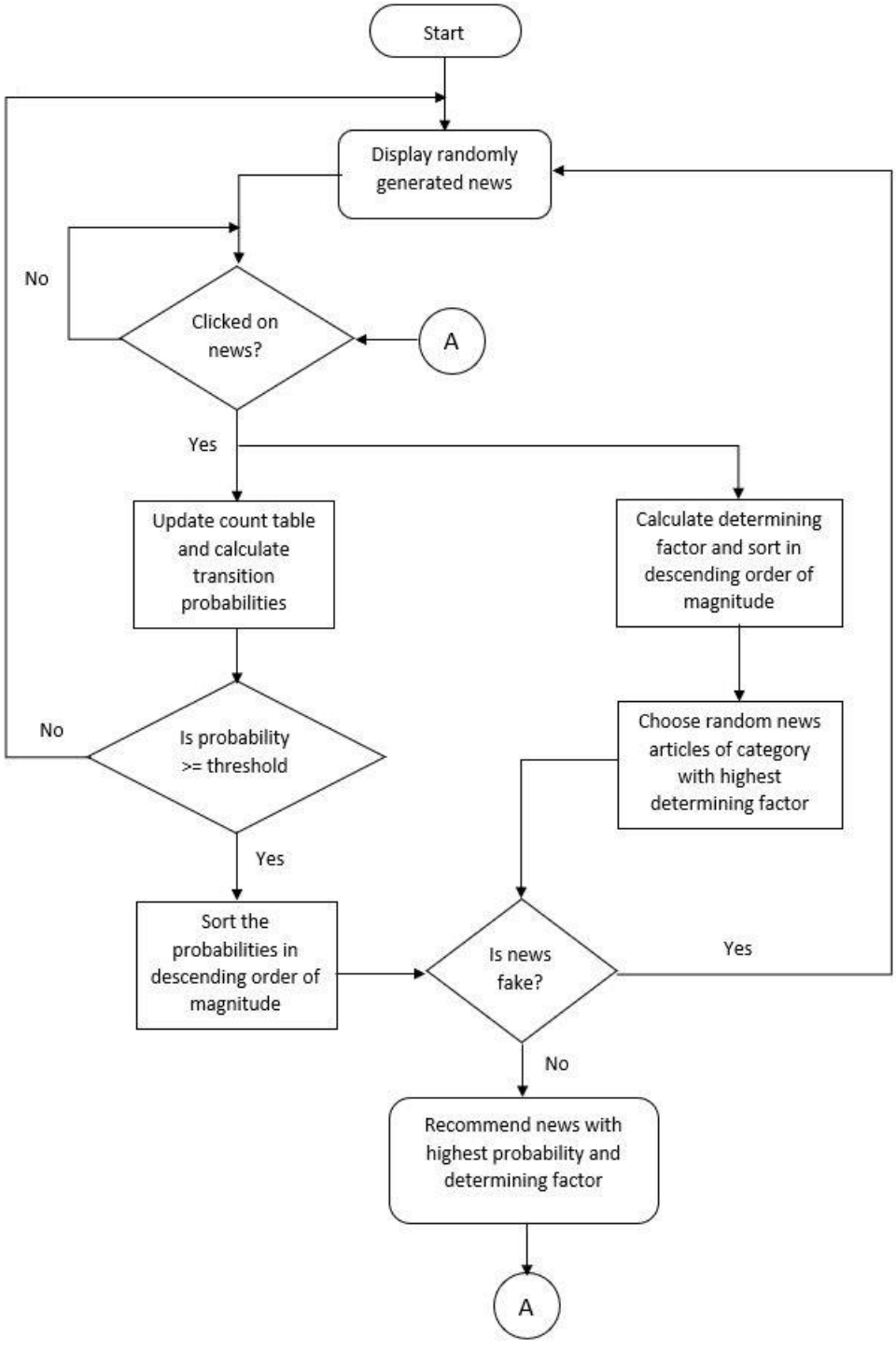


Figure 3-3 : Flow Chart

3.3 Methodology

Scraping

The first and foremost process of our program is the scraping of web sites. In our case, we crawled through some of the most prominent news portals of Nepal for instance, Onlinekhabar, Ujyaaloonline and so on. This was the article collection process wherein we scraped through the news portals for the news articles of different categories that are to be recommend to the users.

The multiple source article collection ensures a multi-dimensionality to the simple objective of recommendation. Hence, the user can find out the latest news articles that contain the relevant and legitimate news articles recommended to them based on their click behaviour on the platform.

The Transition Probability

Transition probability for a news article can be defined as the likelihood of the next news article being viewed after the current news article. This depends on how many times users have transitioned from the current news to the next news on the website. The number of transitions from one news to the other is stored on a count table that only stores the number of clicks.

For e.g.: if there are 4 news on the website, then the counting table can be maintained as follows:

	N1	N2	N3	N4
N1	0	1	10	3
N2	4	0	2	6
N3	2	8	0	5
N4	12	4	2	0

The count table shows how many times the users have transitioned from one news to the other. The diagonals of the matrix are zero as we do not want the news to transition to itself. The transition probability for each news is calculated as follows:

$$P(B/A) = (\text{no of clicks in column B, row A}) / \Sigma (\text{no of clicks in row A}) \quad \ldots (1)$$

After calculating probability of transition for each news, the probability is sorted in descending order of magnitude.

The Category Selection and Rate calculation

These are some other procedures for recommending news articles to users. The category selection procedure utilizes the click behaviour of users during a session. The number of clicks in different category of news is stored locally in the cookies of the browser. These clicks are utilized for determining the category view factor. It is given as follows:

View factor = no of clicks in one category / Σ (clicks in all categories) … (2)

The next parameter to be calculated is rate at which a certain category is being viewed. This is done as:

Rate = no of clicks of that category in certain amount of time / (time) … (3)

These factors are weighted. View factor takes 60 percent weight and rate takes 40 percent weight so that both the factors can play roles in recommendation. The final determining factor is calculated as:

Determining factor = (0.6 * View factor) + (0.4 * Rate) … (4)

Fake News Detection Algorithms

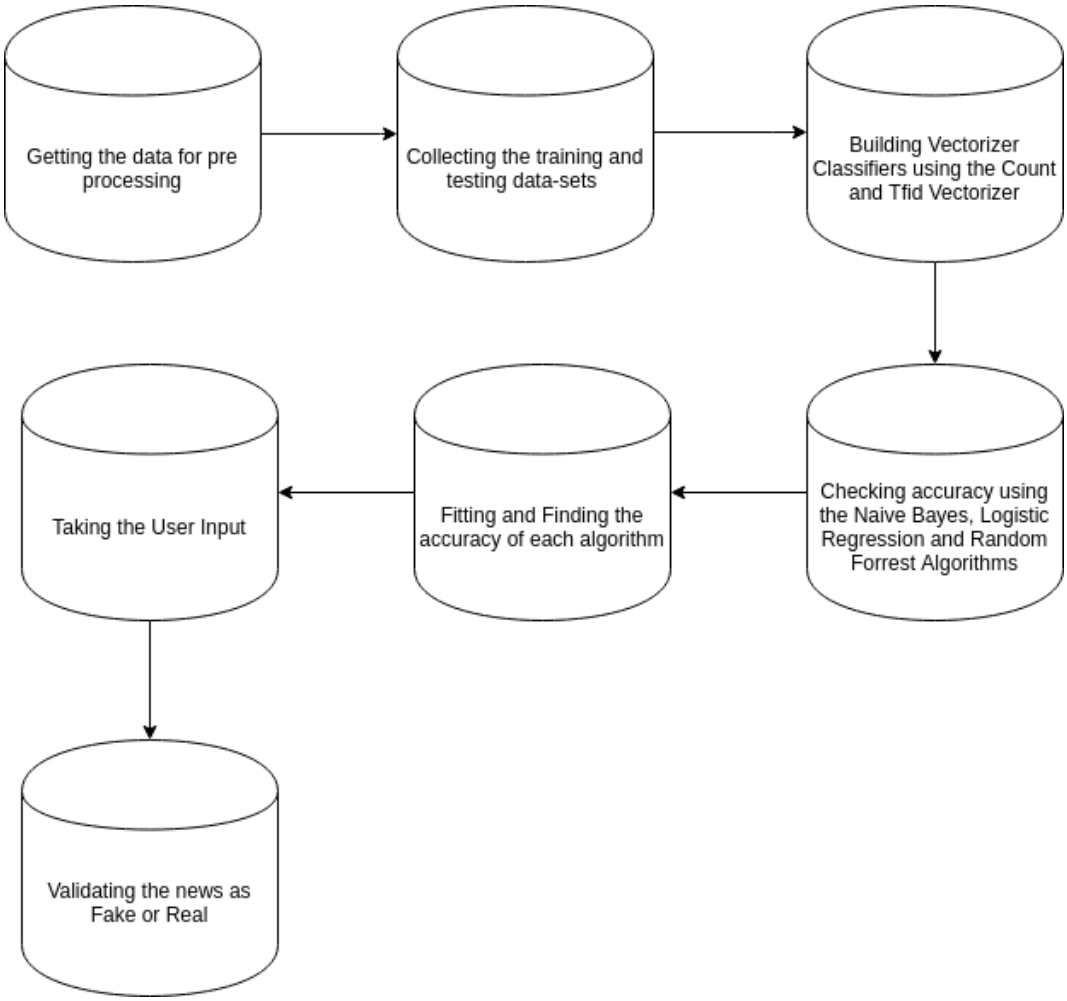
We tried various algorithms to find the accuracy and precision. The accuracy and precision of various algorithms were noted and compared among each other. The various algorithms and their accuracy obtained on our project are:

1. Naive Bayes Algorithm – 86.04%
2. Logistic Regression – 91.2%
3. Random Forrest – 87.3%

So, for our datasets we decided using the Logistic regression as a measure to validate the news.

Flow Diagram

Following diagram depicts how the fake news detector is working:



Algorithms Implemented:

1. **Naive Bayes Algorithms:**

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

2. **Logistic Regression:**

Logistic regression is basically a supervised classification algorithm. Logistic regression is the most famous machine learning algorithm after linear regression. In a lot of ways, linear regression and logistic regression are similar. But, the

biggest difference lies in what they are used for. Linear regression algorithms are used to predict/forecast values, but logistic regression is used for classification tasks. Logistic regression models the data using the sigmoid function.

3. **Random Forest Algorithm:**

Random forest algorithm is a supervised classification algorithm. As the name suggest, this algorithm creates the forest with several trees. In general, the more trees in the forest the more robust the forest looks like. In the same way, in the random forest classifier, the higher the number of trees in the forest gives the high accuracy results. It uses decision trees to predict and classify the results.

Steps Involved in News Recommendation:

The following steps were involved in the news recommendation:

Step 1: Scraping news articles

Scraping consisted of downloading Nepali news data from online news portals. The news articles were scraped, and the topics, URLs and content of the news articles were separated into different text files. These news articles were also used to train the fake news detection machine learning model.

Step 2: Setting up Count table

The count table was set up as a numpy array and stored as a h5.py file. The count table is same for all the users. Each click on the website is registered in the count table. When a user clicks on any news on the website, the control variable is set to the row of that news on the count table. If a user is on news N1, then the control variable is set to its row. If the user transitions from news N1 to news N2, the count on cell (N1, N2) of the count table is increased by 1.

Step 3: Recommendation

Recommendation is done using two methods. The first one is transition probability calculation and the other is rate and view factor calculation. Whenever a user is viewing a news N1, the transition probability from news N1 to any other

news (N2, N3, N4, …) is given by the formula in equation (1). The transition probabilities are stored in a list and sorted in descending order of their magnitude. The next news with the highest transition probability is then recommended to the user.

The next method is the calculation of a determining factor given in equation (4). The determining factor is used for category recommendation and uses the rate of clicks given in equation (3) and view factor given in equation (2) as its parameters. The determining factor is also stored in a list and sorted in descending order of its magnitude. The category with the highest determining factor is then recommended to the user.

Step 4: Fake news detection

At first the datasets were collected. They were then split up into train and test data. The Tfid and Count Vectorizers were used separately to find out their results. The vectorizer outputs were fed to the classifier algorithms mainly: Naive Bayes, Logistic Regression and Random Forest. The algorithms fit the vectorizer inputs and predicts the accuracy given by each algorithm. The user inputs were taken to enter a fake or real news. And, hence the news was validated as a fake or a real news.

Chapter 4 : Discussion

The web app recommends true news to the users using the platform. The fake news detection technique uses algorithms of Machine Learning whereas the recommendation system does not use ML. The recommendation system simply calculates the probability of a user reading a news article and recommends them the news articles with highest reading probability. Fake news detection has been made possible by training the Machine Learning model on the existing fake news. The machine learning model uses supervised learning techniques to detect whether the news is fake or real. The fake news is not recommended to users. Moreover, a separate space or the entry field is also displayed on the web app so that the users can also input the news themselves and validate whether the news is real or fake. So, the system is able to scrap the latest news, validate the news, recommend the news and also provide a platform where user themselves can validate the news by entering on the entry field.

Chapter 5 : Conclusion and Recommendation

Limitations

Our project effectively detects fake news and recommends true and relevant news to the users. However, there are limitations to the implemented algorithms. The transition probability is an effective way of recommending news but the lack of personal data of users on news sites restricted us from using Machine Learning models for recommendation. The weight of view factor and rate have also been hardcoded, but better machine learning techniques can be applied to automate the weight values. The fake news detection algorithm did not have enough fake news to be trained on which makes it less accurate. Without the use of machine learning algorithms, accuracy calculation of the recommender system is very difficult.

Future Enhancements

A tech project is never fully complete because we can always improve and find better algorithms for the system. So, in the future, we would like to improve our recommendation system by using machine learning techniques such as collaborative filtering and content-based filtering after significant amount of data has been collected from the users using our system. We would also like to further improve upon the transition probability calculation by creating mathematics that would calculate its accuracy. The fake news detection and recommendation both can also be improved upon by using deep learning models (Neural networks) for higher accuracy. The scarcity of fake news data and users personal has made our recommendation and fake news detection model less accurate. So, in the future we would like to use GANs (General Adversarial Networks) to create new data that would resemble the real-world data. This data can then be trained on the deep learning models for recommendation system to get more accurate results.

Chapter 6 : References

(2019) Fakenewsai.com

(2019) nltk.org

(2019) Recommendation Systems for PhD thesis, Rajani Chulyadyo

(2018) towardsscience.com/fakenewsdetection

(2018) medium.com/fake_news_detection