A

Mini Project Report

On

“QuickReads: Your Personalized Content Guide Using Hybrid Filtering”

Submitted in partial fulfillment of the requirements for the

Degree

Third Year Engineering – Computer Science Engineering (Data Science)

By

SONAL SONARGHARE	**21107033**
JANHAVI KASAR	**21107030**
SANSKRUTI CHAVAN	**21107047**
HARSH SHELKE	**21107022**

Under the guidance of

PROF.SARALA MARY

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING (DATA SCIENCE)
A.P. SHAH INSTITUTE OF TECHNOLOGY
G.B. Road, Kasarvadavali, Thane (W)-400615
UNIVERSITY OF MUMBAI

Academic year: 2023-24

CERTIFICATE

This to certify that the Mini Project report on **"QuickReads: Your Personalized Content Guide using Hybrid Filtering"** has been submitted by Sonal Sonarghare (21107033), Janhavi Kasar (21107030), Sanskruti Chavan (21107047) and Harsh Shelke (21107022) who are bonafide students of A. P. Shah Institute of Technology, Thane as a partial fulfillment of the requirement for the degree in **Computer Science Engineering (Data Science)**, during the academic year **2023-2024** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Sarala Mary
Guide

Ms. Anagha Aher
HOD, CSE (Data Science)

Dr. Uttam D. Kolekar
Principal

External Examiner:
1.

Internal Examiner:
1.

Place: A. P. Shah Institute of Technology, Thane
Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Ms. Sarala Mary** Expressing gratitude towards our HoD, **Ms. Anagha Aher**, and the Department of Computer Science Engineering (Data Science) for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our project coordinator **Ms. Poonam Pangarkar** who gave us his/her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

ABSTRACT

This report presents the development and evaluation of QuickReads, an innovative hybrid recommendation system tailored to deliver personalized article recommendations. QuickReads employs a synergistic blend of collaborative filtering and content-based filtering techniques, augmented by sophisticated web scraping methodologies to curate a diverse array of pertinent articles from a multitude of sources. The system's robust architecture encompasses meticulous data collection procedures, comprehensive data preprocessing steps, and advanced feature extraction methodologies, all contributing to its robust recommendation engine The system includes features designed to enrich the user experience include summary generation, dictionary lookup, and read-aloud functionality, which collectively enhance usability and engagement. The performance of QuickReads is rigorously evaluated using a diverse set of metrics, including precision, recall, F1-score, and user satisfaction surveys. These evaluations underscore the system's efficacy in providing tailored, compelling content recommendations that resonate with users' preferences and interests.

Keyword*s—QuickReads, Hybrid recommendation system, Collaborative filtering, Content-based filtering, Web scraping methodologies, Data collection procedures, Data preprocessing techniques, Feature extraction, User experience, Summary generation, Dictionary lookup, Read-aloud functionality, Performance evaluation, User satisfaction surveys.*

Chapter 1

Introduction

In an age where information is abundant yet time remains scarce, navigating through the vast sea of online articles to find content that aligns with individual interests can be a daunting task. Recognizing this challenge, we introduce “QuickReads”, an innovative article recommendation system designed to streamline the discovery process for users across diverse domains such as technology, health, finance, and more.

QuickReads harnesses the power of web scraping to aggregate articles from a multitude of reputable sources spanning various domains. By systematically curating content from these sources, QuickReads ensures a comprehensive and dynamic repository of articles that cater to a wide array of interests and preferences.

In this there lies a user-centric approach that prioritizes customization and personalization. Upon entering the platform, users are invited to explore content within their domain of choice. Whether delving into the latest advancements in technology, exploring wellness tips in health, discovering culinary delights in food, or staying updated on financial trends, QuickReads empowers users to navigate their preferred domains with ease and efficiency.

However, QuickReads goes beyond mere domain-based recommendations [2]. Leveraging sophisticated hybrid models [3], the system continuously learns from user behavior and preferences to refine and enhance the recommendation process. As users engage with articles, QuickReads dynamically adapts, presenting tailored recommendations that align with their evolving interests and consumption patterns.

This report delves into the core mechanisms of QuickReads, exploring its inception, architectural framework, and the methodologies that drive its recommendation engine. By blending technology with a user-centric approach, QuickReads aims to provide users with a carefully curated selection of articles that enhance their digital experience.

1.1 Purpose:

The purpose of QuickReads is multifaceted, aiming to revolutionize the way users discover and engage with online articles. In today's fast-paced digital landscape, the sheer volume of information available can be overwhelming, often leaving users feeling lost in a sea of content. QuickReads ensures a comprehensive repository of articles.

Through a fusion of web scraping, hybrid modeling, user engagement and feedback, and a suite of features including read aloud option [1], summaries, and integrated dictionary, QuickReads continually enhances its recommendation algorithms, thereby refining recommendations to match users' ever-evolving interests and preferences.

The core purpose of QuickReads is to empower users to effortlessly discover and engage with relevant articles, enriching their digital experience and knowledge exploration journey. By offering personalized recommendations curated to individual preferences, QuickReads aims to alleviate the overwhelming nature of information overload. With its intuitive interface and refined recommendation algorithms, QuickReads endeavors to make the process of article discovery seamless, enjoyable, and rewarding for users across various domains and interests.

1.2 Problem Statement:

The problem statement for QuickReads revolves around users' struggle to efficiently discover and engage with relevant online content across various domains. Existing systems often fail to provide personalized recommendations, leading to information overload and limited exploration. QuickReads aims to address these challenges by offering a curated selection of articles tailored to individual interests, enhancing users' content discovery and browsing experiences.

1. **Lack of Personalization:**

   Existing content recommendation systems often lack personalization, providing generic suggestions that may not align with individual user preferences. QuickReads addresses this by offering tailored recommendations based on user interests, enhancing the content discovery experience and encouraging deeper engagement.

2. **Time Constraints:**

   With busy schedules and limited time, users require efficient and streamlined methods for discovering and consuming online articles that cater to their interests.

3. **Exploration Limitations:**

   Users may feel constrained by the limited options for exploring new topics and domains, hindering their ability to broaden their knowledge and interests.

4. **No Article Summarization:**

   Without article summarization functionality, users may face challenges in quickly grasping the key points of lengthy articles. This limitation hinders efficient information consumption and may deter users from engaging with content due to time constraints or information overload.

5. **Limited Dictionary Integration:**

   The current system may lack integration with a dictionary feature, restricting users from easily accessing definitions or meaning.

6. **Absence of Read-Aloud Functionality:**

   The absence of a read-aloud feature for summarized content restricts accessibility and user experience, especially for those with visual impairments or engaged in multitasking.

7. **Quality and Trustworthiness:**

   Users seek high-quality and trustworthy content from reputable sources, necessitating mechanisms to filter out irrelevant or unreliable articles.

1.3 Objectives:

In a bid to revolutionize online content consumption, QuickReads sets out with clear objectives. It endeavors to redefine content discovery by curating articles across diverse domains and offering personalized recommendations. Leveraging a hybrid recommendation model that combines the strengths of content-based filtering and collaborative filtering techniques, QuickReads aims to optimize user engagement and satisfaction. Through continuous evaluation and refinement, QuickReads aspires to uphold excellence in content discovery, remaining adaptable to the ever-evolving needs of its users.

1. **Domain-Based Recommendations:**

   Provide users with personalized article recommendations within their chosen domain of interest, allowing for focused exploration and discovery.

2. **Enhance Content Discovery:**

   Develop a system that enables users to discover relevant articles across various domains such as technology, health, food, finance, etc., based on their interests and preferences.

3. **Hybrid Recommendation Model:**

   Develop a hybrid recommendation model that combines content-based filtering and collaborative filtering techniques to offer a balanced mix of personalized and diverse recommendations [3].

4. **User Engagement and Feedback:**

   Incorporate features that encourage user engagement, such as ratings, likes, comments, and sharing options, to gather feedback and improve the quality of recommendations.

5. **Implement Read Aloud Functionality:**

   Integrate a read-aloud feature that allows users to listen to summarized content, enhancing accessibility for visually impaired individuals and users engaged in multitasking [1].

6. **Integrate Dictionary Feature:**

   Incorporate a dictionary feature into the platform, enabling users to easily access definitions or meanings of unfamiliar terms encountered in articles.

7. **Article Summarization:** Utilize Natural Language Processing (NLP) techniques to generate summaries of articles, providing users with concise overviews to quickly grasp key points and improve information consumption efficiency.

1.4 Scope:

The scope of the QuickReads project encompasses the development of a comprehensive article recommendation system designed to cater to users' diverse interests and preferences. QuickReads aims to revolutionize the process of discovering relevant articles by leveraging advanced technologies and user-centric design principles.

1. **Content Discovery:**

   QuickReads facilitates the relevant articles, enabling users to discover content aligned with their interests and preferences. It identifies different topics and helps users stay informed about emerging issues, ensuring accurate content discovery tailored to individual needs.

2. **User Interaction:**

   QuickReads fosters engagement across diverse user demographics by offering customized solutions and curated content. It promotes exploration, discovery, and meaningful interactions with diverse topics.

3. **Innovation Hub:**

   QuickReads serves as a hub for research and innovation in content recommendation systems, continuously advancing predictive technologies to enhance user experience. By fostering collaboration and knowledge sharing, QuickReads contributes to the evolution of personalized content discovery in the digital realm.

4. **Educational Outreach:**

   By incorporating QuickReads into educational endeavors centered around the nation's historical narrative, technology, science and other knowledge topics. QuickReads promotes digital literacy and responsible information consumption.

5. **Rapid Updates:**

   QuickReads ensures users can stay quickly updated on the latest developments by providing timely access to newly published articles and breaking news. Through real-time updates and curated feeds, QuickReads keeps users informed about relevant information in their areas of interest, enabling them to stay ahead of trends and developments in their fields.

Chapter 2

Literature Review

The literature review delves into recommendation systems' evolution, vital for personalized content delivery amidst the surging digital content. It encompasses collaborative filtering, content-based filtering, hybrid methods, and advanced techniques like deep learning. This analysis aims to scrutinize key concepts, challenges, and trends while pinpointing areas for future research and enhancements, shaping the understanding and development of recommendation systems.

Essien, Uwah, and Ododo (2021) [1] developed a web-based text-to-speech system for visually impaired users. Using JavaScript, Natural Language Processing, and Digital Signal Processing, the system enables real-time conversion of text to speech, enhancing accessibility across devices. It emphasizes compatibility with speech synthesis libraries and advances in accessibility technology for education and digital environments.

Singhal (2021) [2] presents a research paper on recommendation systems, emphasizing context-awareness and discussing collaborative filtering and content-based approaches. The study reviews existing recommendation methods, analyzes their real-world operation, and highlights the importance of context-awareness. The result underscores a comprehensive understanding of recommendation systems and their pivotal role in information retrieval.

Ye, Tu, and Liang (2019) [3] developed a hybrid recommendation system to address limitations in traditional content-based and collaborative filtering methods for recommending research articles. While content-based filtering relies on item characteristics and collaborative filtering on user behavior, their hybrid system combines both, leveraging the strengths of each approach. The study demonstrates the superiority of their hybrid system over traditional methods, aligning with prior research advocating for hybrid recommendation systems in various domains.

Sen Zhang (2023) [4] delved into enhancing restaurant recommendations using TF-IDF vectorization, merging content-based and collaborative filtering approaches with deep learning. By leveraging TF-IDF to extract textual features from reviews, it recommends restaurants based on similarities in reviews, improving recommendation accuracy and user satisfaction.

The study conducted by Rusdiansyah, Rusdiansyah (2024) [5] explored web program testing using Selenium Python, emphasizing automation for login, transactions, and customer data on website. It discusses best practices, benefits in efficiency, security, user experience, and automated testing, contributing insights for web developers and researchers.

Chapter 3

Proposed System

The proposed system for QuickReads aims to transform the way users discover and engage with online content. By aggregating articles from diverse domains and employing a hybrid recommendation model, QuickReads offers personalized suggestions tailored to user preferences. With its visionary approach, it seeks to redefine content discovery in the digital era, empowering users to explore and interact with articles that resonate with their interests.

1. **Article Aggregation:**

   Quickreads employs powerful web scraping technique Selenium to gather articles from a diverse range of reputable websites across various domains such as technology, health, food, finance, and more. This process ensures a comprehensive collection of articles available for users to explore [5].

2. **User Registration and Preferences:**

   Users will have the option to register for an account on Quickreads, enabling them to set preferences and customize their browsing experience. During registration, users can specify their areas of interest by selecting domains such as technology, health, food, finance, etc.

3. **Domain-Based Recommendations:**

   Once registered, Quickreads will provide personalized article recommendations based on the domains selected by the user. The system will analyze user interaction within each domain to deliver relevant and engaging recommendations tailored to individual interests.

4. **Hybrid Recommendation Model:**

   Quickreads will leverage a hybrid recommendation model that combines content-based filtering and collaborative filtering techniques. Content-based filtering analyzes article content, keywords, and metadata to suggest relevant articles within the selected domains. Collaborative filtering considers user interactions, such as likes, shares, and comments, to recommend articles that align with similar users' preferences [3].

5. **User Interface Design:**

   Quickreads will feature an intuitive and user-friendly interface designed to enhance the browsing experience. The interface will include easy navigation options, visually appealing layouts, and interactive elements to facilitate seamless exploration of articles across different domains.

By implementing these features, Quickreads aims to provide users with a comprehensive and user-centric platform for discovering, exploring, and engaging with online content across diverse domains.

3.1 Features and Functionality: -

QuickReads revolutionizes content consumption with its innovative features. From concise article summaries to cross-language recommendations, QuickReads offers a personalized reading experience tailored to diverse user needs. Let's explore its key features, designed to empower users in navigating the digital landscape effortlessly.

1. **Article Summaries:**

   Leveraging libraries like NLTK (Natural Language Toolkit) for natural language processing, the system generates concise summaries of articles. These summaries offer users a quick overview of the content, allowing them to grasp key points without having to read the entire article.

2. **Read-Aloud Option:**

   The system includes a read-aloud feature that enables users to listen the summaries of the articles being read aloud. This feature enhances accessibility for users with visual impairments or those who prefer auditory learning styles [1].

3. **Dictionary:**

   QuickReads includes a dictionary feature, providing users with instant access to definitions and explanations of unfamiliar terms encountered while reading. This functionality enhances users' understanding of content and fosters continuous learning.

4. **Recommendations Based on Trends and User Behaviour:**

   The system continuously analyses current trends, user behaviour patterns, and emerging topics of interest. Based on this analysis, it provides personalized recommendations for articles, books, and news articles that align with users' preferences and interests.

QuickReads offers personalized recommendations for articles, books, and news articles, along with concise article summaries facilitated by natural language processing. The platform enhances accessibility through a Read-Aloud option and a Dictionary, catering to diverse user needs. With its user-centric features and innovative approach, QuickReads revolutionizes content discovery, empowering users to navigate the digital landscape effortlessly.

Chapter 4

Requirements Analysis

For the requirement analysis of the QuickReads project, we need to identify and document the functional and non-functional requirements that the system must meet to fulfill its objectives effectively. Here's a breakdown of the requirement analysis for QuickReads:

A. Functional Requirements:

1. **Domain-Based Recommendations:**

   The system should provide personalized article recommendations within users' chosen domains of interest. Users should be able to select and explore specific domains for focused content discovery.

2. **Enhance Content Discovery:**

   The system should enable users to discover relevant articles across various domains based on their interests and preferences. Users should have options to explore content in categories such as technology, health, food, finance, etc.

3. **Hybrid Recommendation Model:**

   Develop a hybrid recommendation model that combines content-based filtering and collaborative filtering techniques. The system should provide a balanced mix of personalized recommendations and diverse content suggestions [3].

4. **User Engagement and Feedback:**

   Incorporate features like ratings, likes, comments, and sharing options to encourage user engagement. The system should collect user feedback to improve the quality of recommendations over time.

5. **Implement Read Aloud Functionality:**

   Integrate a read-aloud feature that allows users to listen to summarized content. The feature should enhance accessibility for visually impaired users and those engaged in multitasking [1].

6. **Integrate Dictionary Feature:**

   Incorporate a dictionary feature into the platform for users to access definitions of unfamiliar terms in articles. Users should be able to easily look up meanings or explanations while reading.

7. **Article Summarization:**

   Utilize Natural Language Processing (NLP) techniques to generate summaries of articles. The system should provide concise overviews to help users quickly grasp key points and improve information consumption efficiency.

B. Non-Functional Requirements:

1. **Performance:**

   The system should load articles and recommendations quickly to ensure a seamless user experience. It should handle concurrent user requests efficiently.

2. **Scalability:**

   The system should be scalable to handle a growing user base and increasing content volume.

3. **Security:**

   User data should be stored securely, and the platform should have measures to prevent unauthorized access and data breaches.

4. **Usability:**

   The user interface should be intuitive and easy to navigate, promoting user engagement and satisfaction. Accessibility features should be implemented for users with disabilities.

Chapter 5

Project Design

5.1 Use Case diagram:

Use case diagram of QuickReads (Fig 5.1) captures the main functionalities and interactions between users and the system, including personalized recommendations, content discovery, user engagement, and accessibility features like read-aloud and dictionary integration.

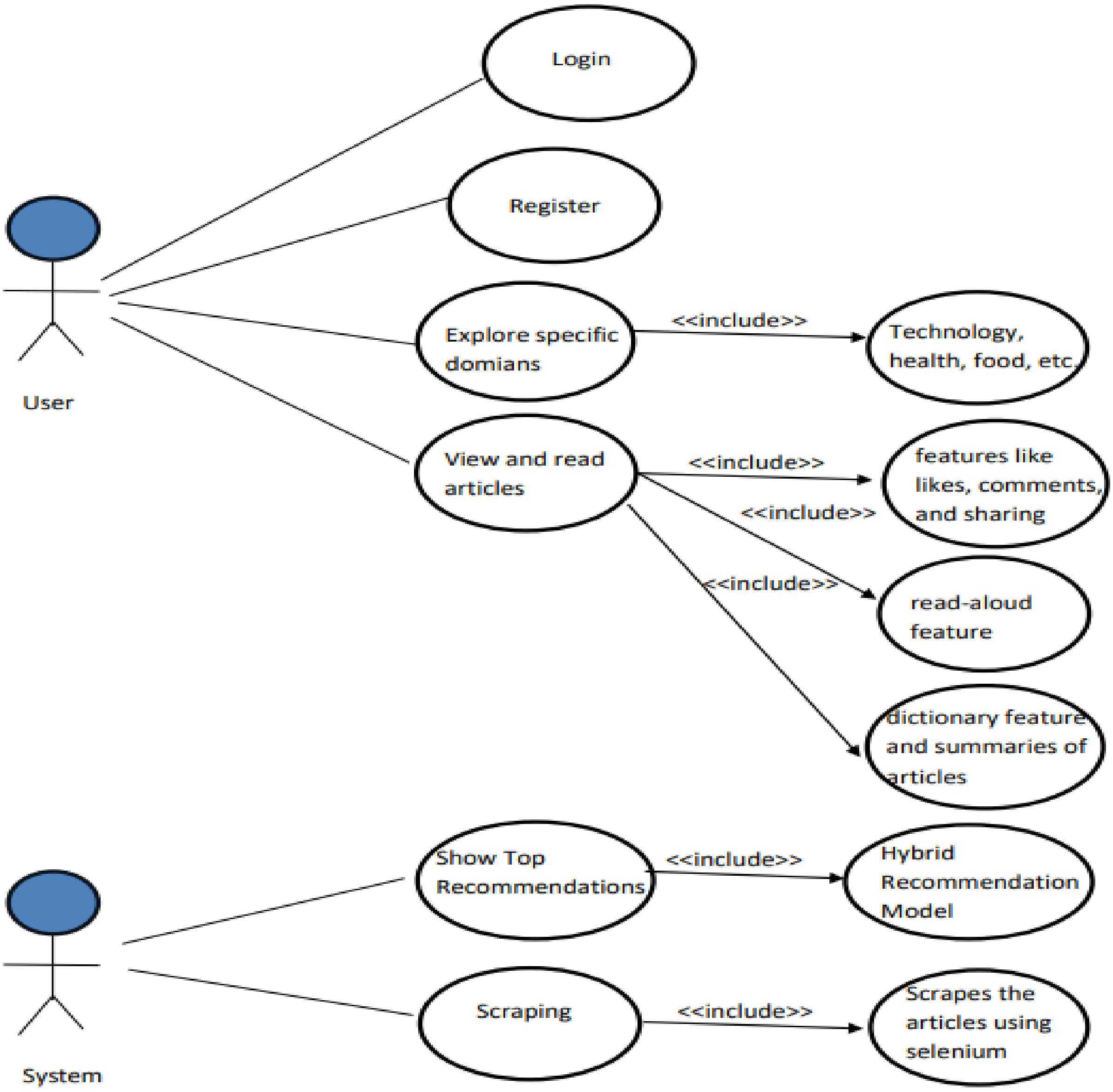


Fig 5.1: Use Case Diagram

The Fig 5.1 illustrates the Use case Diagram of the system.

Actors:

- **User:** Represents the main user interacting with QuickReads.
- **System:** Represents the admin or moderator managing the system.

Use Cases:

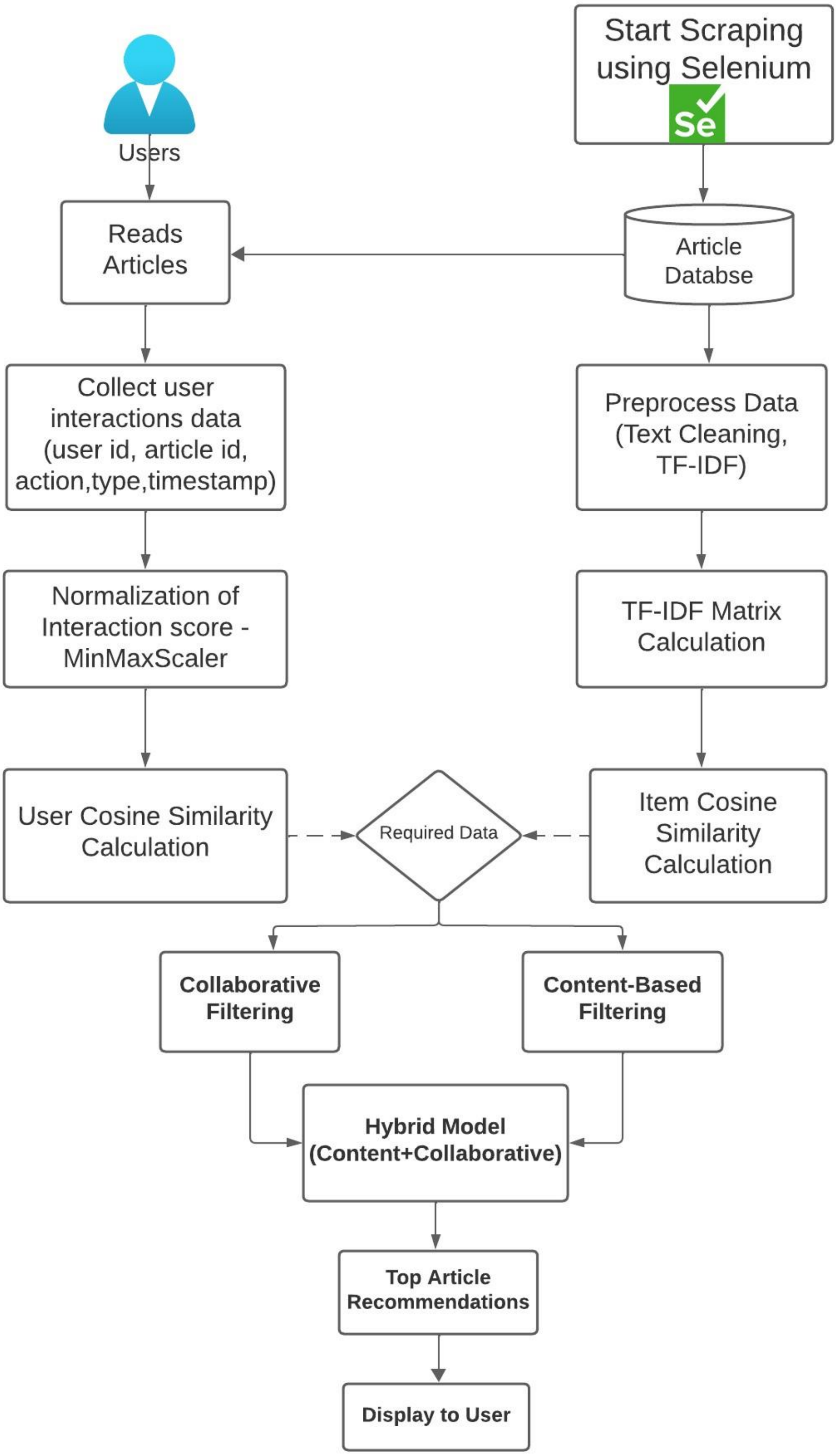
1. **Browse Articles:** Allows users to browse articles based on categories or domains of interest.
2. **Search Articles:** Enables users to search for articles using keywords, titles, or authors.
3. **Read Article:** Allows users to read articles online.
4. **Rate Article:** Enables users to rate articles based on their experience.
5. **Write Review:** Allows users to write reviews for articles.
6. **View Recommendations:** Displays personalized article recommendations based on user preferences and browsing history.
7. **Generate Article Summary:** Utilizes NLP techniques to generate summaries of articles.
8. **Access Dictionary:** Allows users to access definitions of unfamiliar terms within articles.
9. **Read Aloud:** Integrates a read-aloud feature for users to listen to summarized content.

Relationships:

- **Association:** Connects actors with the use cases they interact with.
- **Include Relationships:**
  - Browse Articles includes View Recommendations and Search Articles.
  - Read Article includes Generate Article Summary and Read Aloud.
- **Extend Relationships:** Read Article may extend to include Read Aloud as an optional feature.

5.2 DFD (Data Flow Diagram):

The data flow diagram (Fig 5.2) illustrates the process of building a hybrid recommendation model for the QuickReads article recommendation system:



Start Scraping using Selenium
Se
Users
Reads Articles
Article Databse
Collect user interactions data (user id, article id, action,type,timestamp)
Preprocess Data (Text Cleaning, TF-IDF)
Normalization of Interaction score - MinMaxScaler
TF-IDF Matrix Calculation
User Cosine Similarity Calculation
Required Data
Item Cosine Similarity Calculation
Collaborative Filtering
Content-Based Filtering
Hybrid Model (Content+Collaborative)
Top Article Recommendations
Display to User

Fig 5.2: Data Flow Diagram

1. **User Interaction:**

   - Users read articles from an article database

     Users interact with and read articles from the available collection of articles.

2. **Data Collection:**

   - User interactions (user id, article id, action type, timestamp) are collected

     The system records user interactions with articles, including user ID, article ID, the type of action performed (e.g., read, like, share), and the timestamp of the interaction.

   - Web scraping using Selenium to gather data

     The system employs web scraping techniques using the Selenium tool to collect data from various websites or sources [5].

3. **Article Data Preprocessing:**

   - Text cleaning

     The article text data is cleaned by removing stopwords, stemming, and other preprocessing techniques.

   - TF-IDF calculation

     Term Frequency-Inverse Document Frequency (TF-IDF) scores are calculated for the articles, representing the importance of words in each article relative to the entire corpus.

   - TF-IDF matrix creation

     A TF-IDF matrix is created, which is a numerical representation of the articles based on their TF-IDF scores [4].

4. **User Data Processing:**

   - Normalization of user interaction data using MinMaxScaler

     The user interaction data is normalized using a MinMaxScaler technique, which scales the data to a common range (e.g., 0 to 1).

- Calculation of user cosine similarity scores

  Based on the normalized interaction data, user-cosine similarity scores are calculated to measure the similarity between users based on their interactions with articles.

5. **Filtering Approaches:**

   - Collaborative filtering based on user similarities

     This approach recommends articles based on the preferences of similar users, leveraging the user cosine similarity scores calculated in the previous step.

   - Content-based filtering based on item (article) similarities

     This approach recommends articles that are similar in content to the articles a user has previously engaged with, based on the item (article) cosine similarity scores calculated from the TF-IDF matrix [4].

6. **Hybrid Model:**

   - Combination of collaborative and content-based filtering

     A hybrid model is built by combining the collaborative filtering and content-based filtering approaches, considering both user similarities and item (article) similarities [3].

7. **Recommendation Generation:**

   - Top article recommendations from the hybrid model

     The hybrid model generates a list of top article recommendations tailored for each user, considering both collaborative and content-based information.

8. **Result Display:**

   - Recommended articles displayed to users

     The recommended articles are presented to the users through an appropriate interface or medium, such as a website, mobile app, or email.

This detailed description outlines the end-to-end process of building the hybrid recommendation system, including data collection, preprocessing, filtering techniques, model creation, and delivering personalized article recommendations to users.

5.3 System Architecture:

The System Architecture (Fig 5.3) delineates the step-by-step process, including data collection, preprocessing techniques, featureengineering, model development, and evaluation criteria, all designed to provide a comprehensive understanding of our innovative approach to enhancing QuickReads Hybrid Recommendation System.

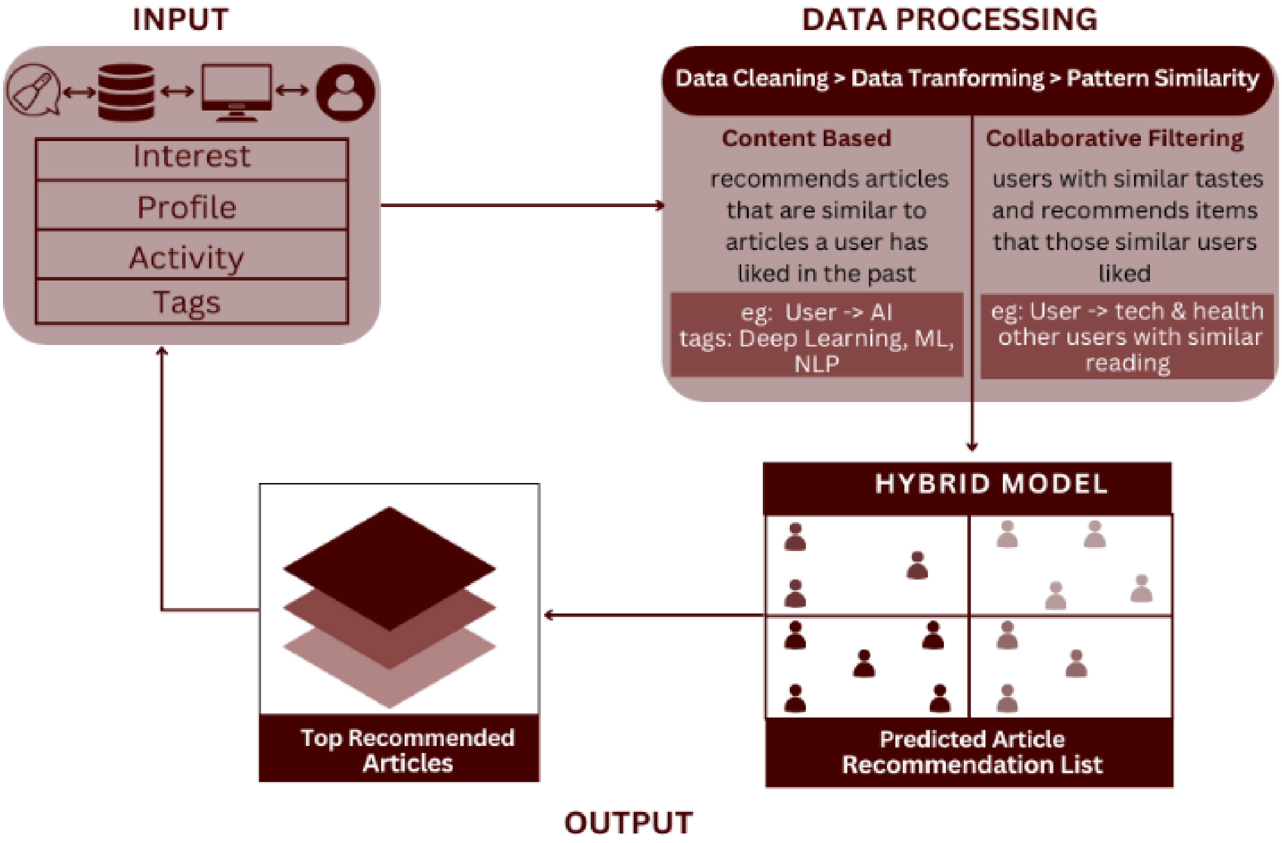

INPUT
Interest
Profile
Activity
Tags
DATA PROCESSING
Data Cleaning > Data Tranforming > Pattern Similarity
Content Based
recommends articles that are similar to articles a user has liked in the past
eg: User -> AI tags: Deep Learning, ML, NLP
Collaborative Filtering
users with similar tastes and recommends items that those similar users liked
eg: User -> tech & health other users with similar reading
HYBRID MODEL
Predicted Article Recommendation List
Top Recommended Articles
OUTPUT

Fig 5.3: System Architecture

1. **Data Collection:**

   Article Data is collected using web scraping. For this Selenium tool is used to perform dynamic web scraping of the articles [5].

2. **Data Preprocessing:**

   User interactions data, including user IDs, article IDs, and actions, are collected and preprocessed to remove duplicates and format timestamps. Articles data, such as article IDs, titles, and types are preprocessed to standardize date formats and prepare for feature extraction.

3. **Feature Engineering:**

TF-IDF vectors are generated for article titles and types using Tfidf Vectorizer [4]. Similarity scores are computed based on article titles using linear_kernels.

$$TF(t,d) = \frac{number\ of\ times\ t\ appears\ in\ d}{total\ number\ of\ terms\ in\ d}$$

$$IDF(t) = log\frac{N}{1+df}$$

$$TF-IDF(t,d) = TF(t,d) * IDF(t)$$

where, d refers to a title or type of article,
N is the total number of articles,
df is the number of articles with term t.

4. **Normalization and Matrix Creation:**

Interaction scores are normalized using MinMaxScaler. A user-item interaction matrix is created to represent user-article interactions, and it is converted to a sparse matrix for efficient computation.

5. **Cosine-similarity:**

Model computes similarity scores based on article titles using linear kernel (cosine_sim_title) and article types using linear kernel (cosine_sim_type).

The cosine similarity formula measures the cosine of the angle between two vectors A and B, representing the similarity between the vectors. In this case, A and B are TF-IDF vectors.

$$\text{cosine\_sim}(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|}$$

6. **User Similarity Matrix:**

User similarity matrix is computed based on interactions using pairwise_distances.

7. Recommendation Functions:

i. get_collaborative_recommendations: This function utilizes the user_similarity matrix to generate top N recommendations for a user. It calculates the similarity scores between the target user and other users. Identifies the most similar users based on these scores. Aggregates the interaction scores of these similar users with articles to recommend the top N articles that the target user is likely to be interested in.

ii. get_content_based_recommendations: This function provides top N recommendations for a user using content-based filtering. It identifies articles that the user has viewed. Computes the similarity scores between these viewed articles and all other articles based on their TF-IDF vectors. Recommends articles with the highest similarity scores, excluding those already viewed by the user.

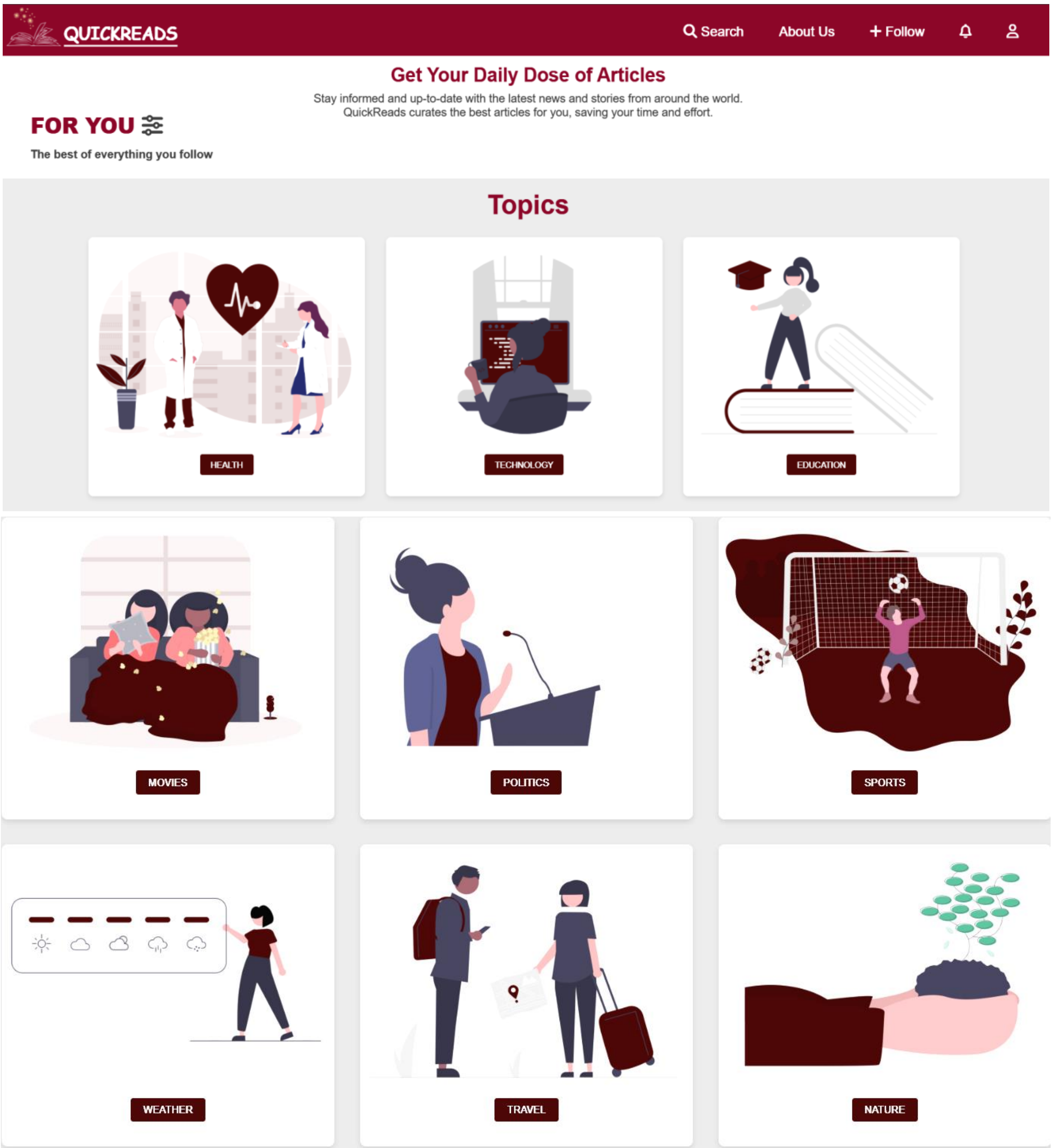
iii. get_hybrid_recommendation: Combines collaborative and content-based recommendations to provide hybrid recommendations. Removes duplicates and presents the top N hybrid recommendations to the user.[3]

iv. Evaluation Metrics: Accuracy metrics such as precision, recall, and F1-score are computed to evaluate the model's performance. Data is split into training and test sets, and recommendations are made for users in the test set to assess the model's accuracy.

5.4 Implementation:

The Implementation demonstrate the effectiveness of the proposed article recommendation system. The hybrid approach achieves a significant improvement in accuracy compared to individual techniques. The precision, recall, and F1-score metrics indicate the system's ability to make relevant and personalized recommendations to users.

In QuickReads, users have the freedom to choose from various domains or topics of interest right from the start as illustrated in Fig 5.4.1 This user-centric approach enhances the personalization of article recommendations and allows users to tailor their content discovery experience according to their preferences.



Get Your Daily Dose of Articles

Stay informed and up-to-date with the latest news and stories from around the world. QuickReads curates the best articles for you, saving your time and effort.

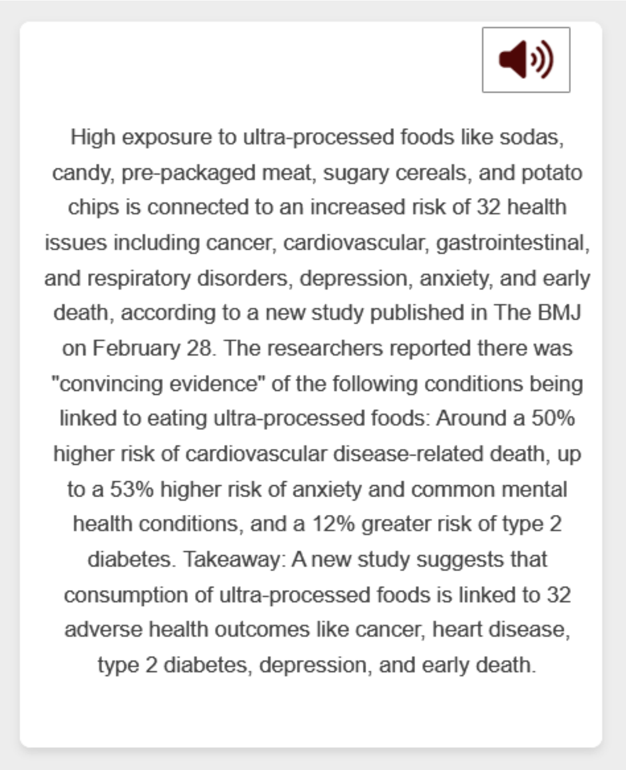
FOR YOU

The best of everything you follow

Topics

Figure 5.4.1: Topics of Interest

QuickReads includes a summary feature to provide users with concise and informative summaries of articles, enhancing the browsing experience and facilitating quick comprehension of content. This feature as illustrated in Fig 5.4.2 is particularly useful for users who want to grasp the main points of an article without reading the entire text. The process of summarizing text data from a CSV file by utilizing Natural Language Processing (NLP) techniques. It reads text from the 'Content' column of the CSV file, tokenizes the text into words and sentences, removes stop words, and calculates the frequency of words in each sentence. Based on the frequency analysis, it assigns values to sentences and generates a summary by selecting sentences with values above a certain threshold, ensuring that the summary captures the most relevant information. The script then updates the CSV file by adding a new column containing the generated summaries. This approach streamlines the text summarization process, making it efficient for handling large volumes of textual data and extracting key insights from the text for further analysis or presentation.



High exposure to ultra-processed foods like sodas, candy, pre-packaged meat, sugary cereals, and potato chips is connected to an increased risk of 32 health issues including cancer, cardiovascular, gastrointestinal, and respiratory disorders, depression, anxiety, and early death, according to a new study published in The BMJ on February 28. The researchers reported there was "convincing evidence" of the following conditions being linked to eating ultra-processed foods: Around a 50% higher risk of cardiovascular disease-related death, up to a 53% higher risk of anxiety and common mental health conditions, and a 12% greater risk of type 2 diabetes. Takeaway: A new study suggests that consumption of ultra-processed foods is linked to 32 adverse health outcomes like cancer, heart disease, type 2 diabetes, depression, and early death.

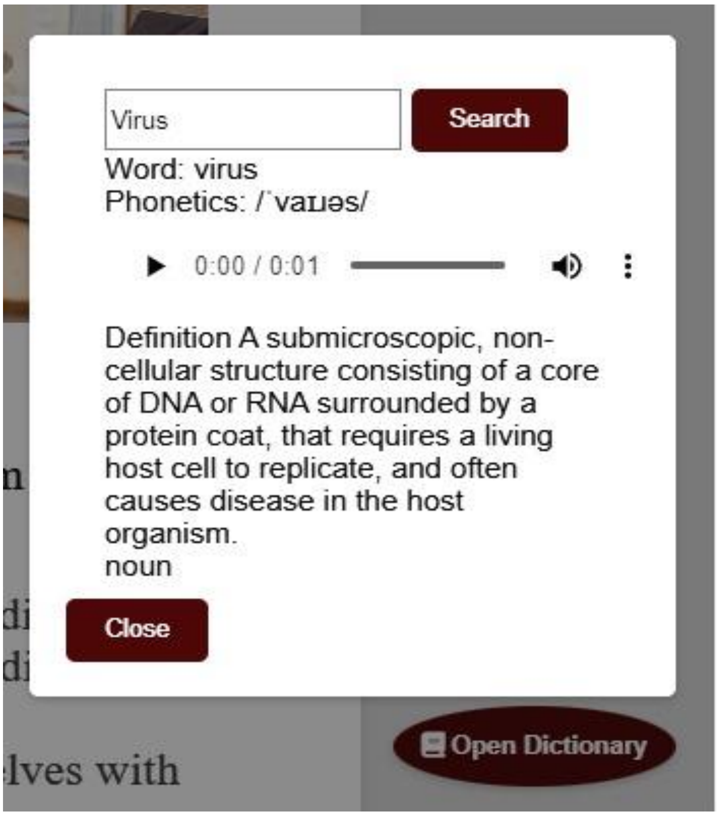
Figure 5.4.2: Summary and Read Aloud for Article

The "Read Aloud" option [1] for summaries in QuickReads enhances accessibility and user experience by providing an audio version of the summarized content. This feature is especially beneficial for users who prefer auditory learning styles. When users choose the "Read Aloud" option for a summary, QuickReads utilizes SpeechSynthesisUtterance API to convert the summarized text into spoken audio. Users can listen to the summary through their device's speakers or headphones, allowing them to absorb the information hands-free while engaging in other activities.

The dictionary feature in Fig 5.4.3 allows users to search for word meanings, pronunciation, and definitions using a dictionary API. When a user enters a word in the text box, they receive the following information:

1. Meaning: The meaning or definitions of the word.
2. Pronunciation: The pronunciation of the word.
3. Read Aloud: An option to listen to the pronunciation of the word.

This feature enhances the user experience by providing comprehensive information about the word they are searching for, including its definition and how it is pronounced.



Word: virus
Phonetics: /ˈvaɪɹəs/

Definition A submicroscopic, non-cellular structure consisting of a core of DNA or RNA surrounded by a protein coat, that requires a living host cell to replicate, and often causes disease in the host organism.
noun

Figure 5.4.3: Dictionary

Chapter 6

Technical Specification

The technical specifications of a project outline the critical architectural elements, tools, and technologies that underpin its development and operation. This section provides a detailed description of the system architecture, data management approaches, algorithms employed, and the specific software and hardware components integrated into the project. It includes information about the frameworks, databases, APIs, and user interface technologies used, as well as details on security measures, accessibility considerations, and performance optimization strategies.

Technical specifications are essential for understanding how the project is built and maintained; offering insights into its scalability, reliability, and the user experience it delivers. This documentation is crucial for developers, stakeholders, and technical teams involved in the project as it serves as a blueprint for current operations and future enhancements. By detailing the technical environment and the interactions between various components, these specifications help ensure that the project meets its intended functionality and performance goals efficiently and effectively

- **Frontend:**

  1. HTML & CSS
  2. Python 3.8
  3. JavaScript

Technologies: The frontend of QuickReads is developed using HTML, CSS, and JavaScript. These foundational technologies ensure that the user interface is intuitive, responsive, and accessible across various devices and platforms.

Python Version: Python 3.11.0 is utilized for scripting and automation tasks within the frontend environment, enhancing features such as dynamic content loading and real-time updates.

- **Backend Development:**

  1. Django 4.2.5
  2. SQLite3

Framework: Django 4.2.5 serves as the backend framework for QuickReads. It provides a robust and scalable structure for handling server-side logic, database operations, and application routing.

Database: SQLite3 is employed as the relational database management system to store and manage structured data including user profiles, article metadata, and interaction logs.

These technical specifications outline a comprehensive and forward-thinking approach, ensuring that QuickReads remains at the cutting edge of technology for content recommendation systems. The integration of advanced tools and methodologies allows QuickReads to deliver a highly personalized and engaging user experience, setting a new standard in the domain of digital content discovery.

Chapter 7

Project Scheduling

In project management, a schedule is a listing of a project's milestones, activities, and deliverables. A schedule is commonly used in the project planning and project portfolio management parts of project management. The project schedule (Table 7.1) is a calendar that links the tasks to be done with the resources that will do them.

Sr. No.	Group Members	Duration	Task Performed
1.	Sonal Sonarghare Janhavi Kasar Sanskruti Chavan Harsh Shelke	2nd Week of January	Group formation and Topic finalization. Identifying the scope and objectives of the Mini Project. Discussing the project topic with the help of a paper prototype.
		1st Week of February	Identifying the functionalities of the Mini Project.
2.	Janhavi Kasar	2nd Week of February	Training the model of hybrid recommendation system based on articles and user interaction dataset.
3.	Sonal Sonarghare	3rd Week of February	Designing the Graphical User Interface (GUI)
4.	Janhavi Kasar Sanskruti Chavan	4th Week of February	Adding the features of the QuickReads like Summary, Read Aloud and Dictionary.
5.	Sonal Sonarghare Janhavi Kasar	1st Week of March	Integrating the model on GUI and connecting with the database.
6.	Sonal Sonarghare Harsh Shelke	3rd Week of March	Database connectivity of all modules.

Table 7.1: Project Task Distribution

A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. Gantt chart (Fig 7.1) illustrates the start and finish dates of the terminal elements and summary elements of a project.

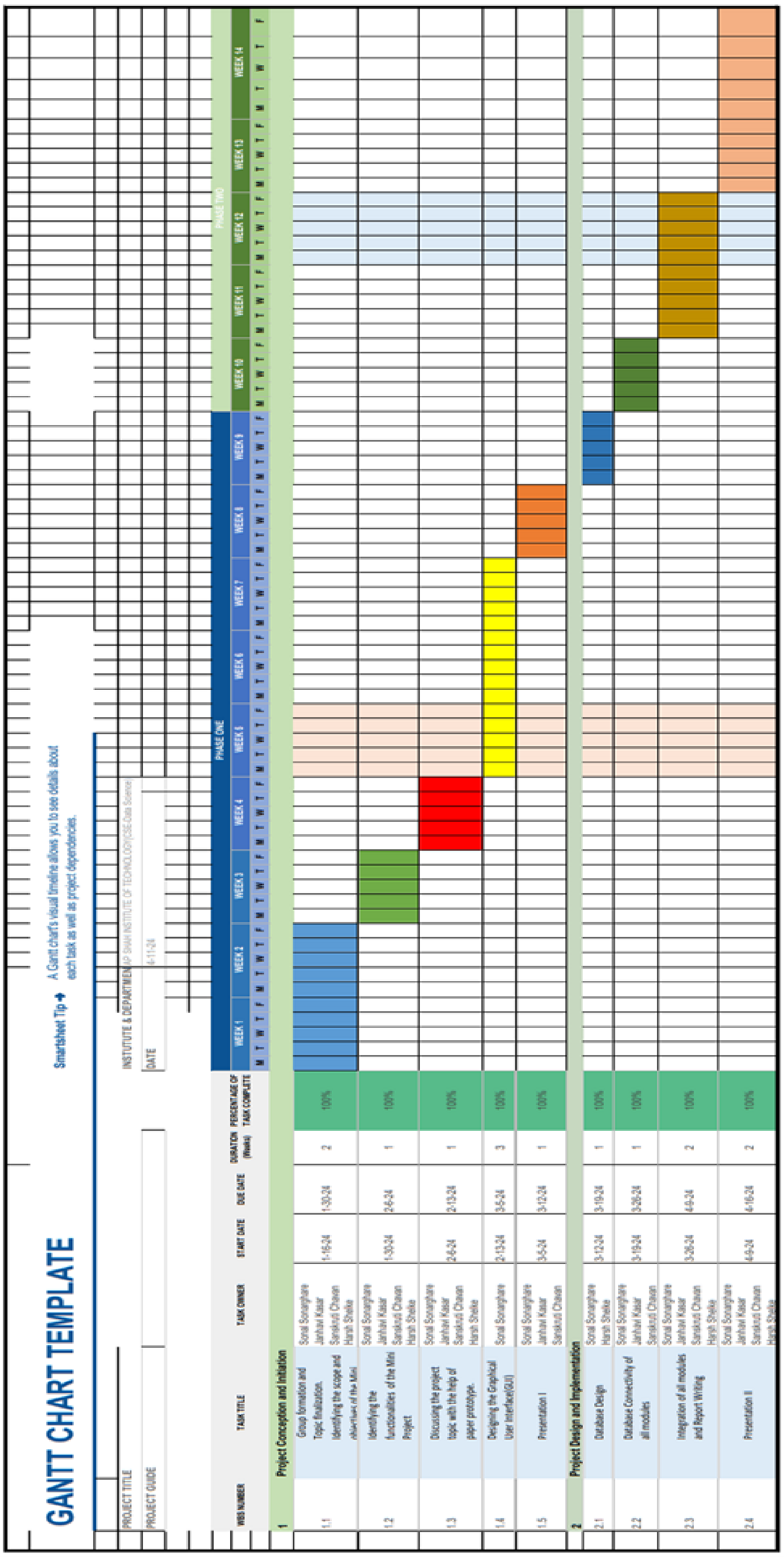


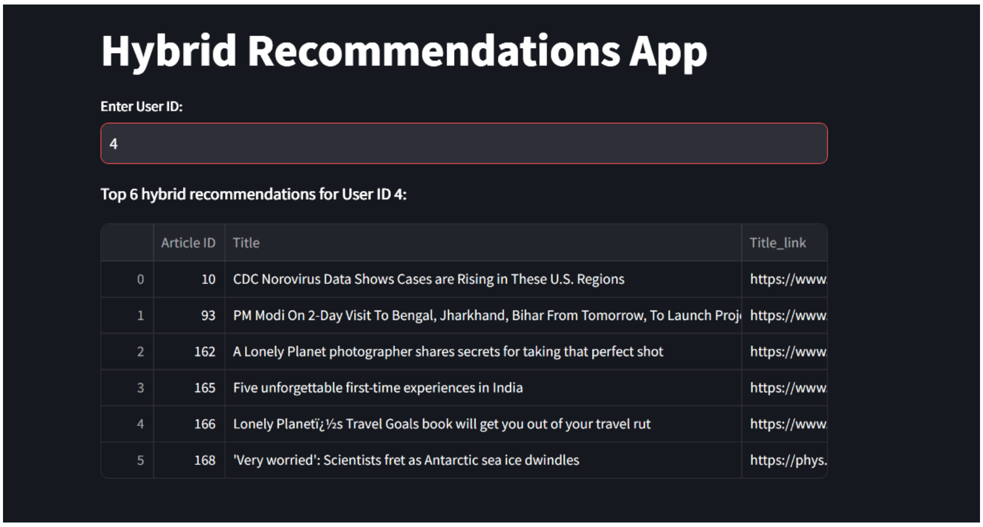
Fig 7.1: Gantt Chart of QuickReads

Chapter 8

Results

The project results section provides a concise overview of the outcomes achieved through the implementation of the project. Highlighting key findings, deliverables, and the final implementation of the project lifecycle. This section serves to summarize the tangible outcomes and impacts of the project, providing stakeholders with valuable insights into its overall effectiveness and contribution to the intended objectives.

QuickReads gives top 5/6 recommendations to individual users. These recommendations (Fig 8.1) are generated using a hybrid recommendation model, which combines collaborative filtering and content-based filtering techniques to offer more accurate and diverse suggestions. This represents the articles deemed most relevant and interesting for the user based on their past interactions and preferences.



Hybrid Recommendations App

Enter User ID:

4

Top 6 hybrid recommendations for User ID 4:

	Article ID	Title	Title_link
0	10	CDC Norovirus Data Shows Cases are Rising in These U.S. Regions	https://www.
1	93	PM Modi On 2-Day Visit To Bengal, Jharkhand, Bihar From Tomorrow, To Launch Proj	https://www.
2	162	A Lonely Planet photographer shares secrets for taking that perfect shot	https://www.
3	165	Five unforgettable first-time experiences in India	https://www.
4	166	Lonely Planetï¿½s Travel Goals book will get you out of your travel rut	https://www.
5	168	'Very worried': Scientists fret as Antarctic sea ice dwindles	https://phys.

Fig 8.1: Top Recommended article

The figure (Fig 8.2) shows the distribution of user interactions on the QuickReads platform. Each bar represents a different type of interaction (e.g., viewing, liking, commenting), and the height of each bar indicates the frequency of that interaction type. This visualization offers a quick overview of how users engage with the platform and helps identify popular actions and user behavior patterns.

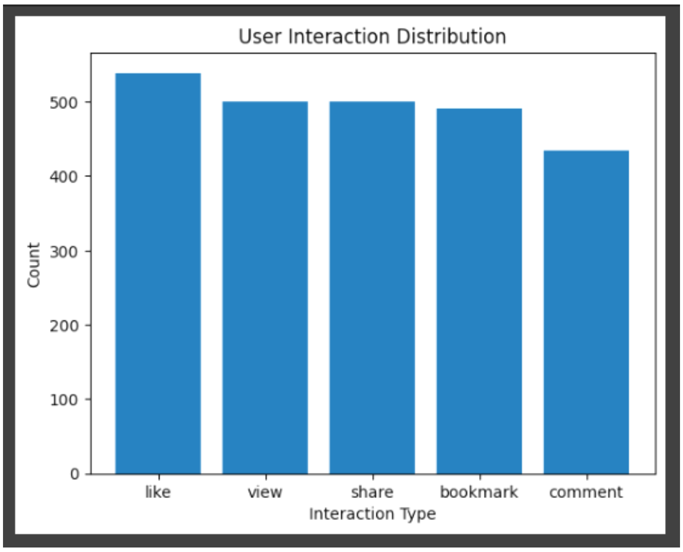
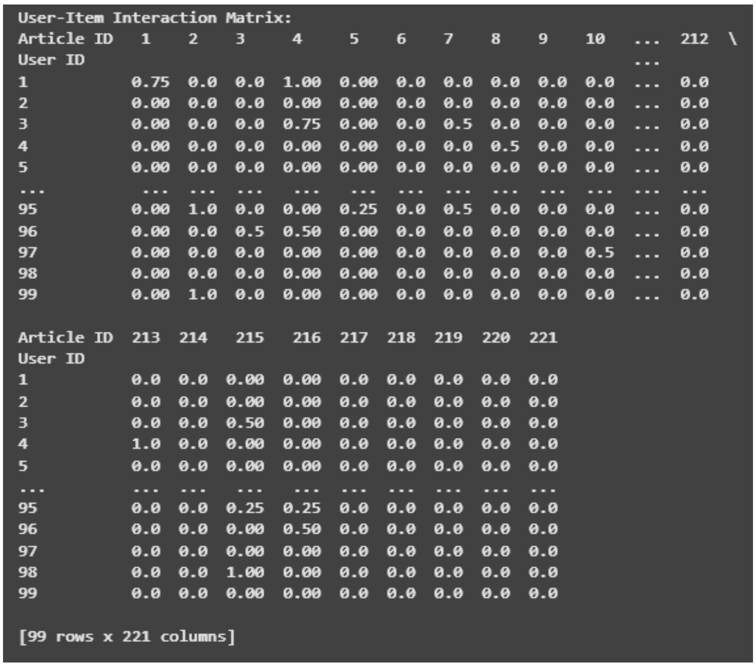


Fig 8.2: User Interaction Distribution

The user-item interaction matrix (Fig 8.3) is a fundamental component in recommendation systems, especially collaborative filtering. This matrix organizes users along the rows and articles along the columns, with each cell representing a user's interaction score with a particular article. These scores could signify various user actions such as views, likes, comments, etc., which are quantified and standardized to facilitate comparison. By filling missing values with zeros, it ensures a consistent structure. This matrix serves as the foundation for computing user similarities and generating personalized recommendations. Printing the matrix offers insights into user preferences and interactions with articles, crucial for understanding user behavior and tailoring recommendations accordingly.



```
User-Item Interaction Matrix:
Article ID     1    2    3     4     5    6    7    8    9   10  ...  212  \
User ID                                                           ...
1           0.75  0.0  0.0  1.00  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0
2           0.00  0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0
3           0.00  0.0  0.0  0.75  0.00  0.0  0.5  0.0  0.0  0.0  ...  0.0
4           0.00  0.0  0.0  0.00  0.00  0.0  0.0  0.5  0.0  0.0  ...  0.0
5           0.00  0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0
...          ...  ...  ...   ...   ...  ...  ...  ...  ...  ...  ...  ...
95          0.00  1.0  0.0  0.00  0.25  0.0  0.5  0.0  0.0  0.0  ...  0.0
96          0.00  0.0  0.5  0.50  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0
97          0.00  0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.5  ...  0.0
98          0.00  0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0
99          0.00  1.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0  ...  0.0

Article ID  213  214   215   216  217  218  219  220  221
User ID
1           0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0
2           0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0
3           0.0  0.0  0.50  0.00  0.0  0.0  0.0  0.0  0.0
4           1.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0
5           0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0
...         ...  ...   ...   ...  ...  ...  ...  ...  ...
95          0.0  0.0  0.25  0.25  0.0  0.0  0.0  0.0  0.0
96          0.0  0.0  0.00  0.50  0.0  0.0  0.0  0.0  0.0
97          0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0
98          0.0  0.0  1.00  0.00  0.0  0.0  0.0  0.0  0.0
99          0.0  0.0  0.00  0.00  0.0  0.0  0.0  0.0  0.0

[99 rows x 221 columns]
```

Fig 8.3: User-Item Interaction Matrix

Chapter 9

Conclusion

In conclusion, QuickReads stands as a groundbreaking advancement in the landscape of content discovery platforms. Its revolutionary blend of cutting-edge web scraping technologies and intricate hybrid recommendation models heralds a new era of personalized and dynamic user experiences, surpassing the constraints of conventional article recommendation systems.

This innovative platform not only streamlines navigation through the vast expanse of online content but also elevates user engagement through its astute curation of articles finely attuned to individual preferences. Empowered by an intuitive user interface and a steadfast commitment to user-centric design, QuickReads empowers users to seamlessly explore, engage with, and derive value from content that resonates deeply with their specific educational and informational needs.

Moreover, QuickReads transcends its role as a mere informational tool, evolving into a cornerstone of intellectual enrichment and personal growth. With its advanced algorithms continuously honed through iterative learning from user interactions, the platform facilitates a bespoke journey of knowledge exploration, igniting curiosity, and nurturing the expansion of intellectual horizons.

This transformative approach not only enhances the user experience but also cultivates a profound understanding and appreciation of diverse subjects, cementing QuickReads as an indispensable companion in the pursuit of knowledge and lifelong learning in our digital age. As users traverse the ever-expanding realm of information, QuickReads stands ready as a steadfast ally, guiding them towards enlightenment and empowerment on their intellectual odyssey.

Chapter 10

Future Scope

Expanding QuickReads to include article generation, image generation, and other features can significantly enhance its value proposition. Here are some potential future scope ideas:

1. **Article Generation:**

   Implementing a feature that generates articles based on user preferences, trending topics, or specific keywords can automate content creation. This could involve utilizing natural language processing (NLP) techniques like text generation models (e.g., GPT) to produce high-quality articles.

2. **Image Generation:**

   Integrating image generation capabilities can complement the articles and enhance user engagement. This could involve generating relevant images based on article content, user preferences, or using generative adversarial networks (GANs) to create images from scratch.

3. **Multi-Modal Content:**

   Combine text, images, and potentially other media formats (such as videos or audio) to provide a richer user experience. This could involve integrating multimedia content creation and consumption capabilities into the platform.

4. **Interactive Features:**

   Implement interactive features such as quizzes, polls, or interactive visualizations within articles to engage users and encourage active participation.

5. **Advanced Analytics and Insights:**

   Provide users with analytics and insights on their reading habits, preferences, and interests. This could include tracking metrics such as reading time, popular topics, user engagement, and sentiment analysis to provide personalized recommendations and content insights.

6. **Mobile Application:**

   Develop a mobile application to extend the reach of QuickReads and provide users with on-the-go access to content, offline reading capabilities, and personalized notifications.

REFERENCES

[1] Essien, Nesabasi P., Victoria A. Uwah, and Emmanuel P. Ododo. "An interactive intelligent web-based text-to-speech system for the visually impaired." Asia-Africa Journal of Recent Scientific Research 1 (2021): 76-98.

[2] Singhal, A., Rastogi, S., Panchal, N., Chauhan, S., & Varshney, S.(2021). "Research Paper On Recommendation System." Global Scientific Journals.

[3] Ye, Bo Kai, Yu Ju Tony Tu, and Ting Peng Liang. "A hybrid system for personalized content recommendation." Journal of Electronic Commerce Research 20.2 (2019): 91-104.

[4] Zhang, Sen. "Restaurant Recommendation System Based on TF-IDF Vectorization: Integrating Content-Based and Collaborative Filtering Approaches." 2023 International Conference on Data Science, Advanced Algorithm and Intelligent Computing (DAI 2023). Atlantis Press, 2024.

[5] Rusdiansyah, Rusdiansyah, et al. "Web Program Testing Using Selenium Python: Best Practices and Effective Approaches." *Sinkron: jurnal dan penelitian teknik informatika* 8.2 (2024): 994-1000.