
Project Proposal

for

Mirror News Summarizer

Student Name: Zijun Li

Student ID: 2272583

Supervisor Name: Jizheng Wan

Project Category/Topic: AI

1 Project Aim:

- **Objective:** The goal is to develop 'Mirror News Summarizer', an advanced platform that revamps the user experience of consuming news online. It extracts content from selected mirror news sites, eliminates advertisements, and replaces full articles with succinct summaries. This system aims to streamline the process of information acquisition, making news reading more efficient, and user-friendly.
- **Significance:** With the advent of the digital age and the vast array of information available online, it's becoming increasingly challenging for users to sift through extensive content efficiently. Especially in the realm of news where staying updated is crucial, navigating through lengthy articles can be time-consuming. The "Mirror News Summarizer" strives to bridge this gap by allowing users to view compressed news articles that retain the core essence of the original. This not only ensures the efficient consumption of news but also reduces the cognitive load on the user.
- **Relevance to AI:** The 'Mirror News Summarizer' leverages cutting-edge AI technologies and methodologies for several core functionalities:

Content Retrieval: Utilizes 'request crawlers' for scraping news content from various sources, ensuring real-time, diverse, and vast information procurement.

Summarization: Employs Summary APIs, capitalizing on advanced NLP models like GPT (Generative Pre-trained Transformer), to generate precise and coherent news summaries. This AI-driven approach understands the context and extracts essential details, contributing to a succinct yet comprehensive summary.

Recommendation System: Integrates existing AI recommendation models that analyze user behavior, preferences, and interaction history to provide personalized news content, enhancing user engagement and satisfaction.

Text Similarity Analysis: Implements AI algorithms for text similarity checks to identify and eliminate duplicate content from different sources, presenting unique content to the users.

2 Literature Review:

-

3 Project Objectives/Deliverables

1. User-Centric News Access and Customization

- Develop a user interface within the 'Mirror News Summarizer' that allows individuals to choose their preferred news sources and categories, enhancing personalized interaction.
- Implement a content filtration and retrieval system that curates and presents news from selected mirror news websites, optimizing the influx of information based on user preferences.

2. Enhanced Content Presentation

- Design an intuitive layout to display summarized news content, incorporating essential details like the headline, source, publication date, and core points, all within a user-friendly, visually appealing framework.
- Establish a mechanism for users to customize the news summary display, including aspects like theming (dark mode, font choices) and organization (list view, card view), further improving user experience and accessibility.

3. Intelligent Content Summarization and Duplication Check

- Execute an AI-driven process for dynamic news content summarization, ensuring that the essence and critical points of the full articles are cohesively maintained in the shortened version.
- Integrate a text similarity analysis feature that identifies and filters out repetitive news from different sources, thereby avoiding content redundancy and enhancing the quality of content presented to the user.

4. Adaptive Recommendation and Interaction System

- Construct a recommendation engine that analyses user interactions, preferences, and trending news to suggest relevant content, thereby personalizing the user's news feed and improving content relevancy.
- Develop interactive components within the platform, such as feedback options for summaries and bookmarking capabilities for preferred news articles, fostering user engagement and platform loyalty.

5. Voice Integration and Multilingual Support

- Integrate a text-to-speech feature, enabling users to listen to news summaries, thereby accommodating different user needs and enhancing accessibility.

- Incorporate a multilingual interface, allowing the platform to serve a broader demographic, catering to users from various linguistic backgrounds, and expanding the platform's reach.

6. User Engagement and Analytic Reports

- Implement a system for tracking user engagement metrics, preferences, and feedback, utilized for continuous improvement of the platform's features and user experience.
- Generate periodic analytic reports summarizing user interaction trends, content popularity, and feedback, providing valuable insights for future enhancements and content curation strategies.

These objectives comprehensively address the primary goal of this project: to provide users with an efficient way to access and consume news articles from their favorite websites, without the distractions of unnecessary content. The initial objective guarantees that the system is equipped to fetch content autonomously, enabling a continuous flow of updated news. The following objectives focus on processing and optimizing this content, from extraction and filtering to summarization and storage. The final objective ensures that the users receive their content in a familiar and user-friendly interface, promoting engagement and satisfaction. Overall, this approach ensures users can quickly access the essence of news articles in a format that's convenient and recognizable.

4 methodologies

1. **User Preferences and Customization Setup:** Initiate user interaction by offering a setup interface where they can select preferred news categories, sources, and UI/UX elements (e.g., dark mode, text size). This step personalizes the subsequent content curation.
2. **Content Aggregation and Filtration:** Utilize web scraping tools (like requests) and APIs to aggregate news content from various mirror websites selected by the user. This process includes filtering out ads and non-essential elements, focusing solely on the actual news content.
3. **AI-driven News Summarization:** Implement an AI model API to generate concise, coherent summaries of the aggregated news articles. This step requires maintaining the integrity and critical nuances of the original news content while reducing the length.
4. **Duplication and Redundancy Removal:** Apply text similarity algorithms to identify and eliminate duplicate content across multiple news sources, ensuring a unique, clutter-free user feed.

| Week | Dates | Tasks |
|------|-------|-------|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

5. **Dynamic Content Presentation:** Display the summarized content to users in a structured, customizable UI, allowing for different viewing preferences (e.g., list view, card view). This stage involves frontend technologies like React for a responsive design.
6. **User-Interactive Recommendation System:** Integrate a recommendation system that tracks user engagement (clicks, reading time) and utilizes existing models to suggest relevant news, enhancing the personalization aspect of content delivery.
7. **User Feedback and Continuous Improvement:** Allow users to provide feedback on news summaries and recommendations, which the system will use for continuous learning and improvement. This iterative process is crucial for maintaining content relevance and accuracy.
8. **Text-to-Speech (TTS) and Multilingual Support Integration:** Incorporate TTS functionality for users to listen to summaries, enhancing accessibility, and convenience. Additionally, implement multilingual support to cater to a diverse user base.
9. **Backend Support and Data Management:** Use backend technologies like Flask for managing data transactions, user requests, and system responses, ensuring a seamless, efficient backend operation.
10. **Evaluation and User Testing:** Conduct comprehensive user testing to evaluate the system's performance, particularly the accuracy of news summaries, effectiveness of the recommendation engine, and overall user satisfaction. Collect feedback for future enhancements.

This methodology embodies a holistic approach to developing the 'Mirror News Summarizer' platform. Each step is designed to ensure that the final product is efficient, user-centric, and innovative, significantly enhancing the digital news consumption experience for users.

5 Project Plan

6 Risks and contingency plan

- **Risk:** Web scraping tools often depend on the consistent structure of a webpage. If major news sites undergo a significant design overhaul, the scraping tools developed for the 'Mirror News Summarizer' might fail to aggregate content effectively.

Contingency: Should our primary targeted news sources change their webpage structures, we plan to quickly adapt our scraping scripts. Furthermore, we'll set up periodic monitoring tools to check the efficiency of our scrapers and get alerts for potential failures.

- **Risk:** News websites frequently employ anti-scraping technologies and techniques to prevent automated bots from accessing their content. This can hinder the consistent inflow of news data to the platform.

Contingency: In the event we encounter difficulties accessing content from a particular source, we'll pivot to integrate RSS feeds or official news APIs, which offer structured and consistent access to news articles.

- **Risk:** Third-party APIs often have usage limits or rate limits that could restrict the extent and frequency of operations, affecting the performance of our summarization system if the project scales up.

Contingency: If we approach the API rate limits, our plan is to integrate caching mechanisms to store frequently accessed summaries. Additionally, we'll explore the possibility of incorporating other summarization algorithms or tools to diversify our backend and reduce dependency on a single API.

7 Hardware/Software Resource

1. Hardware:

- RAM: 4 GB (recommended for efficient data processing and handling multiple tasks)
- Storage: 50 GB SSD (for faster read/write speeds and storage of large datasets, cache data, and logs)

2. Software Requirements:

- OS: Windows (with the latest security updates)

- Python 3.8 or later (for compatibility with the latest libraries and frameworks)
- Text Editor or IDE: VS Code/PyCharm (equipped with relevant Python extensions and plugins)
- Browser: Google Chrome/Edge (latest version, needed for testing the web interface)
- Database Management System: MySQL/PostgreSQL/SQLite (for storing user profiles, interactions, and logs)
- Backend: Flask/Django (Python-based web frameworks for handling backend operations)
- Frontend: React.js (for building a dynamic and responsive user interface)
- Web scraping tools: BeautifulSoup/Selenium (libraries for scraping news content)
- Summary API key (for content summarization functionalities)
- Version Control: Git (for source code management and collaborative development)

8 Data

- The primary data for the project comprises news articles and content fetched from various online news websites through web scraping techniques.
- The data includes textual content, publication dates, author names, and other relevant metadata associated with each news article.
- User interaction data, such as reading preferences, feedback on summaries, and browsing history, will be collected through the platform to enhance the recommendation system.
- Ethical considerations and privacy compliance measures are in place for handling user data, with protocols established for secure data storage, retrieval, and processing.