Intelligent Research Assistant Application for Online Academic Research

Introduction

This proposal outlines the design and development of an intelligent software system specializing in facilitating online academic research tasks. This system, hereafter referred to as the "research assistant application", incorporates advanced functionalities to enhance its capabilities beyond basic web scraping. The application aims to assist researchers by automating information retrieval from online sources based on user-defined criteria. It will intelligently search websites, extract relevant data points, and organize them for offline storage and further analysis.

Domain and Requirements

The chosen domain is **Academic Research Online**. The research assistant application will address the following requirements:

- **Identify and retrieve data:** The application will intelligently search websites based on user-provided keywords or search terms. It will consider synonyms, related concepts, and explore various academic databases to gather comprehensive results.
- **Process data:** The extracted data will be formatted and organized for easy reference and analysis. The application will utilize Natural Language Processing (NLP) techniques to understand the context and relationships between extracted information.
- **Store/save/present data:** The processed information will be saved for offline access and further and faster research. The application can also present key findings and insights in a summarized format.

Research Assistant Application Design and Functionality

The proposed application will consist of several modules, leveraging intelligent techniques:

- **Intelligent Search Module:** Will take user input specifying the research topic. It will utilize NLP to understand the intent and context of the search. The application can then identify relevant keywords, synonyms, and related concepts to broaden the search scope. It will also query academic databases and scholarly search engines for comprehensive results.
- **Data Extraction Module:** Will employ web scraping techniques to retrieve relevant information from web pages. NLP techniques will be used to parse the content, identify key data points, and extract them with high accuracy.
- **Data Processing and Analysis Module:** This will use NLP to understand the relationships between extracted information. It can perform tasks like entity recognition and sentiment analysis. The data will be organized and formatted.
- **Data Presentation Module:** This will leverage Natural Language Generation (NLG) techniques to summarize key findings and insights extracted from the research data. This can provide researchers with a quick overview of the gathered information.

The sequence diagram shows the interaction sequence between different parts of the system (see Fig. 1).

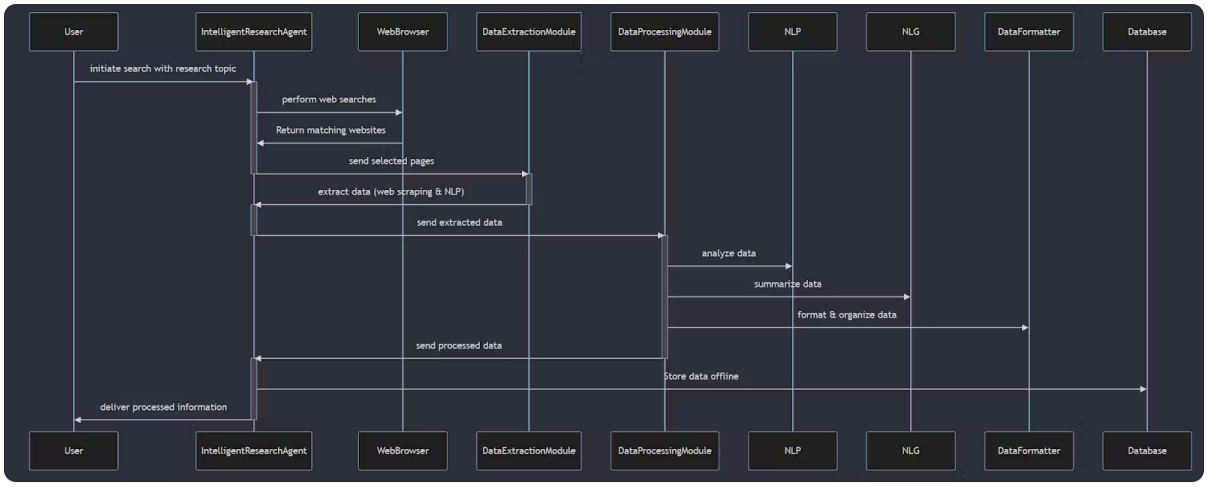

Figure 1. Sequence diagram.

System Architecture

The application will operate as a standalone application with an intelligent core. The user interface will allow for specifying research topics and defining data extraction preferences. The application will run in the background, performing intelligent search, data extraction, processing, analysis, and presentation tasks autonomously.

The user activity diagram better represents the workflow on a high level (see Fig. 2).

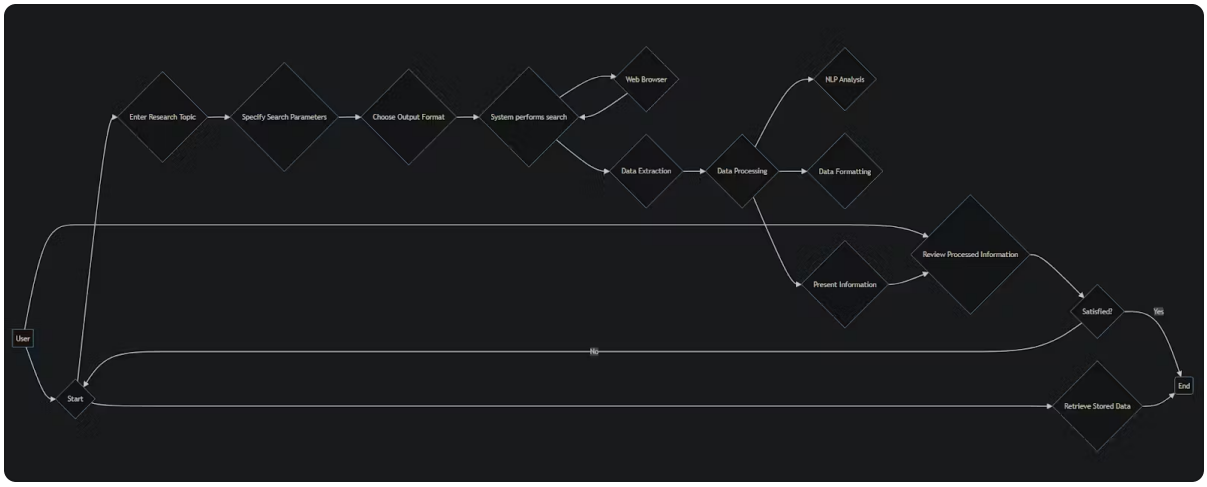

Figure 2. User activity diagram

Development Tools and Technologies

Jennings (2000), Kinny & Georgeff (2005), and Gascueña *et al.* (2012) describe various techniques and approaches for agent-based software to address real-world problems. We propose to use Prometheus' methodology. Prometheus comprises three phases such as system specification, architectural design, and detailed design. The architectural design phase determines system agents, interactions, and environmental events. The detailed design phase delves into agent internals, outlining functionality, processes, data structures, plans, and triggers (Winikoff & Padgham, 2004). For this project, we propose to use these tools:

- **Programming Language:** Python (or similar) for its extensive libraries in AI and web development.
- **Web Scraping:** Libraries like **BeautifulSoup** or **Scrapy** will be used for web content parsing.

- **Natural Language Processing (NLP):** Libraries like **spaCy** or **NLTK** will be crucial for tasks like information extraction, sentiment analysis, and entity recognition.
- **Natural Language Generation (NLG):** Libraries like **Gensim** or **TextBlob** can be used for data summarization and presentation.

Strengths and Weaknesses

Strengths:

- **Intelligent Search:** Considers synonyms, related concepts, and explores academic databases for comprehensive results.
- **Advanced Data Processing:** Leverages NLP for information extraction, analysis, and understanding of relationships within the data.
- **Customization:** Users can define search parameters, and data extraction rules based on their research needs.
- **Efficiency:** Automates the research process, saving researchers time and effort.
- **Data Summarization:** Provides researchers with a quick overview of key findings.

Weaknesses:

- **Website Compatibility:** This may not function effectively on websites with complex structures or anti-scraping measures.
- **Data Accuracy:** Relies on the accuracy of source websites and NLP models. Extracted data may require human review for verification.
- **Data obsolescence:** The system's reliance on pre-processed data for offline access introduces the challenge of identifying and addressing data obsolescence.
- **Limited Scope:** May not capture the full context of complex research papers, and researcher expertise remains crucial for critical analysis.

Success Criteria

The success criteria for the intelligent academic research online agent must be SMART as described by Selvik *at el.* (2021) and Bjerke & Renger (2017). Some of the points of interest are:

- The application can successfully identify relevant academic resources based on user-provided research topics, leveraging NLP to explore synonyms and related concepts.
- The application can extract data from retrieved web pages with high accuracy using web scraping and NLP techniques.
- The extracted data is processed and analyzed using NLP to understand the relationships between information points.
- The processed information is presented and stored for offline access and further research.
- The data presentation module can effectively summarize key findings and insights from the research data using NLG techniques.

Related Work

Several research tools exist to assist with online academic research. Reference management software like Zotero and Mendeley helps organize research materials. Search engines like Google Scholar and Web of Science provide access to academic publications. These tools primarily focus on literature retrieval and organization. Our application differentiates

itself by offering functionalities beyond basic retrieval. It leverages NLP to automate information extraction, processing, and analysis.

Conclusion

This proposal outlines the design and development of an intelligent research assistant application for online academic research. This application utilizes intelligent functionalities to enhance search capabilities, data processing, and data presentation. While limitations exist, this application offers researchers a valuable tool to streamline the online research process, improve efficiency, and gain deeper insights from academic data. We are confident that this intelligent approach will be a significant advancement in research workflows.

References

Jennings, N.R. (2002). Agent-Based Computing. *Intelligent Information Processing* 93(1):17–30. DOI: https://doi.org/10.1007/978-0-387-35602-0_3

Kinny, D. & Georgeff, M. (2005) Modelling and design of multi-agent systems. Intelligent Agents III Agent Theories, Architectures, and Languages 1193(1):1–20. DOI: https://doi.org/10.1007/BFb0013569

Mendeley (no date) *Datasets API quick start guide mendeley developer portal*. Available at: https://dev.mendeley.com/code/datasets_quick_start_guides.html.

Zotero | About (no date) *Zotero*. Available at: https://www.zotero.org/about.

Bjerke, M. B. & Renger, R. (2017) Being smart about writing SMART objectives. *Evaluation and Program Planning* 61(1):125-127. DOI: https://doi.org/10.1016/j.evalprogplan.2016.12.009

Gascueña, J.M., Navarro, E. & Fernández-Caballero, A. (2012) Model-driven engineering techniques for the development of multi-agent systems. *Engineering Applications of Artificial Intelligence* 25(1):159-173. DOI: https://doi.org/10.1016/j.engappai.2011.08.008

Jennings, N.R. (2000) On Agent-Based Software Engineering. *Artificial Intelligence* 117(2): 277–296. Available from: https://www.sciencedirect.com/science/article/pii/S0004370299001071 [Accessed 20 April 2024]

Kinny, D. & Georgeff, M. (2005) Modeling and design of multi-agent systems. *Intelligent Agents III Agent Theories, Architectures, and Languages* 1193(1):1–20. DOI: https://doi.org/10.1007/BFb0013569

Selvik, J. T., Bansal, S. & Abrahamsen, E.B. (2021) On the use of criteria based on the SMART acronym to assess quality of performance indicators for safety management in process industries. *Journal of Loss Prevention in the Process Industries* 70(1):104392. DOI: https://doi.org/10.1016/j.jlp.2021.104392

Winikoff, M. & Padgham, L. (2004). The Prometheus Methodology. *Methodologies and Software Engineering for Agent Systems* 11(1):217–234. DOI: https://doi.org/10.1007/1-4020-8058-1_14