A PROJECT STAGE-II REPORT ON

News Web Application Evaluation using Machine Learning Algorithm

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

STUDENT NAME	Exam No:
Vaishnavi Dobe	B190524230
Rutika Chougale	B190524222
Praful Kalhapure	B190524261
Varsha Chavan	B190524221

INDIRA

DEPARTMENT OF COMPUTER ENGINEERING

INDIRA COLLEGE OF ENGINEERING & MANAGEMENT

PARANDWADI, PUNE 410506

SAVITRIBAI PHULE PUNE UNIVERSITY

2023-24

CERTIFICATE

This is to certify that the project report entitles

NEWS WEB APPLICATION EVALUATION USING MACHINE LEARNING

ALGORITHM

Submitted by

STUDENT NAME

Vaishnavi Dobe

Rutika Chougale

Praful Kalhapure

Varsha Chavan

is a Bonafede student of this institute and the work has been carried out by him/her under the

supervision of **Prof. Sunil Rathod** and it is approved for the partial fulfillment of the requirement

of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering**

(Computer Engineering).

Prof. Sunil Rathod   **Dr. Vikas Nandgaonkar**

Guide,   Head,

Department of Computer Engineering   Department of Computer Engineering

Dr. Soumitra Das

Principal,

Indira College of Engineering & Management, Pune

Place: Pune

Date : / /2023

Sign of Internal Examiner   Sign of External Examiner

Project Approval Sheet

A
Project

On

**NEWS WEB APPLICATION EVALUATION USING MACHINE LEARNING
ALGORITHM**

By

Mr. Praful Kalhapure   **Ms. Vaishnavi Dobe**

Ms. Rutika Chougale   **Ms. Varsha Chavan**

At

DEPARTMENT OF COMPUTER ENGINEERING

INDIRA COLLEGE OF ENGINEERING & MANAGEMENT

PARANDWADI, PUNE 410506

[2023-24]

Prof. Sunil Rathod   **Dr. Vikas Nandgaonkar**

Project Guide   HOD Computer Engineering

ACKNOWLEDGEMENT

With immense pleasure, we are presenting this Project Report (Stage-1) as a part of the curriculum of B.E Computer Engineering. We wish to thank all the people who gave us endless support right from the stage the idea was conceived.

We are heartily thankful to **Prof. Sunil Rathod** (Project Guide) whose encouragement, guidance and support from the initial to the final level enabled us to develop an understanding of the subject. We would also like to thank **Dr. Vikas Nandgaonkar** (HOD, Computer Engineering Department), **Prof. Sunil D. Rathod** (BE-Project Coordinator) and **Dr. Soumitra Das** (Principal), for giving us the opportunity to make project on this interesting topic.

This project would not be possible without help of library department who helped us gathering the information from various sources. Lastly, we offer our regards to all those who supported us in any respect during the completion of the stage-1 of our project.

NAME OF THE STUDENTS
1. **Vaishnavi Dobe**
2. **Rutika Chougale**
3. **Praful Kalhapure**
4. **Varsha Chavan**

ABSTRACT

In our modern era where the internet is ubiquitous, everyone relies on various online resources for news. Along with the increase in the use of social media platforms like Facebook, Twitter, etc. news spread rapidly among millions of users within a very short span of time.

There are number of websites increasing in the internet, which provides various facilities like shopping, trading, online betting etc. Some of them are good and some are questionable like whether the website is beneficial or not. Many of them are facing the issues when choosing a website, in order to overcome some of these fraud websites, we proposed a system that evaluates the websites based on the reviews or feedback provided by the customer.

This paper emphasizes that the critiques expressed in the form of reviews are used to analyze the behavior is positive, negative, or neutral and the system is used to rank satisfactory classifiers for sentiments.

TABLE OF CONTENTS

Contents

LIST OF ABBREVATIONS

ABBREVIATION	**ILLUSTRATION**
NLP	Natural Language Processing
KRI	Establish Key Risk Indicators
EMV	Expected Monetary Value
DFD	Data Flow Diagram

LIST OF FIGURES

1. INTRODUCTION

1.1   Introduction

In the digital age, the internet serves as a primary source of information for individuals worldwide. However, in the vast array of websites and content available online, ensuring the authenticity and reliability of news sources has become increasingly challenging. With the proliferation of fake news websites and misinformation campaigns, there is a pressing need for effective tools to evaluate the credibility of online news sources.

The project "News website evaluation using opinion mining" addresses this need by leveraging machine learning techniques to analyse and evaluate the authenticity of news websites. By harnessing the power of sentiment analysis and classification algorithms, the system aims to provide users with valuable insights into the trustworthiness and credibility of news sources.

Through a combination of static and dynamic analysis methodologies, the project offers a comprehensive approach to news website evaluation. The static component utilizes machine learning classifiers to assess the inherent characteristics of news websites, while the dynamic aspect involves real-time analysis of content to determine the probability of truthfulness.

Key technologies such as Python, Flask, Sci-Kit Learn, and web scraping tools are employed to develop a user-friendly web interface that allows users to input URLs and receive accurate evaluations of news website authenticity. Additionally, the system incorporates features for user feedback and interaction, enabling continuous improvement and refinement.

By empowering users with the ability to make informed decisions about the reliability of news sources, this project contributes to combating misinformation and promoting media literacy in the digital era. Through its innovative approach to news website evaluation, the project seeks to enhance trust in online information and foster a more informed society.

The most goals of news websites are to affect the public opinion on certain matters (mostly political). Samples of such websites could also be found in Ukraine, United States of America, Germany, China and much of other countries.

1.2   Overview

As an increasing amount of our lives is spent interacting online through social media platforms, more and more people tend to hunt out and consume news from social media instead of traditional news organizations. The explanations for this alteration in consumption behaviors are inherent within the nature of those social media platforms:

  i. It's often more timely and less expensive to consume news on social media compared with traditional journalism, like newspapers or television.
  ii. it's easier to further share, discuss and discuss the news with friends or other readers on social media. For instance, 62 percent of U.S. adults get news on social media in 2016, while in 2012; only 49 percent reported seeing news on social media.

It had been also found that social media now outperforms television because the major news source. Despite the benefits provided by social media, the standard of stories on social media is less than traditional news organizations. However, because it's inexpensive to supply news online and far faster and easier to propagate through social media, large volumes of faux news, i.e., those news articles with intentionally false information, are produced online for a spread of purposes, like financial and political gain. it had been estimated that over 1 million tweets are associated with the news "Pizzagate" by the top of the presidential election. Given the prevalence of this new phenomenon, "News" was even named the word of the year by the Macquarie dictionary in 2016.

The news is typically manipulated by propagandists to convey political messages or influence for instance, some report shows that Russia has created accounts and social bots to spread false stories. Third, news changes the way people interpret and answer real news, for instance, some news was just created to trigger people's distrust and make them confused; impeding their abilities to differentiate what's true from what's not. To assist mitigate the negative effects caused by the news (both to profit the public and therefore the news ecosystem).

Internet and social media have made the access to the news information much easier and comfortable. Often Internet users can pursue the events of their concern in online form, and increased number of the mobile devices makes this process even easier.

But with great possibilities come great challenges. Mass media have an enormous influence on the society, and because it often happens, there's someone who wants to require advantage of this fact. Sometimes to realize some goals mass-media may manipulate the knowledge in several ways. This result in producing of the news articles that isn't completely true or maybe completely false. There even exist many websites that produce the news almost exclusively. They intentionally publish hoaxes, half-truths, propaganda, and disinformation asserting to be real news – often using social media to drive web traffic and magnify their effect.

The most goals of faux news websites are to affect the public opinion on certain matters (mostly political). Samples of such websites could also be found in Ukraine, United States of America, Germany, China and much of other countries. Thus, news may be a global issue also as a worldwide challenge. Many scientists believe that news issue could also be addressed by means of machine learning and AI. There's a reason for that: recently AI algorithms have begun to work far better on many classification problems (image recognition, voice detection then on) because hardware is cheaper and larger datasets are available.

There are several influential articles about automatic deception detection. In the authors provide a general overview of the available techniques for the matter. In the authors develop two systems for deception detection supported support vector machines and Naive Bayes classifier (this method is employed within the system described during this paper as well) respectively. They collect the info by means of asking people to directly provide true or false information on several topics – abortion, execution, and friendship. The accuracy of the detection achieved by the system is around 70%. This text describes an easy news evaluation supported one among the synthetic intelligence algorithms – naïve Bayes classifier, Random Forest, and Logistic Regression.

The goal of the research is to look at how these particular methods work for this particular problem given a manually labelled news dataset and to support (or not) the thought of using AI for news evaluation.

The difference between these article and articles on the similar topics is that during this paper Logistic Regression was specifically used for news evaluation; also, the developed system was tested on a comparatively new data set, which gave a chance to gauge its performance on a recent data. A. Characteristics of News: They often have grammatical mistakes. They are often emotionally colored.

They often try to affect readers' opinion on some topics. Their content is not always true. They often use attention seeking words and news format and click baits. They are too good to be true. Their sources are not genuine most of the times.

1.3   Motivation

The foremost motivation is the desire to prioritize user satisfaction by understanding their sentiments and preferences. evaluating the web sites might make it greater green and may offer a better way to choose the better website. Hence it became vital to provide you with the answer for the clients which will be the assessment of web sites primarily based totally at the assessment of the clients and price it in line with the emotions recognized with inside the reviews.

The ability to make informed decisions backed by data and user opinions allows organizations to continuously adapt and improve, staying relevant in a dynamic news environment.

It's crucial that we build up methods to automatically evaluate news broadcast on social media. Internet and social media have made the access to the news information much easier and comfortable. Often Internet users can pursue the events of their concern in online form, and increased number of the mobile devices makes this process even easier. But with great possibilities come great challenges. Mass media have an enormous influence on the society, and because it often happens, there's someone who wants to require advantage of this fact. Sometimes to realize some goals mass-media may manipulate the knowledge in several ways.

This result in producing of the news articles that isn't completely true or may be completely false. There even exist many websites that produce news almost exclusively. They intentionally publish hoaxes, half-truths, propaganda, and disinformation asserting to be real news – often using social media to drive web traffic and magnify their effect.

1.4   Problem Definition and Objective

The aim of this system is to make better suggestions about the design of websites or web application by analyzing users opinion and sentiments.

The system will be able to identify the areas where users are dissatisfied with the website. With the use of this information, we can make particular changes in the design of the application that will enhance user satisfaction and engagement.

It will also help to identify issues related to content quality, relevance, and accuracy of the web app.

- Improve the overall user experience by identifying and addressing user concerns related to website design, content quality, and usability.

- Ensure that news articles and content align with user preferences and interests, enhancing content relevance and engagement.

- Utilize user feedback and sentiment analysis to drive data-driven decision-making for continuous improvements in website design and content curation.

- Optimize website/application performance by addressing technical issues and ensuring a seamless user experience

1.5   Project Scope and Limitations

System improvement is likewise taken into consideration as a system subsidized with the aid of using engineering techniques. We have attempted to incorporate & expand new debris for our training debris were observed now no longer all through the coding however additionally all through the analysis, layout phases & in documentation.

Website evaluated the use of opinion mining assignment is taken into consideration as an enlargement of commercial enterprise relations. It contributes lots with the aid of using imparting quick & rapid offerings of sending files letters (formal & casual both) to commercial enterprise because it allows any commercial enterprise to flourish Following change or improvements may be finished in the system.

1. More than one organization may be incorporated via this software.
2. Web offerings may be used to realize the genuine transport popularity of packets.

3. Clients can test the repacked transport popularity online.
4. Distributed database technique in location of centralized technique

1.6   Methodology and Problem Solving

This paper outlines a system developed in three key components. The first part is static and revolves around machine learning classifiers. Initially, four different classifiers were studied and trained, with the optimal classifier selected for final implementation. The second part is dynamic and involves taking keywords or text input from the user and searching online to determine the truth probability of the news. Lastly, the third part focuses on verifying the authenticity of URLs inputted by the user. Python, along with its Sci-kit libraries, serves as the primary technology stack for this project. Python offers a vast array of libraries and extensions, making it conducive to machine learning tasks. The Sci-Kit Learn library stands out as a comprehensive resource for various machine learning algorithms, enabling straightforward evaluation and implementation.

For web-based deployment of the model, Flask is utilized, providing a robust framework for creating web applications. The client-side implementation is facilitated through HTML, CSS, and JavaScript, ensuring an interactive user experience. Additionally, Beautiful Soup (bs4) and requests are employed for web scraping tasks, enabling dynamic retrieval of online information.

This approach leverages the strengths of Python's ecosystem and web development frameworks like Flask to create a versatile and effective system for evaluating news authenticity and URL validity.

A statistical model typically used to model a binary dependent variable with the help of logistic function. Another name for the logistic function is a sigmoid function and is given by:

$$F(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

This function assists the logistic regression model to squeeze the values from (-k,k) to (0,1). Logistic regression is majorly used for binary classification tasks; however, it can be used for multiclass classification.

In the execution of the venture "News site assessment utilizing assessment mining," the emphasis was on fostering an exhaustive assessment framework. The cycle starts with an easy-to-understand login and enrolment connection point to work with client cooperation. Upon fruitful confirmation,

clients are coordinated to a powerful dashboard where they can enter the URL of the news site they wish to assess.

After entering the URL, the site is reviewed inside an assigned window for simplicity of reference. In this manner, after setting off the examination cycle by tapping the assigned button, clients are coordinated to a resulting page highlighting input fields for assessing site usefulness, online entertainment effect, and web composition.

To play out these assessments, datasets were utilized to prepare and test the hidden assessment mining model. This model successfully investigates the site against predefined standards, considering an educated assessment.

The summit of this assessment interaction is introduced to the client on an eventual outcomes page. Here, clients are furnished with a far-reaching evaluation of the site's usefulness, virtual entertainment effect, and plan quality, sorted as either positive or troublesome. Also, visual guides, for example, pie diagrams portraying site content dispersion and line outlines showing the site's usefulness, web-based entertainment effect, and configuration patterns are incorporated to upgrade understanding and work with correlation.

This paper explains the system which is developed in three parts. The first part is static which works on machine learning classifier. We studied and trained the model with 4 different classifiers and chose the best classifier for final execution. The second part is dynamic which takes the keyword/text from user and searches online for the truth probability of the news. The third part provides the authenticity of the URL input by user.

In this paper, we have used Python and its Sci-kit libraries. Python has a huge set of libraries and extensions, which can be easily used in Machine Learning. Sci-Kit Learn library is the best source for machine learning algorithms where nearly all types of machine learning algorithms are readily available for Python, thus easy and quick evaluation of ML algorithms is possible.

We have used Django for the web-based deployment of the model, provides client-side implementation using HTML, CSS, and JavaScript. We have also used Beautiful Soup (bs4), requests for online scrapping.

2. LITERATURE SURVEY

Mykhailo Granik et. al. in their paper shows a simple approach for news detection using naive Bayes classifier. This approach was implemented as a software system and tested against a data set of Facebook news posts. They were collected from three large Facebook pages each from the right and from the left, as well as three large mainstream political news pages (Politico, CNN, ABC News). They achieved classification accuracy of approximately 74%. Classification accuracy for news is slightly worse. This may be caused by the skewness of the dataset: only 4.9% of it is news. Himank Gupta et. al. gave a framework based on different machine learning approach that deals with various problems including accuracy shortage, time lag (BotMaker) and high processing time to handle thousands of tweets in 1 sec. Firstly, they have collected 400,000 tweets from HSpam14 dataset.

Then they further characterize the 150,000 spam tweets and 250,000 non- spam tweets. They also derived some lightweight features along with the Top-30 words that are providing highest information gain from Bag-of-Words model. They were able to achieve an accuracy of 91.65% and surpassed the existing solution by approximately 18%. Marco L. Della Vedova et. al. first proposed a novel ML news application detection method which, by combining news content and social context features, outperforms existing methods in the literature, increasing its accuracy up to 78.8%. Second, they implemented their method within a Facebook Messenger Chabot and validate it with a real-world application, obtaining an application detection accuracy of 81.7%.

Their goal was to classify a news item as reliable; they first described the datasets they used for their test, then presented the content-based approach they implemented and the method they proposed to combine it with a social-based approach available in the literature. The resulting dataset is composed of 15,500 posts, coming from 32 pages (14 conspiracy pages, 18 scientific pages), with more than 2, 300, 00 likes by 900,000+ users. 8,923 (57.6%) posts are hoaxes and 6,577 (42.4%) are non-hoaxes. Cody Buntain et. al. develops a method for automating news website detection on Twitter by learning to predict accuracy assessments in two credibility-focused Twitter datasets: CREDBANK, a crowd sourced dataset of accuracy assessments for events in Twitter, and PHEME, a dataset of potential rumors in Twitter and journalistic assessments of their accuracies. They apply this method to Twitter content sourced from BuzzFeed's news dataset. A feature analysis identifies features that are most predictive for crowd sourced and journalistic accuracy assessments, results of which are consistent with prior work.

They rely on identifying highly retweeted threads of conversation and use the features of these threads to classify stories, limiting this work's applicability only to the set of popular tweets. Since the majority of tweets are rarely retweeted, this method therefore is only usable on a minority of Twitter conversation threads. In his paper, Shivam B. Parikh et. al. aims to present an insight of characterization of news story in the modern diaspora combined with the differential content types of news story and its impact on readers.

Subsequently, we dive into existing news website detection approaches that are heavily based on text-based analysis, and also describe popular news web application. We conclude the paper by identifying 4 key open research challenges that can guide future research. It is a theoretical Approach which gives Illustrations of design issues of news website detection by analyzing the psychological factors

3. SOFTWARE AND HARDWARE REQUIREMENT

3.1   Software Requirements Specification

Table-3.1: System Requirements

Sr. No.	Components	Specification
1	Operating System	Windows 10 or later / MacOS 10.14 or later
2	Technology	HTML, CSS, JavaScript, Python
3	Tool	Visual Studio, Text Editor
4	Database	SQLite

3.1.1. Software Requirements

- Operating system: Windows 2000, XP, 7/8/8.1/1
- Client-side scripting tool: html, JavaScript
- PHP Server: flask
- Database connection: MySQL
- Web browser: Google chrome

3.1.2 Hardware Requirements

- Processor: Any processor above 500 MHz
- RAM: 512MB RAM
- Hard Disk: 10GB
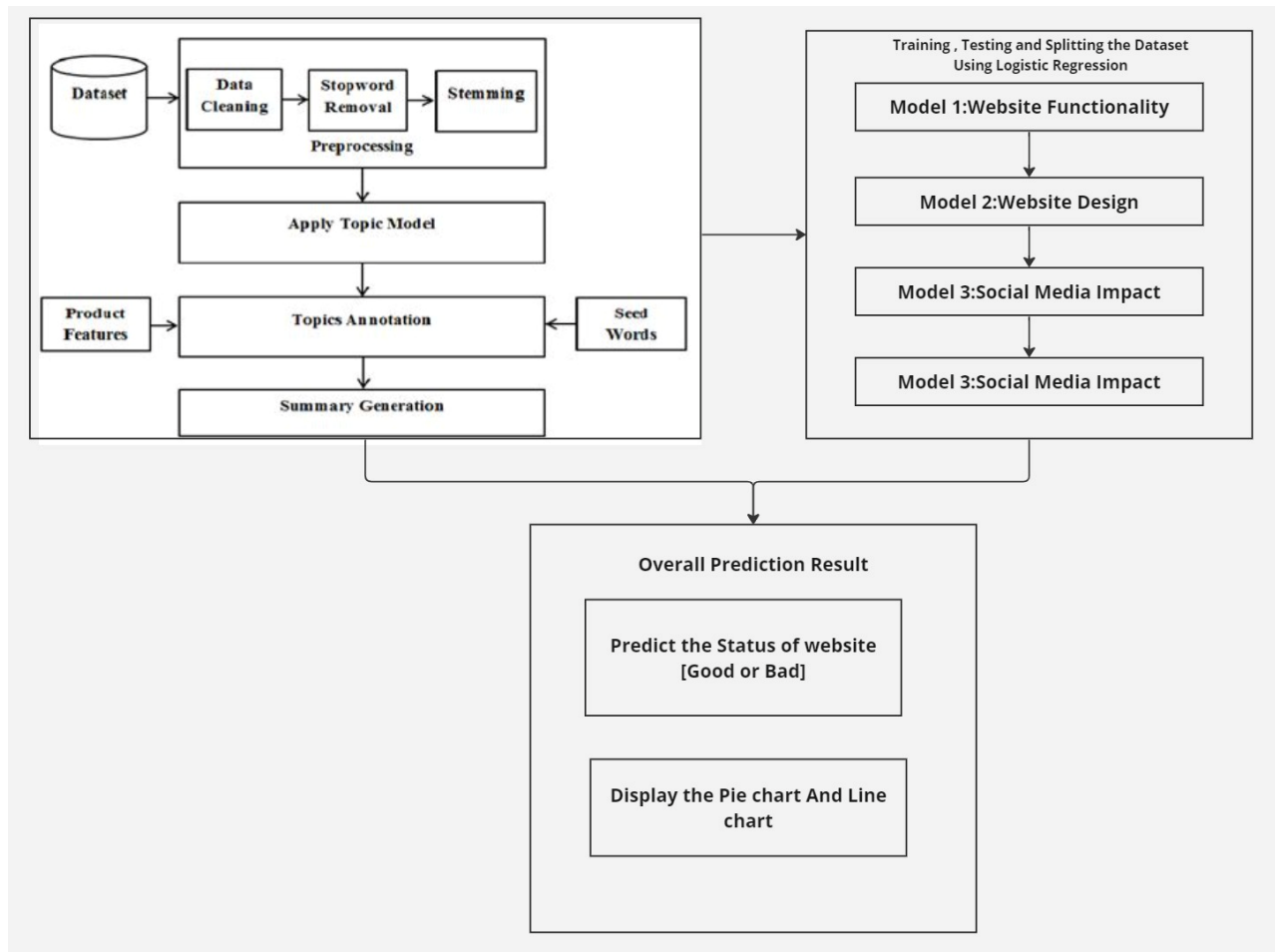
4. SYSTEM ARCHITECTURE

4.1   System Architecture

Figure-4.1: System Architecture

4.1.1   Static Search

The architecture of Static part of news web application evaluation system is quite simple and is done keeping in mind the basic machine learning process flow. The system design is shown below and self- explanatory. The main processes in the design are:
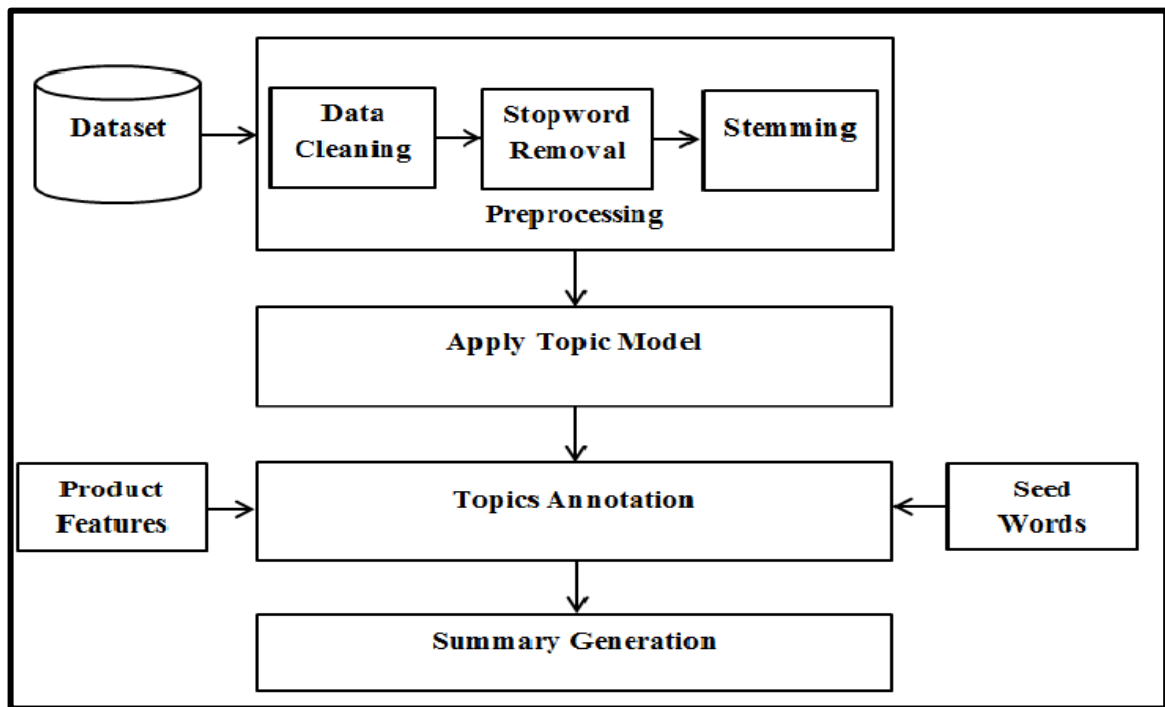
Figure-4.2: Static Search

4.1.2 Dynamic Search

The second search field of the site asks for specific keywords to be searched on the net upon which it provides a suitable output for the percentage probability of that term actually being present in an article or a similar article with those keyword references in it.

4.1.3 URL Search

The third search field of the site accepts a specific website domain name upon which the implementation looks for the site in our true sites database or the blacklisted sites database. The true sites database holds the domain names which regularly provide proper and authentic news and vice versa. If the site isn't found in either of the databases, then the implementation doesn't classify the domain it simply states that the news aggregator does not exist.

4.2 Mathematical Model

1. Data Collection and Preprocessing

   Gather user reviews or comments about the news web application. Preprocess the text data by removing noise, stop words, and irrelevant information.

2. Sentiment Analysis:

Use sentiment analysis techniques to determine the sentiment of each review. Common approaches include:

3. Opinion Mining:

Extract specific opinions or aspects related to the application that users are discussing in their reviews. This can involve techniques like aspect-based sentiment analysis.

4. Feature Extraction:

Identify features or aspects of the news web application that users are evaluating. Common features could include user interface, content quality, speed, reliability, etc.

5. Score Aggregation:

Assign a numerical score to each extracted feature based on the sentiment of related opinions. For example, positive sentiments could contribute to a higher score, while negative sentiments could decrease the score.

6. Weighting:

Assign weights to each feature based on its importance or relevance to the overall evaluation of the news web application. Weighting can be determined through user surveys or domain expertise.

7. Overall Evaluation:

Compute an overall evaluation score for the news web application by aggregating the weighted scores of individual features.

8. Normalization:

Normalize the overall evaluation score to a standardized scale (e.g., 0 to 10 or 0 to 100) for better interpretation and comparison.

9. Visualization and Reporting:

    Present the evaluation results using visualizations such as charts or graphs to provide a clear representation of the application's performance in various aspects.

10. Continuous Improvement:

    Regularly collect and analyze user feedback to update the model, refine feature weights, and improve the evaluation process.
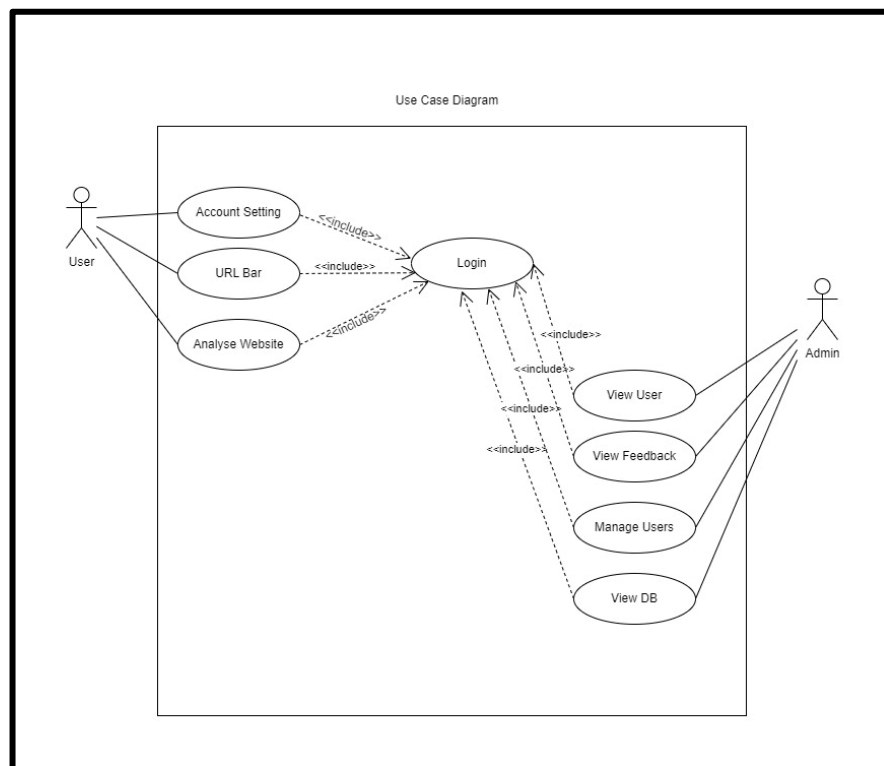
4.3   Use Case Diagram
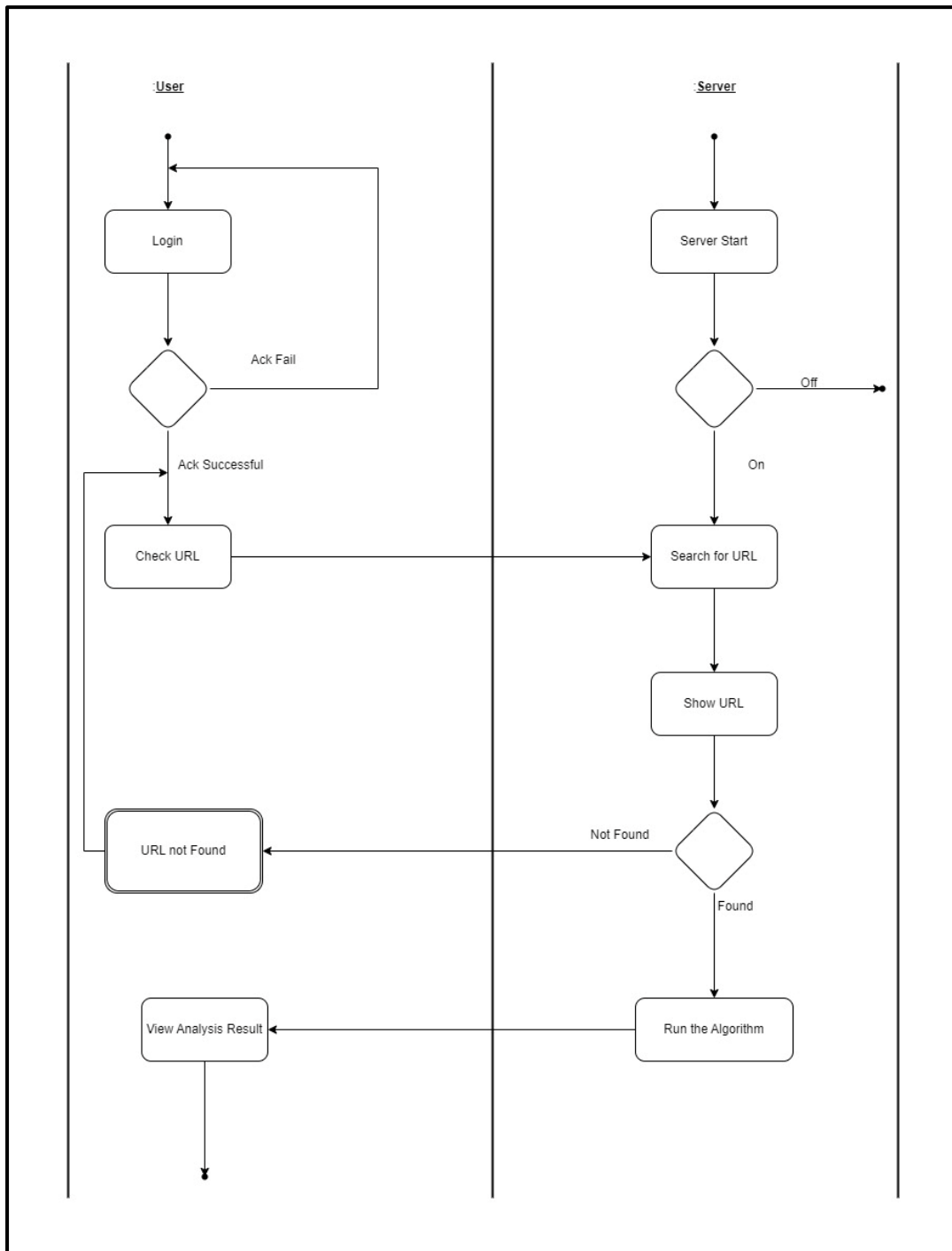

Figure-4.3: Use Case Diagram

4.4 Activity Diagram

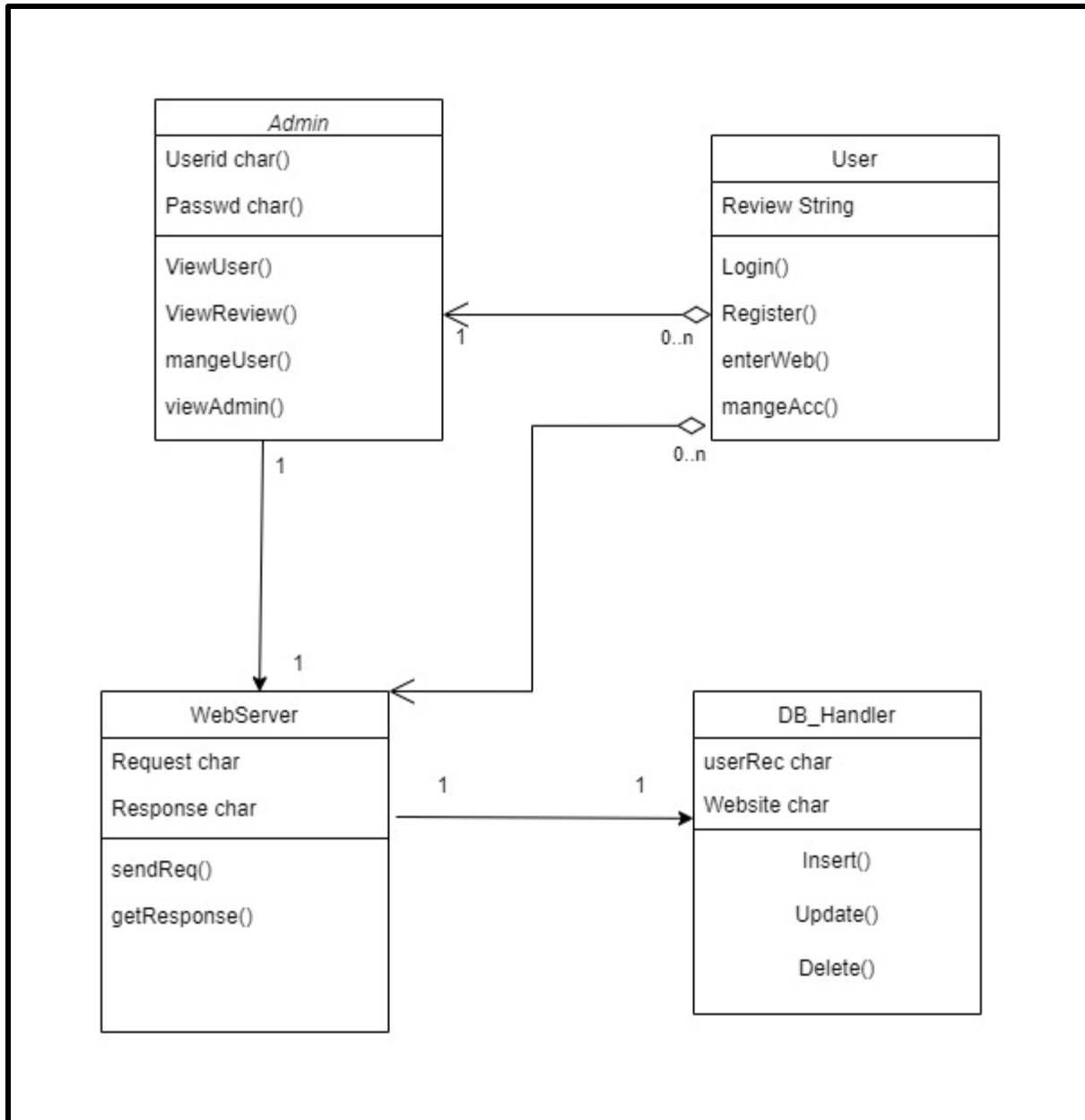**Figure-4.4: Activity Diagram**

4.5 Class Diagram

Figure-4.5: Class Diagram
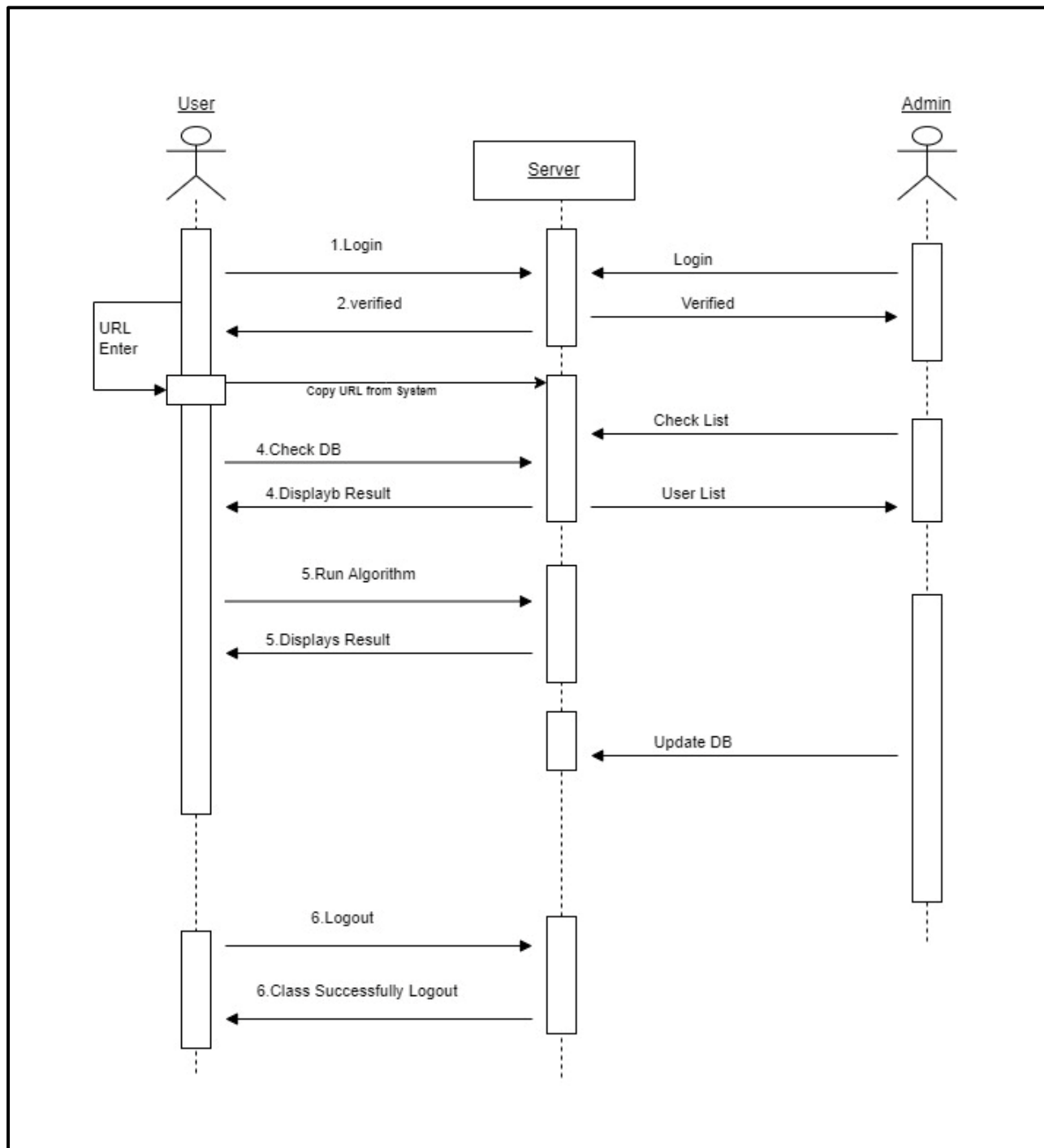
4.6   Sequence Diagram

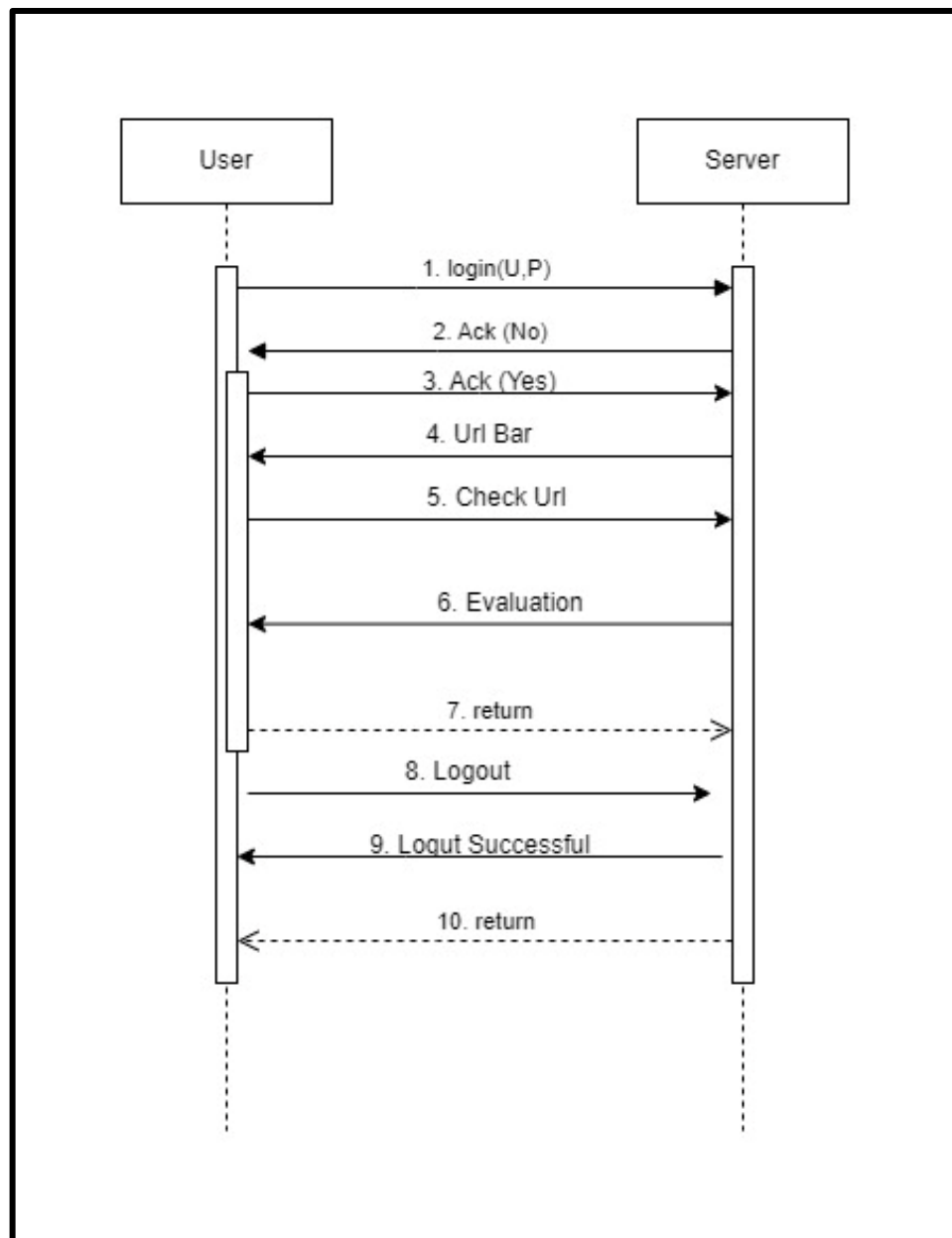**Figure-4.6: Sequence Diagram (User-server-admin)**

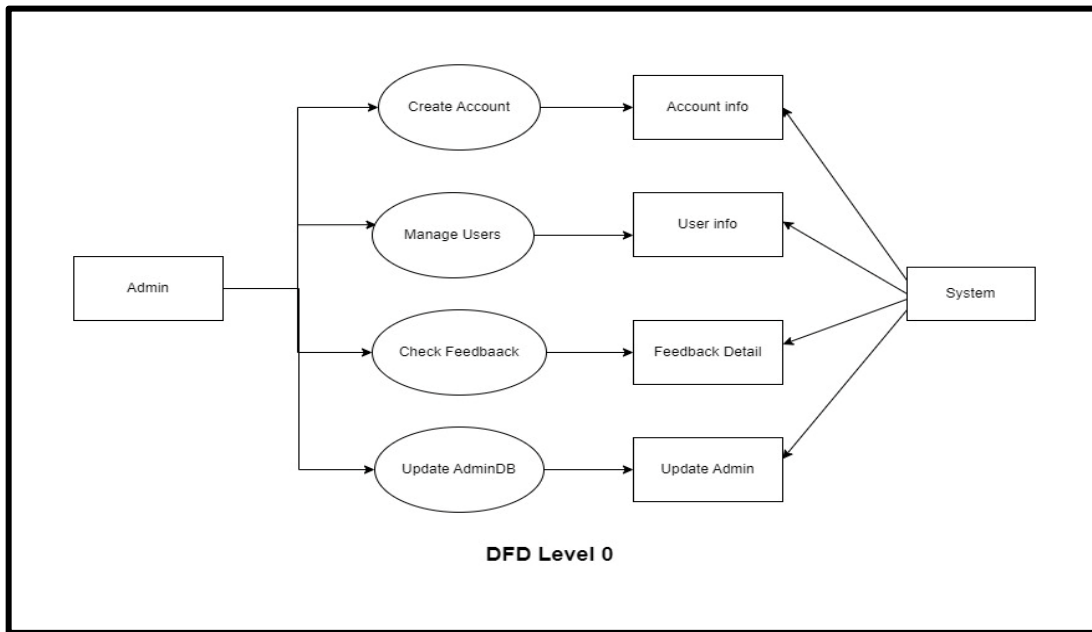**Figure-4.7: Sequence Diagram (User-Server)**

4.7 Data Flow Diagram (DFD)

Figure-4.8: Data Flow Diagram (DFD Level 0)
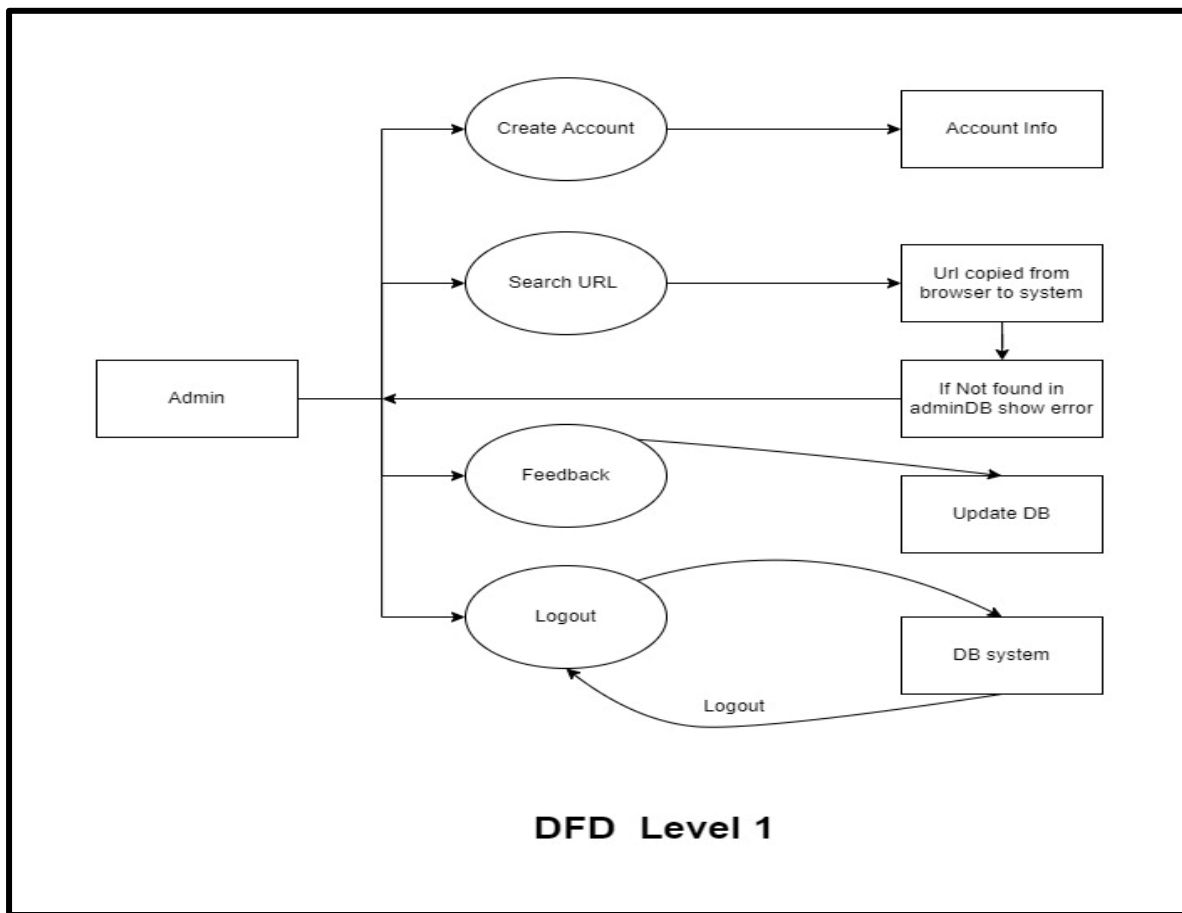

Figure-4.9: Data Flow Diagram (DFD Level 1)

5. PROJECT PLAN

5.1   Project Estimation

5.1.1   Project Resources

For evaluating a news web application using opinion mining, the following project resources may be required:

- A dataset of news articles with user-generated comments or reviews.
- Opinion mining tools or libraries, such as NLTK, spaCy, or Stanford CoreNLP.
- A machine learning framework, such For evaluating a news web application using opinion mining, the following resources may be required:
    - ✓ A dataset of news articles from various sources
    - ✓ A collection of product reviews or social media posts related to the application or news topics.
    - ✓ A machine learning framework or tool for opinion mining
    - ✓ A text preprocessing library or toolkit
    - ✓ A database or data storage solution for storing and retrieving collected data.
    - ✓ A programming language or environment for implementing the opinion mining algorithm.
    - ✓ A web development framework or tool for building the news web application.

5.2   Risk Management

5.2.1   Risk Identification

Risk identification is a crucial step in project management that involves identifying, documenting, and understanding potential risks that could affect the project's objectives. This process helps project managers and stakeholders proactively address and mitigate risks to increase the likelihood of project success. Here's a step-by-step guide on how to identify risks in a project:

1. Project Kickoff Meeting

Start by organizing a project kickoff meeting involving key stakeholders, team members, and subject matter experts. Discuss the project's objectives, scope, constraints, and expectations.

2. Project Documentation Review:

Thoroughly review project documentation such as the project charter, requirements documentation, scope statement, and historical project information to identify potential risks.

3. Stakeholder Interviews and Workshops:

Conduct interviews and workshops with project stakeholders, team members, and subject matter experts to gather their insights and expertise on potential risks related to their areas of involvement or expertise.

4. Brainstorming Sessions:

Organize brainstorming sessions with the project team to generate a list of possible risks. Encourage an open and creative discussion to identify risks related to various aspects of the project.

5. Checklists and Templates:

Use risk identification checklists and templates that cover common risks related to project management, technology, resources, communication, and other project aspects. Tailor these lists to match the specifics of your project.

6. SWOT Analysis:

Conduct a SWOT (Strengths, Weaknesses, Opportunities, Threats) analysis to identify internal and external factors that could impact the project.

7. Industry and Historical Data Analysis:

Analyze industry-specific risks and historical project data from similar projects to identify risks that may be specific to your project type or industry.

8. Assumption and Constraint Analysis:

Review project assumptions and constraints to identify risks associated with these assumptions and constraints potentially impacting the project.

9. Expert Judgment:

Seek expert judgment from industry professionals, consultants, or internal experts to identify risks based on their experience and knowledge.

10. Risk Categories:

Categorize risks into different categories such as technical, financial, organizational, environmental, legal, etc., to ensure comprehensive coverage of potential risks.

11. Feedback and Iteration:

Gather feedback from the project team and stakeholders on identified risks. Iterate and refine the list of risks based on the received feedback.

12. Documentation:

Document identified risks in a Risk Register or Risk Log. Include details such as the risk description, potential impact, likelihood, risk owner, and proposed risk response strategies.

13. Regular Review and Updates:

Continuously review and update the risk identification process throughout the project lifecycle, especially during major project milestones or changes.

By effectively identifying risks early in the project, project managers can develop appropriate risk management strategies and plans to mitigate, transfer, accept, or avoid these risks, ultimately improving the project's chances of success.

5.2.2 Risk Analysis

Risk analysis is a crucial aspect of project management and involves assessing, prioritizing, and understanding the potential risks that could impact a project. The purpose of risk analysis is to evaluate the likelihood and impact of risks, identify their root causes, and develop appropriate strategies to mitigate or manage these risks effectively. Here's a step-by-step approach to conducting risk analysis:

1. Identify Risks:

   Begin by identifying and documenting all potential risks that could impact your project. This includes both internal and external risks. Refer to the Risk Identification process for guidance on this step.

2. Categorize Risks:

   Group the identified risks into categories based on their nature, such as technical, financial, operational, legal, etc. This helps in better understanding and management.

3. Assess Probability and Impact:

   Evaluate the likelihood or probability of each identified risk occurring and assess its potential impact on the project's objectives. Use a scale (e.g., low, medium, high) to rate both probability and impact.

4. Quantitative Analysis:

   For critical risks, employ quantitative analysis techniques such as Expected Monetary Value (EMV) or Decision Tree Analysis to calculate the expected value of a risk considering its probability and impact.

5. Qualitative Analysis:

   Use qualitative analysis techniques like Risk Probability and Impact Assessment (PI Matrix) to prioritize risks based on their probability and impact levels.

6. Risk Assessment Matrix:

   Create a Risk Assessment Matrix that visually represents the risks based on their probability and impact. This helps in focusing on high-risk areas that require attention.

7. Risk Prioritization:

   Prioritize risks based on their assessed levels of probability, impact, and overall risk score. Focus on high-priority risks that need immediate attention.

8. Risk Response Planning:

    Develop appropriate risk response strategies for each identified risk. These strategies can include risk avoidance, risk mitigation, risk transfer, or risk acceptance.

9. Risk Mitigation Strategies:

    Define specific actions and plans to mitigate the high-priority risks. Allocate resources, set deadlines, and establish clear responsibilities for risk mitigation activities.

10. Risk Monitoring and Control:

    Implement a system for ongoing monitoring of identified risks throughout the project. Update risk assessments as needed, track the progress of risk mitigation activities, and continuously identify new risks.

11. Documentation and Communication:

    Document the results of the risk analysis, including the identified risks, their assessment, prioritization, and proposed risk response strategies. Communicate this information to the project team and stakeholders.

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Risk mitigation, monitoring, and management are integral components of the project management process. Effectively addressing and managing risks is essential to ensure project success and minimize the negative impacts of uncertainties. Here's a breakdown of each aspect:

Risk Mitigation

Risk mitigation involves taking proactive steps to minimize the impact or likelihood of identified risks. This is done to reduce the potential negative consequences on the project. Here are the steps involved in risk mitigation:

1. Identify Mitigation Strategies:

    Based on the identified risks, brainstorm and develop strategies to mitigate or reduce the impact and probability of those risks.

2. Develop Action Plans:

Create detailed action plans outlining the steps, responsibilities, and resources required to implement the identified mitigation strategies effectively.

3. Implement Risk Mitigation Actions:

Execute the planned actions and strategies to mitigate the identified risks. This may involve process changes, technology upgrades, contingency planning, etc.

4. Regular Review and Adjustments:

Continuously monitor the effectiveness of the mitigation actions and adjust them as necessary. Sometimes, new risks may emerge, or existing risks may evolve, requiring adjustments to the mitigation strategies.

Risk Monitoring

Risk monitoring involves tracking and assessing identified risks throughout the project lifecycle to ensure that the risk management strategies remain effective and relevant. Here's how you can effectively monitor risks:

1. Establish Key Risk Indicators (KRIs):

Define specific indicators or metrics that will help in tracking the status and severity of identified risks.

2. Regular Risk Reviews:

Conduct regular reviews of identified risks, assessing their current status, any changes in impact or likelihood, and the effectiveness of the mitigation actions.

3. Communication and Reporting:

Ensure that there is clear and consistent communication regarding the status of risks to stakeholders. Provide updates on risk assessments, mitigation progress, and any new risks discovered.

4. Documentation:

   Keep detailed records of all risk-related activities, assessments, and responses. This documentation is essential for future reference and for improving risk management processes in future projects.

Risk Management

Risk management encompasses the overall process of identifying, analyzing, mitigating, and monitoring risks to ensure the successful completion of a project. It's an ongoing and iterative process that involves the following steps:

1. Risk Identification:

   Identify and document potential risks that could affect the project objectives.

2. Risk Assessment:

   Evaluate and prioritize risks based on their probability and impact.

3. Risk Mitigation:

   Develop strategies and action plans to minimize the impact and probability of identified risks.

4. Risk Monitoring:

   Continuously track and assess identified risks throughout the project lifecycle.

5. Feedback and Learning: Collect feedback from risk management activities and use it to improve future risk identification, assessment, and mitigation processes.

5.3.1 Project Task Set

A project task set, often referred to as a Work Breakdown Structure (WBS), is a hierarchical decomposition of the total scope of work to be carried out by the project team. It's a vital tool in project management for organizing and defining the project's activities. Here's a general outline of how you can create a project task set or WBS:

1. Define the Project Scope:

   Clearly define and document the overall project scope. Understand the project's objectives, deliverables, and what needs to be accomplished.

2. Identify Major Deliverables:

   Break down the project into major deliverables or outcomes that need to be produced or achieved by the project.

3. Decompose Deliverables into Sub-Deliverables:

   Break down each major deliverable into smaller, manageable components or sub-deliverables.

4. Identify Work Packages:

   Break down sub-deliverables further into work packages, which are smaller, specific tasks that can be assigned to individuals or teams.

5. Create a WBS Hierarchy:

   Organize the work packages and sub-deliverables in a hierarchical structure to clearly display the breakdown of the project scope.

6. Assign Codes and Identifiers:

   Assign unique codes or identifiers to each work package and sub-deliverable to track and manage them effectively.

7. Estimate Time and Resources:

   Estimate the time and resources needed for each work package. This helps in project scheduling and resource allocation.

8. Define Dependencies

   Identify dependencies between different tasks or work packages. Determine what tasks need to be completed before others can start.

9. Create a Task List:

   Develop a comprehensive list of tasks, including their descriptions, assigned personnel, estimated durations, dependencies, and deadlines.

10. Review and Validate:

    Conduct a thorough review of the task set with project stakeholders to ensure that all project requirements and objectives are adequately captured.

11. Update and Refine:

    Continuously update and refine the task set as the project progresses. Adjust timelines, dependencies, and resources based on actual progress and changing project requirements.

12. Tools and Software:

    Utilize project management software or tools to create, manage, and visualize the project task set effectively.

5.3.2 Task Network

A Task Network, often referred to as a Network Diagram or PERT Chart (Program Evaluation and Review Technique), is a graphical representation of project tasks and their dependencies. It's used in project management to visualize the flow and relationships between different tasks. The diagram helps project managers in scheduling, planning, and managing complex projects. Here's how you can create a Task Network: Steps to Create a Task Network:

1. Identify Tasks:

   List all the tasks that need to be completed for the project.

2. Define Dependencies:

   Determine the relationships and dependencies between tasks. Tasks can be dependent on other tasks for start or finish.

3. Draw Nodes for Tasks:

   Represent each task as a node (or box) in the diagram. Label each node with the task name or a unique identifier.

4. Connect Nodes with Dependencies:

    Draw arrows between nodes to represent task dependencies. The arrow points from the predecessor (dependent) task to the successor (dependency) task.

5. Indicate Task Durations:

    Optionally, add task duration estimates or other relevant information to each node.

6. Critical Path Analysis:

    Identify the critical path, which is the longest path through the network diagram. It represents the shortest time in which the project can be completed.

7. Calculate Task Times:

    Use the estimated task durations and critical path analysis to calculate the total time required to complete the project.

8. Add Milestones:

    Include milestones to signify significant achievements or events in the project.

9. Review and Optimize:

    Review the task network to ensure it accurately represents the project's workflow. Optimize dependencies and durations if necessary.

6. PROJECT IMPLEMENTATION

```python
from flask import Flask, request, render_template, redirect, session
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from flask_sqlalchemy import SQLAlchemy
import bcrypt
import requests
from bs4 import BeautifulSoup
import pandas as pd
import matplotlib.pyplot as plt
from io import BytesIO
import base64
import matplotlib
from flask import jsonify
from sklearn.svm import SVC
matplotlib.use('Agg')
app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] ='sqlite:///database.db'
db = SQLAlchemy(app)
app.secret_key = 'secret key'
class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(100), nullable=False)
    email = db.Column(db.String(100), unique=True)
    password = db.Column(db.String(100))
    def __init__(self, name, email, password):
        self.name = name
        self.email = email
        self.password = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt()).decode('utf-8')
    def check_password(self, password):
```

```python
        return bcrypt.checkpw(password.encode('utf-8'), self.password.encode('utf-8'))
with app.app_context():
    db.create_all()
@app.route('/')
def index():
    return render_template('login.html')
@app.route('/register', methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
        name = request.form['name']
        email = request.form['email']
        password = request.form['password']
        new_user = User(name=name, email=email, password=password)
        db.session.add(new_user)
        db.session.commit()
        return redirect('/login')
    return render_template('register.html')
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']
        user = User.query.filter_by(email=email).first()
        if user and user.check_password(password):
            session['email'] = user.email
            return redirect('/dashboard')
        else:
            return render_template('login.html', error='Invalid user')
    return render_template('login.html')
@app.route('/dashboard')
def dashboard():
    if 'email' in session:
```

```python
    user = User.query.filter_by(email=session['email']).first()
    return render_template('dashboard.html', user=user)
  return render_template('login.html')
@app.route('/logout')
def logout():
  session.pop('email', None)
  return redirect('/login')
@app.route('/result')
def report():
  return render_template('result.html')
@app.route('/about')
def about():
  return render_template('about.html')


@app.route('/analyze', methods=['POST'])
def analyze():
  url_to_scrape = request.form['url']

  # Run your Python code here
  result, img_data = analyze_sentiment(url_to_scrape)

  # Pass the result to the template and render result.html
  return render_template('result.html', result=result, img_data=img_data)

def analyze_sentiment(url_to_scrape):
  html_content = get_html_content(url_to_scrape)

  positive_words, negative_words = load_sentiment_words('list.csv')

  if html_content is not None and positive_words is not None and negative_words is not None:
    positive_count, negative_count, found_words = scrape_and_compare_sentiment(html_content,
positive_words, negative_words)
```

```python
# Construct the result message
result_message = f"<div style='text-align: center;'><b>Positive words count: </b>{positive_count}<br><b>Negative words count:</b> {negative_count}</div>"
n_content=f"{1 if positive_count> negative_count else 0}"


# Construct positive and negative word sections
positive_words_section = f"<b>Positive Words:</b> {', '.join(set([word for word in found_words if word in positive_words]))}"
negative_words_section = f"<b>Negative Words:</b> {', '.join(set([word for word in found_words if word in negative_words]))}"


# Combine the result message, positive words section, and negative words section
result = f"{result_message}{n_content}<br><br><table border='1'>\
        <tr>\
            <th>Positive Words</th>\
            <th>Negative Words</th>\
        </tr>\
        <tr>\
            <td>{', '.join(set([word for word in found_words if word in positive_words]))}</td>\
            <td>{', '.join(set([word for word in found_words if word in negative_words]))}</td>\
        </tr>\
    </table><br>"


# Create a pie chart
labels = ['Positive', 'Negative']
sizes = [positive_count, negative_count]
colors = ['green', 'red']
explode = (0.1, 0)
plt.figure(figsize=(4,4))
plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True, startangle=140)
```

```python
        plt.axis('equal')
        title_text = 'Analysis Pie Chart'
        # Create an underlined version using equal signs (or any character of your choice)
        underlined_title = title_text + '\n' + '----------------' * len(title_text)
        plt.title(underlined_title)


        # Save the plot to a BytesIO object
        img_buf = BytesIO()
        plt.savefig(img_buf, format='png')
        img_buf.seek(0)
        img_data = base64.b64encode(img_buf.read()).decode('utf-8')
        image_path = 'static/sentiment_pie_chart.png'
        plt.savefig(image_path)
        plt.close()


        # Construct the HTML to display the image


        result += f"<br><img src='{image_path}' alt='Sentiment Analysis Pie Chart'>"


        # Add the image HTML to the result
        #result += f"<br>{image_html}"


        return result, img_data
    else:
        return "Error in sentiment analysis"



def get_html_content(url):
    try:
        response = requests.get(url)
        response.raise_for_status()
        return response.text
```

```python
        except requests.exceptions.RequestException as e:
            print(f"Error fetching the webpage: {e}")
            return None


def load_sentiment_words(file_path):
    try:
        df = pd.read_csv(file_path)
        positive_words = df['positive'].tolist()
        negative_words = df['negative'].tolist()
        return positive_words, negative_words
    except Exception as e:
        print(f"Error loading sentiment words from CSV: {e}")
        return None, None


def scrape_and_compare_sentiment(html_content, positive_words, negative_words):
    if html_content is None or positive_words is None or negative_words is None:
        return None, None, None


    soup = BeautifulSoup(html_content, 'html.parser')


    text_content = soup.get_text()


    # Split the text content into words
    words = text_content.split()


    # Create sets to store unique positive and negative words found on the website
    unique_positive_words = set(word.lower() for word in words if word.lower() in positive_words)
    unique_negative_words = set(word.lower() for word in words if word.lower() in negative_words)


    # Count the unique positive and negative words
    positive_count = len(unique_positive_words)
    negative_count = len(unique_negative_words)
```

```python
        # Combine the sets to get all unique found words
        found_words = list(unique_positive_words.union(unique_negative_words))


        return positive_count, negative_count, found_words


class WebsiteAnalyzer:
    def __init__(self):
        self.model_des = None
        self.model_fun = None
        self.model_soc = None
        self.sc_des = None
        self.sc_fun = None
        self.sc_soc = None
        self.load_models()


    def load_models(self):
        # Load the design model
        self.model_des = LogisticRegression()
        dataset_des = pd.read_csv("NEW_WEB_DES.csv")
        X_des = dataset_des.iloc[:, :-1].values
        y_des = dataset_des.iloc[:, -1].values
        self.sc_des = StandardScaler()
        X_des = self.sc_des.fit_transform(X_des)
        self.model_des.fit(X_des, y_des)


        # Load the functionality model
        self.model_fun = LogisticRegression()
        dataset_fun = pd.read_csv("NEW_WEB_FUNT.csv")
        X_fun = dataset_fun.iloc[:, :-1].values
        y_fun = dataset_fun.iloc[:, -1].values
        self.sc_fun = StandardScaler()
```

```python
        X_fun = self.sc_fun.fit_transform(X_fun)
        self.model_fun.fit(X_fun, y_fun)


        # Load the social model
        self.model_soc = LogisticRegression()
        dataset_soc = pd.read_csv("NEW_WEB_SOC.csv")
        X_soc = dataset_soc.iloc[:, :-1].values
        y_soc = dataset_soc.iloc[:, -1].values
        self.sc_soc = StandardScaler()
        X_soc = self.sc_soc.fit_transform(X_soc)
        self.model_soc.fit(X_soc, y_soc)


        self.model_bi = LogisticRegression()
        dataset_bi = pd.read_csv("Final_BI.csv")
        X_bi = dataset_bi.iloc[:, :-1].values
        y_bi = dataset_bi.iloc[:, -1].values
        self.sc_bi = StandardScaler()
        X_bi = self.sc_bi.fit_transform(X_bi)
        self.model_bi.fit(X_bi, y_bi)


    def analyze_design(self, layout, typography, color_scheme, responsive_design):
        input_data = self.sc_des.transform([[layout, typography, color_scheme, responsive_design]])
        prediction = self.model_des.predict(input_data)
        result2 = "1" if prediction[0] == 1 else "0"
        print("Result: ", result2)
        return result2



    def analyze_functionality(self, load_time, personalization, multimedia_support, accessibility):
        input_data = self.sc_fun.transform([[load_time, personalization, multimedia_support,
accessibility]])

        prediction = self.model_fun.predict(input_data)
```

```python
        result3 = "1" if prediction[0] == 1 else "0"
        print("Result: ", result3)
        return result3


    def analyze_social(self, shares, likes, comments, clicks):
        input_data = self.sc_soc.transform([[shares, likes, comments, clicks]])
        prediction = self.model_soc.predict(input_data)
        result4 = "1" if prediction[0] == 1 else "0"
        print("Result: ", result4)
        return result4
        def analyze_bi(self, result2, result3, result4):
        input_data = self.sc_bi.transform([[result2, result3, result4]])
        prediction = self.model_bi.predict(input_data)
        result_5 = "1" if prediction[0] == 1 else "0"
        print("Final result (BI): ", result_5)
        return result_5
analyzer = WebsiteAnalyzer()
@app.route('/analyzeall', methods=['POST'])
def analyzeall():
    layout = float(request.form['layout'])
    typography = float(request.form['typography'])
    color_scheme = float(request.form['color_scheme'])
    responsive_design = float(request.form['responsive_design'])
    load_time = float(request.form['load_time'])
    personalization = float(request.form['personalization'])
    multimedia_support = float(request.form['multimedia_support'])
    accessibility = float(request.form['accessibility'])
    shares = float(request.form['shares'])
    likes = float(request.form['likes'])
    comments = float(request.form['comments'])
    clicks = float(request.form['clicks'])
```

```python
    result2   =   analyzer.analyze_functionality(load_time,   personalization,   multimedia_support,
accessibility)
    result3 = analyzer.analyze_design(layout, typography, color_scheme, responsive_design)
    result4 = analyzer.analyze_social(shares, likes, comments, clicks)
    result5 = analyzer.analyze_bi(result2, result3, result4)
    #result5 = analyzer.analyze_bi(result2, result3, result4)
    #positive_count, negative_count, _ = analyze_sentiment(url_to_scrape)  # Assuming you have
access to these values here
    #n_content = f"<div>{1 if positive_count > negative_count else 0}</div>"


    return               render_template('final.html',result2=result2,result3=result3,result4=result4,
result5=result5)


################################################################################


@app.route('/final.html', methods=['POST'])
def generate_final():
    result2 = request.form.get('result2')
    result3 = request.form.get('result3')
    result4 = request.form.get('result4')
    #result_5 = request.form.get('result5')
    img_data = request.form.get('img_data')
    result5 = analyzer.analyze_bi(result2, result3, result4)

      #n_content = request.form.get('n_content')  # Add this line to get the n_content value
    return            render_template('final.html',            result2=result2,            result3=result3,
result4=result4,result5=result5, img_data=img_data)


if __name__ == '__main__':
    app.run(debug=True)
```

6. TESTING

1- Unit testing

Usually, it's the job of a developer to do it. It focuses on the unit-level and assists in validating the internal implementation of a feature in the project.

2- Integration testing

As the name suggests, the testers perform it to check whether the multiple components of a product work as expected or not.

3- System testing

This type of testing ensures the stability of the overall product. It usually happens after all the proposed features get implemented by the developers.

4- Sanity testing

It is nothing but the execution of a smaller chunk of test cases taken from the main test plan for validating the basic functionality of the product under development.

5- Smoke testing

The alternate name for this type of testing is Build verification testing. It has most non-exhaustive test cases which intend to exercise the core functional areas of the product under test. It helps in establishing the stability of a build and decides whether to continue the test execution or not.

6- Interface testing

It is one of the testing types executed by a tester on the application under test. It probes the errors in the communication channel (Interface) that exist between two distinct Software systems.

7- Regression testing

It is one of the strongest pillars of all testing types. It primarily tests the fact the new development is not breaking the old functionality. Such a kind of testing requires enormous efforts, and hence, mostly the test automation takes care of it.

8- Beta testing

Bets testing is a type of acceptance testing which intends to bring the customer perspective into validation. It means that the end-user (actual user) gets the opportunity to explore the usability, functionality, compatibility, and reliability of the product.

9- Performance testing

The performance testing is a type of testing which runs a Software application under high load and evaluates its behavior. Since every customer wants a product that can respond without any error during the peak load. Hence, the response time, throughput, reliability, and scalability of the application become crucial.

7.TEST CASES

7.1 Test Senario1: - Check the Login Functionality

Input: Collecting data from users for login.

Output: The system must login and display the index page.

Result: working as per the requirement.

Test case1: Valid Id and valid password.

Test case2: Valid Id and invalid password.

Test case3: Invalid Id and valid password.

Test case4: Black fields.

7.2 Test Senario2: - Check the Registration Functionality

Valid Registration:

Input: Valid email, password, and other required information.

Output: User is successfully registered and redirected to a Dashboard page.

Test case1: Check the email is existing.

Test case2: Valid Id and invalid password.

Test case3: Email that is already registered.

7.3 Test Senario3: - Check the news URL bar Functionality

Input: Enter a valid URL for a news article.

Output: The page navigates to the correct article.

Test case1:Enter an invalid URL.

Test case2:Enter a URL with parameters (e.g., ?id=123).

Test case3:Enter a URL with special characters (e.g., %20 for space).

Test case4: Navigate to a news article and then preview the news article.

7.4 Test Senario4: - Test Cases for Overall Analyze Button
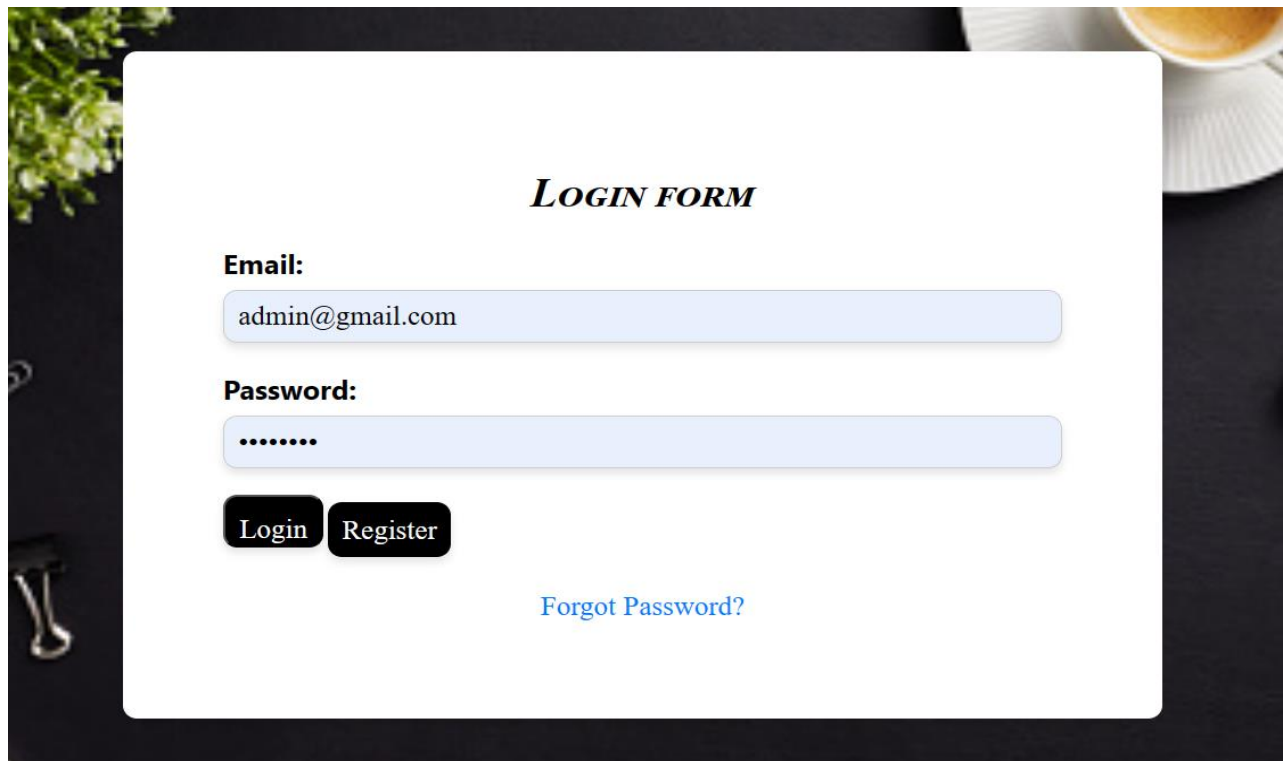
Input: Click the "Overall Analyze Button".

Expected outcome: The page is redirected to the result page and view overall results.

Test case1: Check the page is redirected to the result page.

Test case2: View the overall result.

8. RESULT

Figure-8.1: Login Page

Figure-8.2: Registration Page
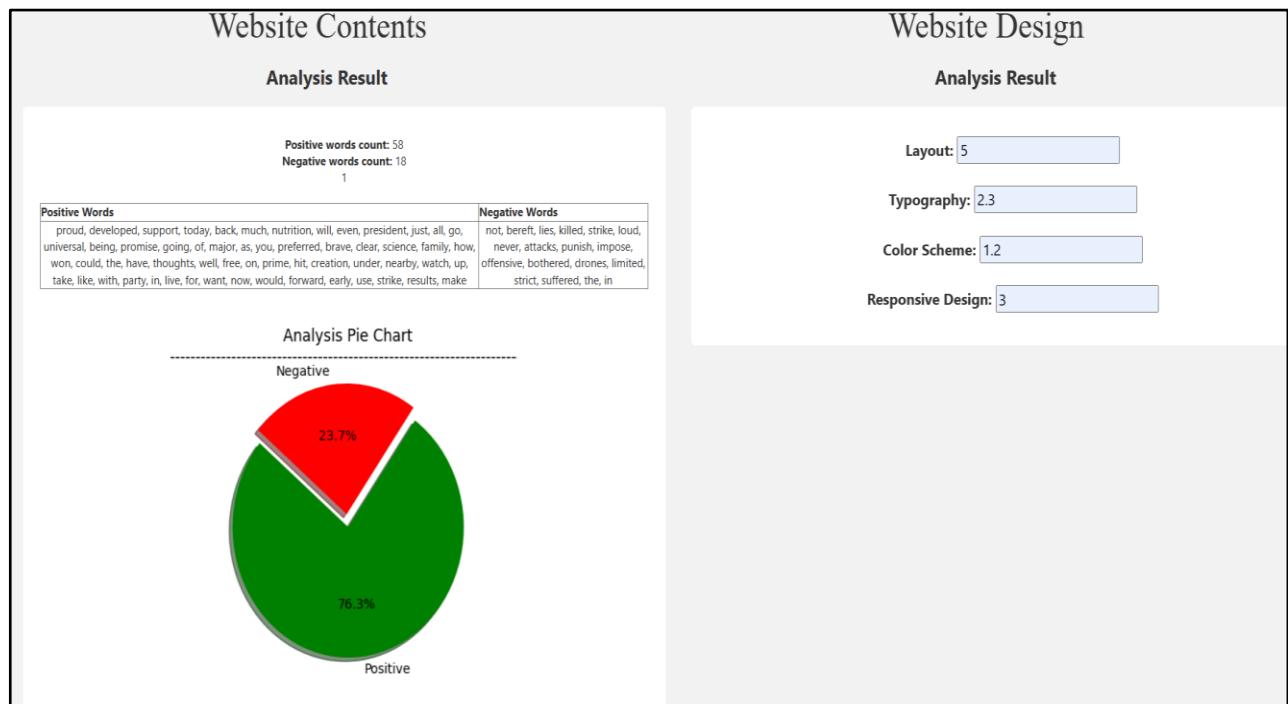
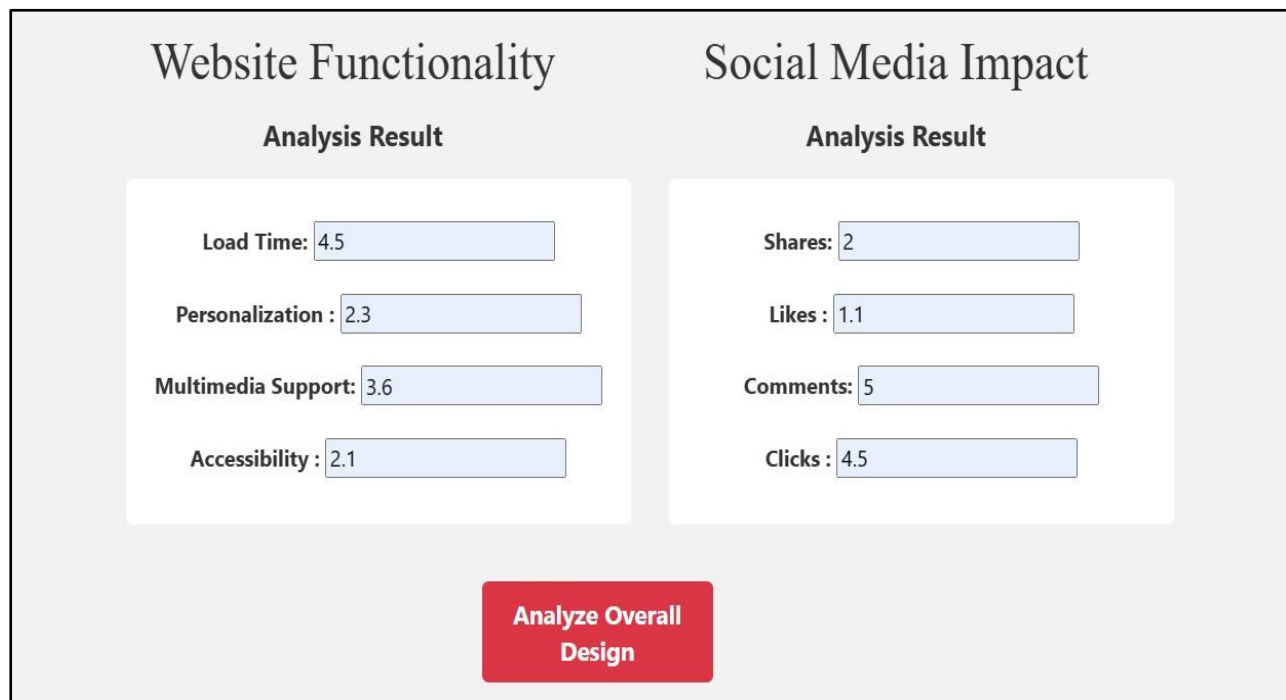**Figure-8.3: Home Page**

Figure-8.4: Evaluation page 1

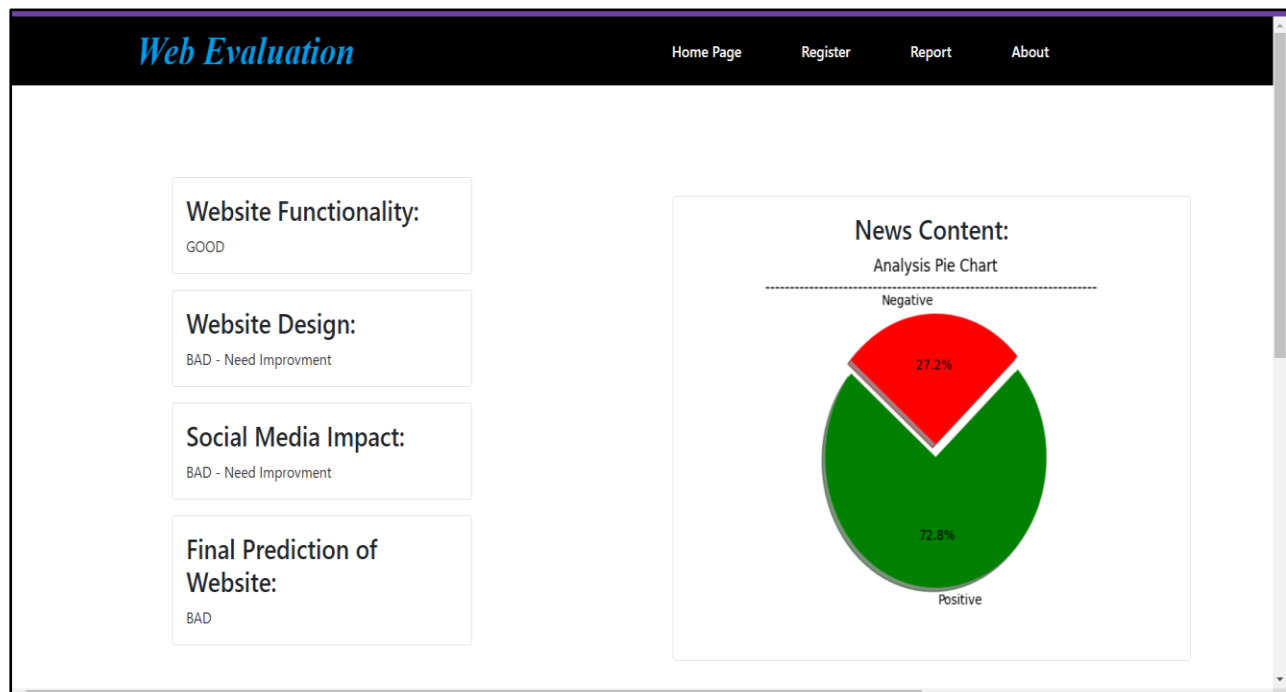**Figure-8.5: Evaluation Page 2**

Figure-8.6: Result Page 1

Overall Result:

Figure-8.7: Result Page 2
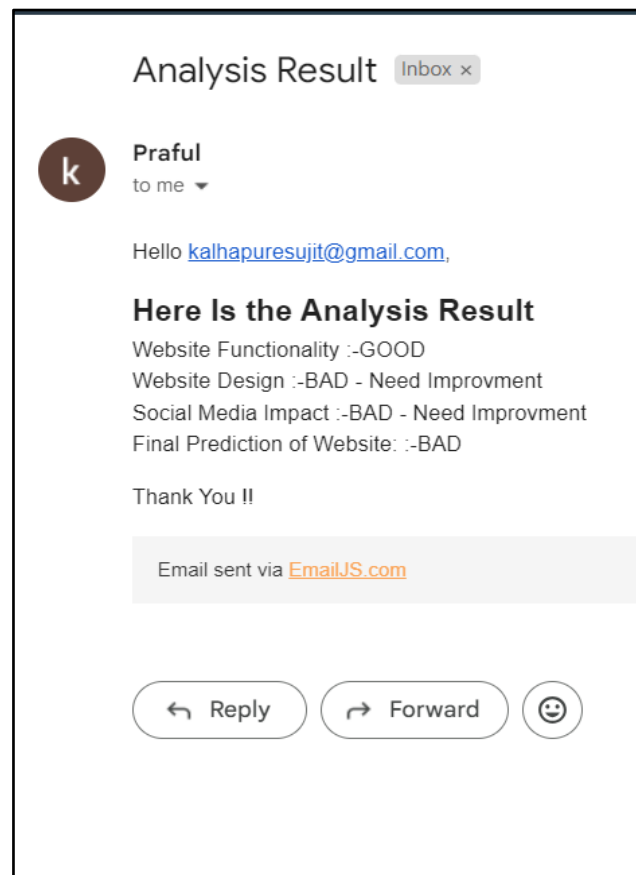
Figure-8.8: Sending Result on Email

Email :-

Figure-8.9: Email Result

Figure-8.10: About Page 1

Figure-8.10: About Page 2

9. OTHER SPECIFICATION

9.1 Applications

Web application evaluation using opinion mining has a wide range of applications across various domains. Opinion mining, also known as sentiment analysis, involves analyzing user-generated content to determine sentiments, opinions, and attitudes expressed towards products, services, or topics. Here are some prominent applications of web application evaluation using opinion mining:

1. Product and Service Reviews:

   Analyzing customer reviews and feedback for products or services offered through web applications. Understanding customer sentiments helps businesses make informed decisions about product improvements, marketing strategies, and customer satisfaction.

2. Brand Monitoring and Reputation Management:

   Tracking online mentions and sentiments associated with a brand or organization. Evaluating opinions helps in managing brand reputation and addressing potential issues promptly.

3. Social Media Analysis:

   Analyzing social media content to gauge public opinion about a brand, event, or topic. It's useful for understanding trends, public perception, and the effectiveness of marketing campaigns.

4. E-Commerce Platforms:

   Evaluating customer reviews and opinions on e-commerce platforms. This helps in recommending products, improving product descriptions, and enhancing the overall shopping experience.

5. Restaurant and Food Service Feedback:

   Analyzing reviews and opinions about restaurants, cafes, and food delivery services. This can aid in menu improvements, service enhancements, and customer satisfaction.

6. Travel and Hospitality Industry:

   Evaluating traveler reviews and opinions about hotels, airlines, travel destinations, and booking platforms. Insights gained can lead to improved services and offerings.

7. Healthcare and Telemedicine:

   Analyzing patient reviews and opinions about healthcare providers, medical facilities, and telemedicine services. This can help healthcare providers enhance patient experiences and service quality.

8. Educational Institutions:

   Evaluating feedback and opinions of students and parents regarding educational institutions, courses, teaching quality, and campus facilities. Institutions can use this data for improvement and marketing efforts.

9. Political Analysis and Public Opinion:

   Analyzing public sentiments regarding political parties, policies, candidates, and government decisions. This helps in understanding public opinion trends and political sentiments.

10. Media and News Outlets:

    Analyzing public opinions and reactions to news articles, TV shows, or movies. Media outlets can use this information to tailor content and understand audience preferences.

11. Financial Services:

    Analyzing opinions on financial products, investment options, banking services, and financial institutions. This assists in understanding customer sentiments and tailoring services to meet customer needs.

12. Gaming Industry:

    Analyzing opinions and feedback on gaming platforms, video games, gaming peripherals, and the gaming community. This helps in game development and marketing strategies.

9.2 Advantages

- User can easily share his view about the website.

- People can easily decide whether the website is genuine or not

- This application is more useful for those who do online transactions.

- This system helps the user to find the website which provides good service and delivers the product in time.

- Since the system ranks the website based on the weight age of the keywords in database, so the result is appropriate.

9.3 Disadvantages

- System will match the opinion with those keywords which are in database rest of the words are ignored by the system.

9.4 Future Work

This system is developed based upon the fundamentals requirement so may there's some limitation in development. But as per global requirement it's going to be enhanced to the following points in the next release.

Predicting the future developments in web application evaluation using opinion mining involves anticipating potential advancements based on current trends and technological progress. As of my last knowledge update in September 2021, here are some potential future directions and areas of improvement for web application evaluation using opinion mining:

1. Improved Sentiment Analysis Techniques:

   Enhancing sentiment analysis algorithms to detect more nuanced and context-aware sentiments. This could involve considering cultural contexts, humor, sarcasm, and regional language variations.

2. Multimodal Opinion Mining:

   Integrating other forms of data, such as images, audio, and video, into opinion mining to analyze a broader range of content and opinions. Combining text and non-textual data can provide a richer understanding of user sentiments.

3. Aspect-Based Opinion Mining:

   Refining opinion mining to analyze opinions at a more granular level, focusing on specific aspects or features of a product or service. This allows for a more detailed understanding of user preferences.

4. Real-Time Opinion Analysis:

   Enabling real-time analysis of opinions and sentiments as they are expressed on the web. This would be valuable for quickly responding to emerging trends and issues.

5. Adaptive Opinion Mining Models:

   Developing opinion mining models that can adapt and evolve over time based on changing language usage, user behavior, or societal shifts. This would ensure the models stay effective and accurate in dynamic environments.

6. Cross-Lingual Opinion Mining:

   Enhancing opinion mining to analyze and interpret opinions in multiple languages. This is particularly important for global applications and businesses that operate in multilingual settings.

7. Explainable AI in Opinion Mining:

   Integrating explainable AI techniques to provide clear and understandable reasons behind sentiment analysis results, improving trust and user acceptance of the evaluation process.

8. Incorporating Domain Knowledge:

    Leveraging domain-specific knowledge and ontologies to improve opinion mining accuracy and relevance for particular industries or areas of interest.

9. Privacy-Preserving Opinion Mining:

    Addressing privacy concerns by developing techniques that allow for opinion mining without compromising user privacy or data confidentiality.

10. Integration with Recommendation Systems:

    Integrating opinion mining with recommendation systems to provide users with personalized recommendations based on their sentiments and opinions.

11. Long-Term Impact Analysis:

    Investigating the long-term impact of opinions and sentiments on user behavior, product adoption, and brand reputation.

Appendix A: Details of paper publication.

8th International Conference on Computing, Communication, Control and Automation : Submission (569) has been created.

Inbox ×

Microsoft CMT <email@msr-cmt.org>
to me ▾

Mon, Apr 22, 11:26 PM (21 hours ago)

Hello,

The following submission has been created.

Track Name: Cognitive Computing and Machine Learning

Paper ID: 569

Paper Title: News Web Application Evaluation using Machine Learning

Abstract:
In our contemporary digital landscape, news consumption has transitioned largely to online platforms, shaping public opinion and influencing societal discourse. However, this shift has also led to the proliferation of misinformation and fake news, which can have profound consequences on public perception and decision-making processes. Moreover, the rise of social media has accelerated the dissemination of news, amplifying the impact of false information.
In response to these challenges, this project focuses on developing a robust framework for evaluating news websites using advanced techniques in Artificial Intelligence (AI), Natural Language Processing (NLP), and Machine Learning (ML). The primary objective is to empower users with the ability to discern between credible and unreliable sources of information by performing binary classification of news articles.
Through the utilization of sophisticated algorithms and datasets, our system aims to analyze the content of online news articles and assess their authenticity. By leveraging AI and ML models, users will be equipped with tools to identify potentially misleading or fabricated news stories, thereby promoting critical thinking and informed decision-making.
Key features of the proposed system include the classification of news articles as either authentic or fake, as well as an evaluation of the credibility of the websites publishing the news. By harnessing the power of AI-driven analysis, this project endeavors to mitigate the spread of misinformation and enhance trust in online news sources.

Created on: Mon, 22 Apr 2024 17:56:16 GMT

Created on: Mon, 22 Apr 2024 17:56:16 GMT

Last Modified: Mon, 22 Apr 2024 17:56:16 GMT

Authors:
 - sunil6kr@gmail.com (Primary)
 - kalhapuresujit@gmail.com
 - rutikachougule2002@gmail.com
 - vaishnavidobe112@gmail.com
 - varsha8856919240@gmail.com

Secondary Subject Areas: Not Entered
Submission Files: News Web Application Evaluation using Opinion Mining_version3.docx (998 Kb, Mon, 22 Apr 2024 17:56:03 GMT)
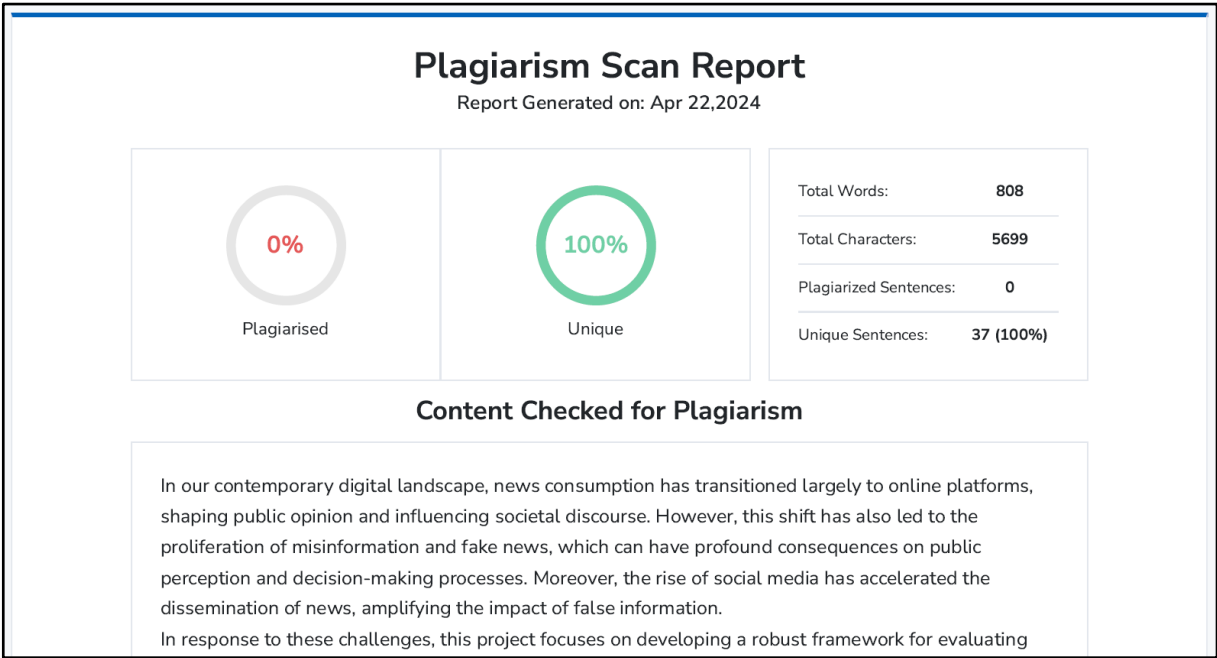
Submission Questions Response: Not Entered

Thanks,
CMT team.

Appendix B: Plagiarism Report of project report.

Plagiarism Scan Report

Report Generated on: Apr 22,2024

0%	**100%**	Total Words: **808**
Plagiarised	Unique	Total Characters: **5699**
		Plagiarized Sentences: **0**
		Unique Sentences: **37 (100%)**

Content Checked for Plagiarism

In our contemporary digital landscape, news consumption has transitioned largely to online platforms, shaping public opinion and influencing societal discourse. However, this shift has also led to the proliferation of misinformation and fake news, which can have profound consequences on public perception and decision-making processes. Moreover, the rise of social media has accelerated the dissemination of news, amplifying the impact of false information.

In response to these challenges, this project focuses on developing a robust framework for evaluating

10.CONCLUSION

In the 21st century, most tasks are done online. Newspapers that were earlier preferred as hard copies are now being substituted by applications like Facebook, Twitter, and news articles to be read online. WhatsApp's forwards are also a major source.

The growing problem of news only makes things more complicated and tries to change or hamper the opinion and attitude of people towards use of digital technology. When a person is deceived by the real news two possible things happen- People start believing that their perceptions about a particular topic are true as assumed. Thus, to curb the phenomenon, we have developed our News Detection system that takes input from the user and classify it to be true or false. To implement this, various NLP and Machine Learning Techniques must be used.

The model is trained using an appropriate dataset and performance evaluation is also done using various performance measures. The best model, i.e., the model with highest accuracy is used to classify the news headlines or articles. As evident above for static search, our best model came out to be Logistic Regression with an accuracy of 65%. Hence, we then used grid search parameter optimization to increase the performance of logistic regression which then gave us the accuracy of 75%. Hence, we can say that if a user feeds a particular news article or its headline in our model, there are 75% chances that it will be classified to its true nature.

The user can check the news article or keywords online; he can also check the authenticity of the website. The accuracy for dynamic system is 93% and it increases with every iteration. We intend to build our own dataset which will be kept up to date according to the latest news. All the live news and latest data will be kept in a database using Web Crawler and online database.

Our system gives the customer a better understanding about websites. It prevents the customer from being a victim of fraud. It will give them a detailed survey about the service or customer feedback regarding those websites. Our system could be a critic for the websites hence driving them towards improvement on their drawbacks. Based on data availability our system can be expanded to different sectors. It is also open to customers' views based on their frequent experience with those websites.

REFERENCES

[1] Priyanka Rathord Dr. Anurag Jain Chetan Agrawal, 'A Comprehensive Review On Online News Popularity Prediction Using   Machine Learning Approach', 2019.

[2] P. Keerthana, B. Meghana, P. Akshaya, 'Website Evaluation Using Opinion Mining', 2021.

[3] Margarita Rodríguez-Ibanez , Antonio Casanez-Ventura , F´Elix Castejon-Mateos, Pedro-Manuel Cuenca-Jim´Enez 'A Review On Sentiment Analysis From Social Media Platforms', 2023

[4] http://www.cs.virginia.edu/~up3f/cs4750/supplement/db-setup-xampp.html

[5] Barbieri, Camacho-Collados, Espinosa Anke, & Neves, L. Tweet Eval: Unified Benchmark And Comparative Evaluation For Tweet Classification. In Findings Of The Association For Computational Linguistics: EMNLP 2020 Pp. 1644–1650

[6] Elena Hensinger, Ilias Flaounas, Nello Cristianini,"Modelling And Predicting News Popularity" Springer,Pattern Anal Applic, 2013, Pp. 623–635.

[7] Arapakis, Ioannis, B. Barla Cambazoglu, And Mounia Lalmas. "On The Feasibility Of Predicting News Popularity At Cold Start." International Conference On Social Informatics. Springer International Publishing, 2014.

[8] Al-Mutairi, Hanadi Muqbil, And Mohammad Badruddin Khan. "Predicting The Popularity Of Trending Arabic Wikipedia Articles Based On External Stimulants Using Data/Text Mining Techniques."Cloud Computing (ICCC), 2015 International Conference On. IEEE, 2015.

[9] Tatar Alexandru, Et Al. "Predicting The Popularity Of Online Articles Based On User Comments." Proceedings Of The International Conference On Web Intelligence, Mining And Semantics. ACM, 2011.

[10] Castillo, Carlos, Et Al. "Characterizing The Life Cycle Of Online News Stories Using Social Media Reactions." Proceedings Of The 17th ACM Conference On Computer Supported Cooperative Work & Social Computing. ACM, 2014.