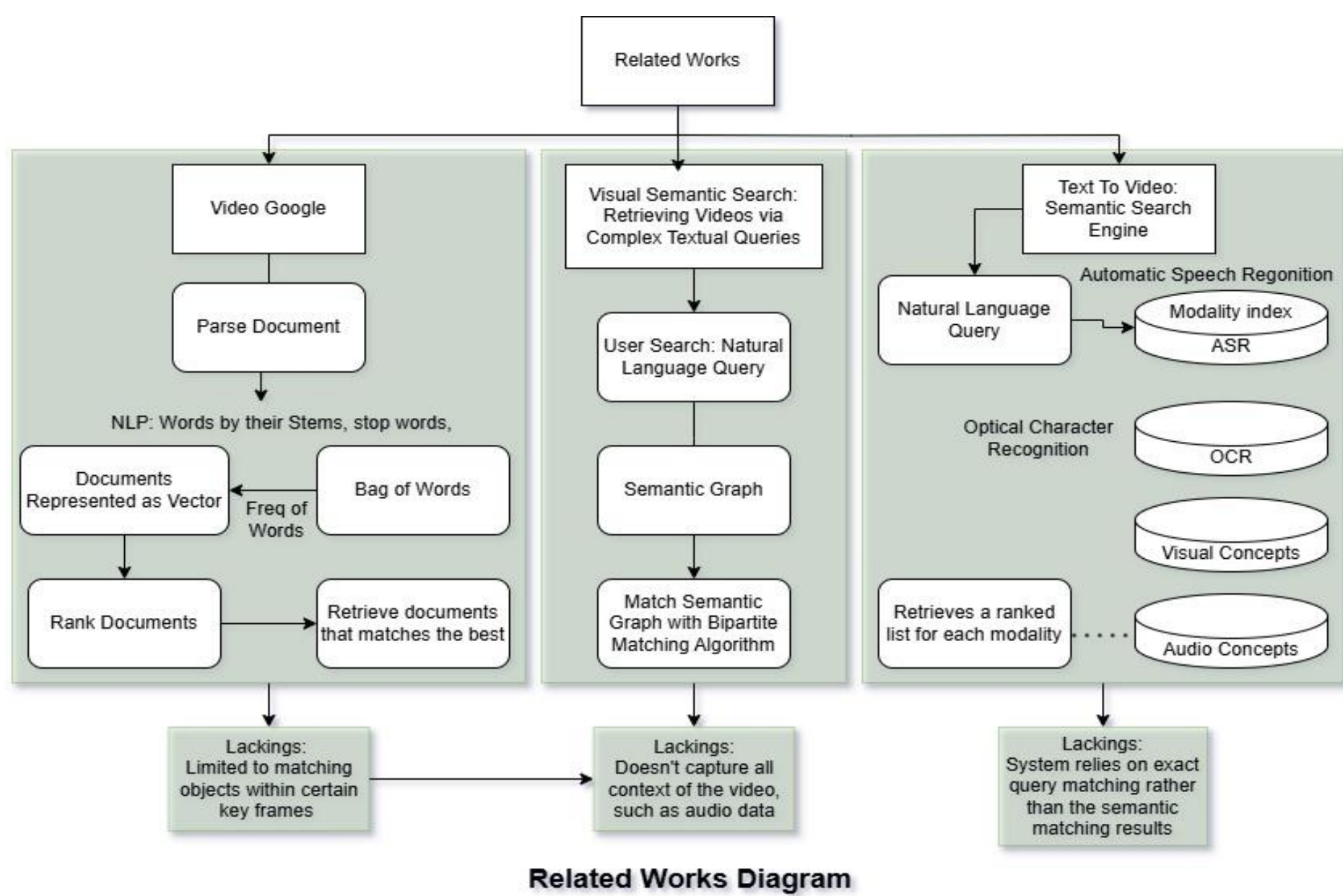


Introduction

Conventional keyword-based searches fail to provide contextual results due to the system's inability to understand the intricate elements within videos. This research focuses on the problem of retrieving video content based on user queries by utilizing large language models (LLMs) and vector embedding database. To create the AI video retrieval system, initially audio is extracted from each video. Next, we generate timestamps and corresponding textual transcriptions for each video and append relevant metadata. An LLM is then used to generate text embeddings, which are stored in a vector database for efficient retrieval. After a query is received from a web interface, the system searches the vector embedding database to retrieve semantically matched results, displaying three relevant videos for each query. This method allows users to perform semantic searches within a collection of videos, returning results that are contextually similar with the user's query.

Related Works

Traditional video search engines heavily rely on metadata, keywords, and manual tagging, which are often limited and imprecise. Several studies have been suggested to address this issue.



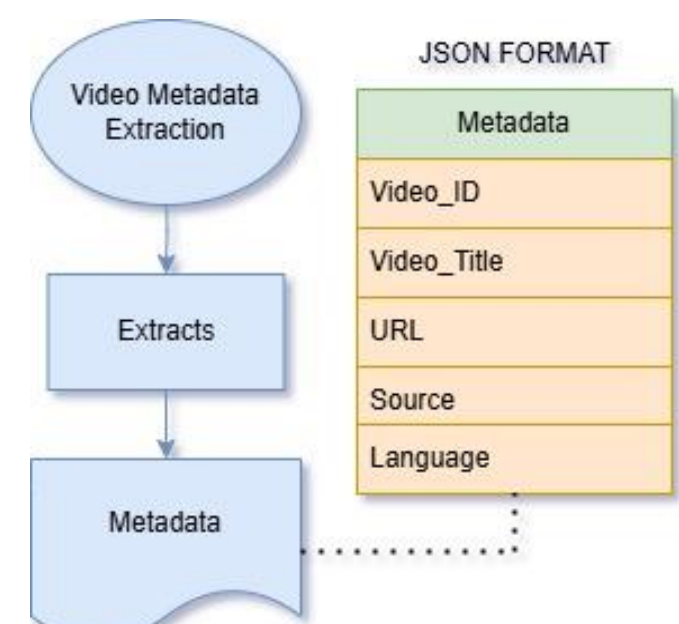
Related Works Diagram

Methodology

This section describes the methodology employed to develop the AI Video Retrieval System. The system is comprised of six components, video data ingestion and metadata extraction, audio extraction and transcription, exporting to JSON format, text embedding and indexing, query processing and retrieval, and returning search results.

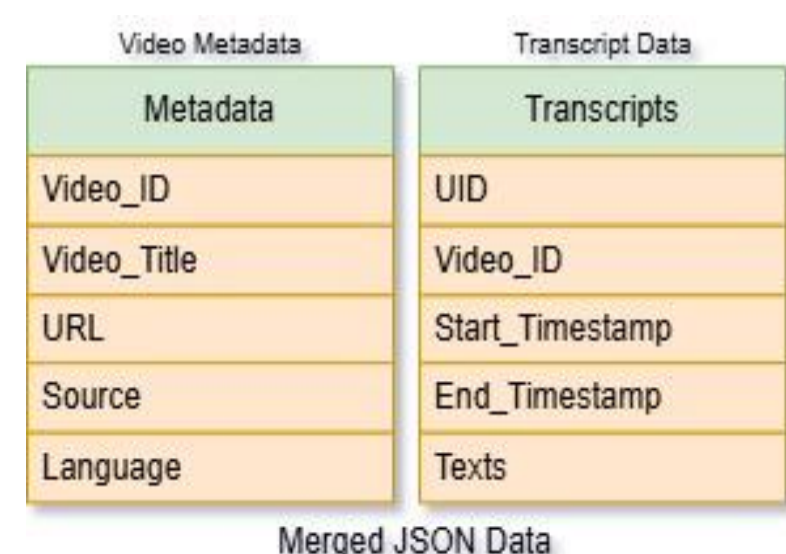


Video Metadata Extraction:

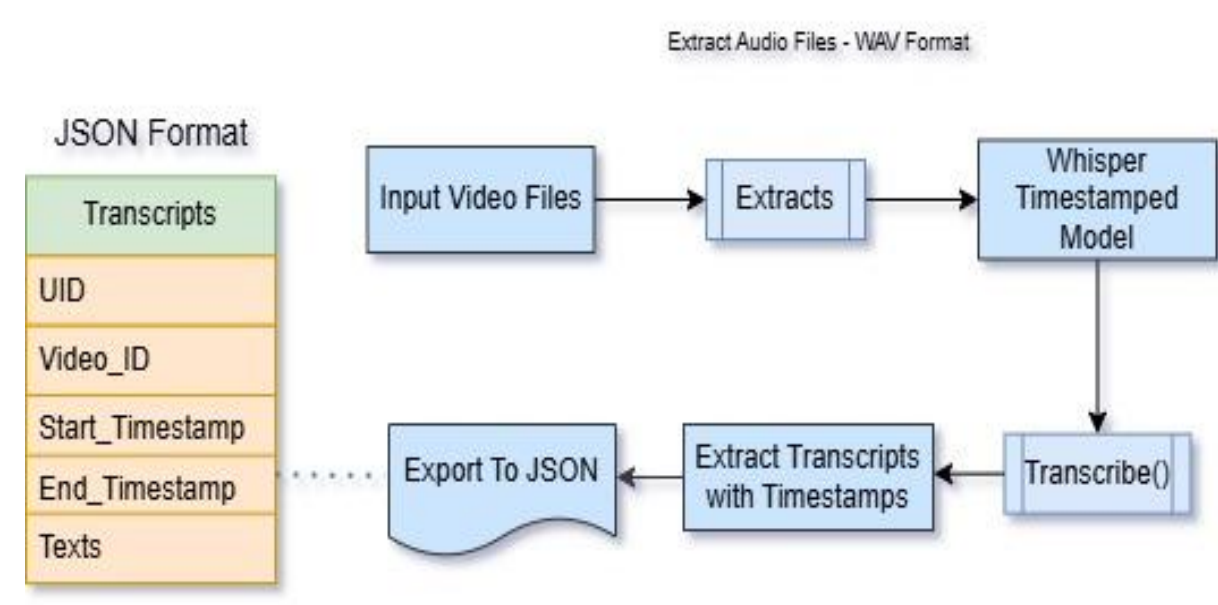


Video Data Extraction

Exporting to JSON format:

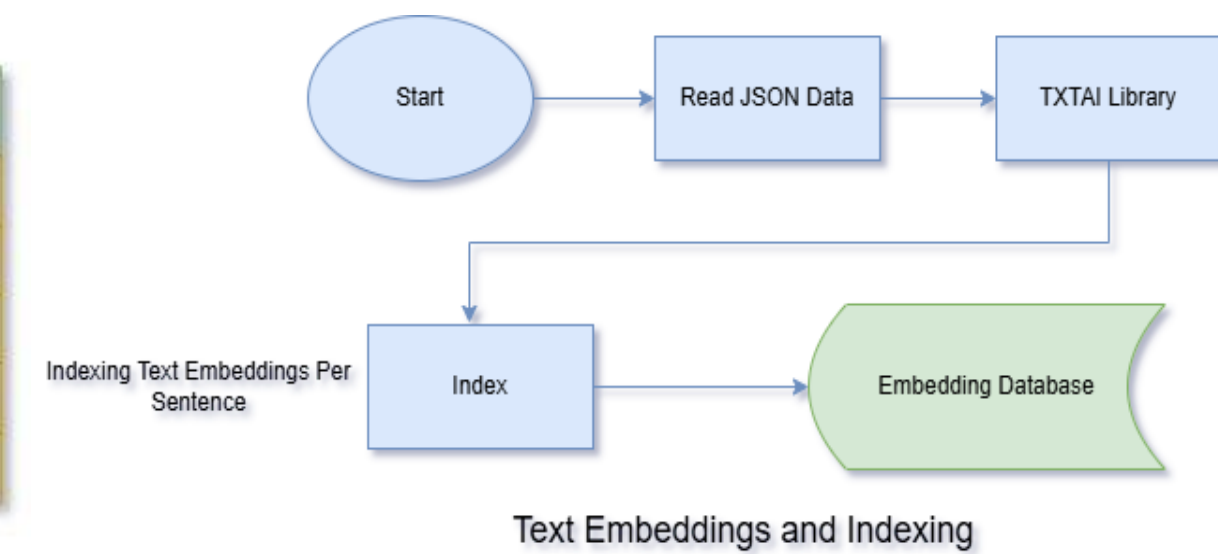


Audio Extraction & Transcription:



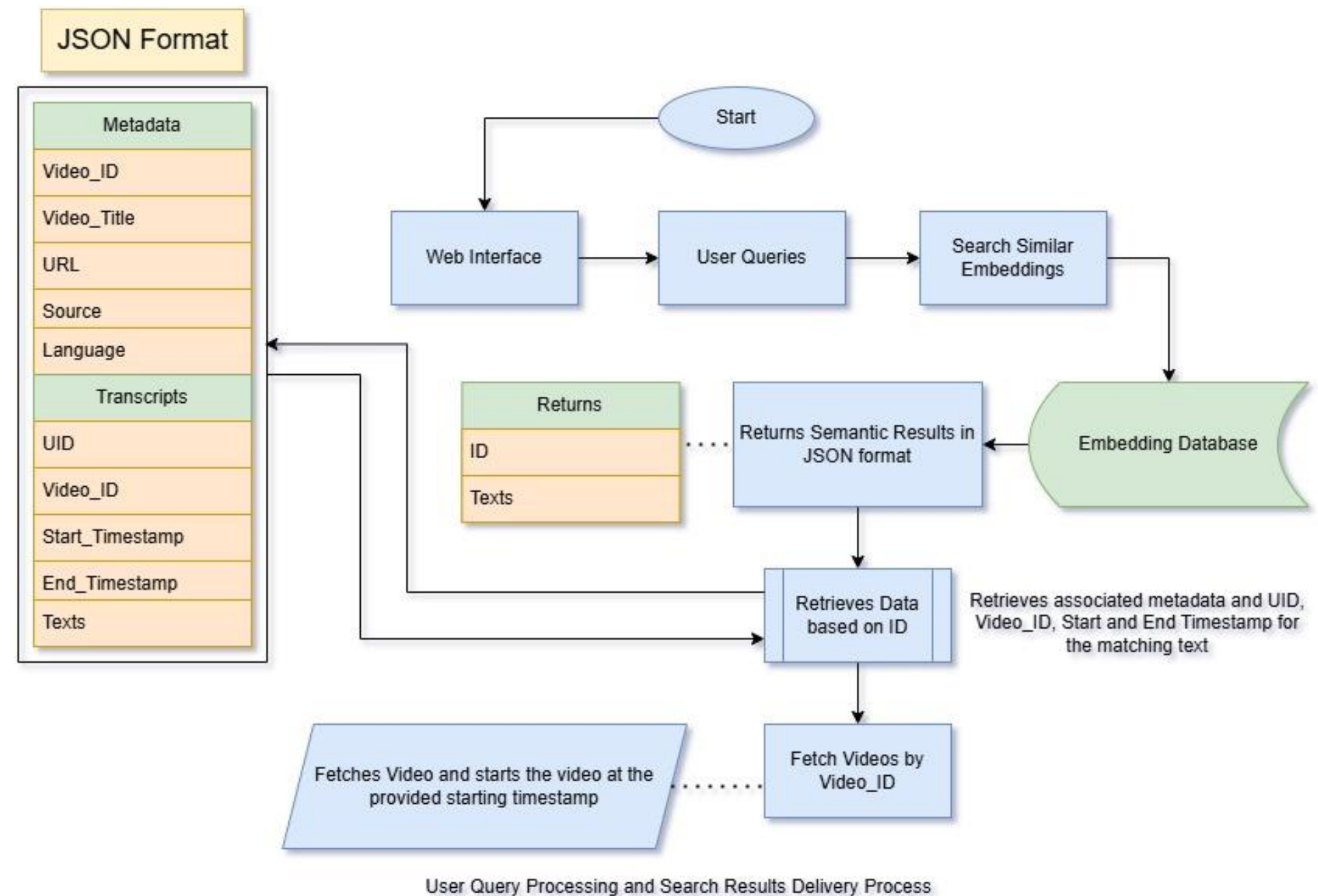
Audio Extraction

Text Embeddings and Indexing:

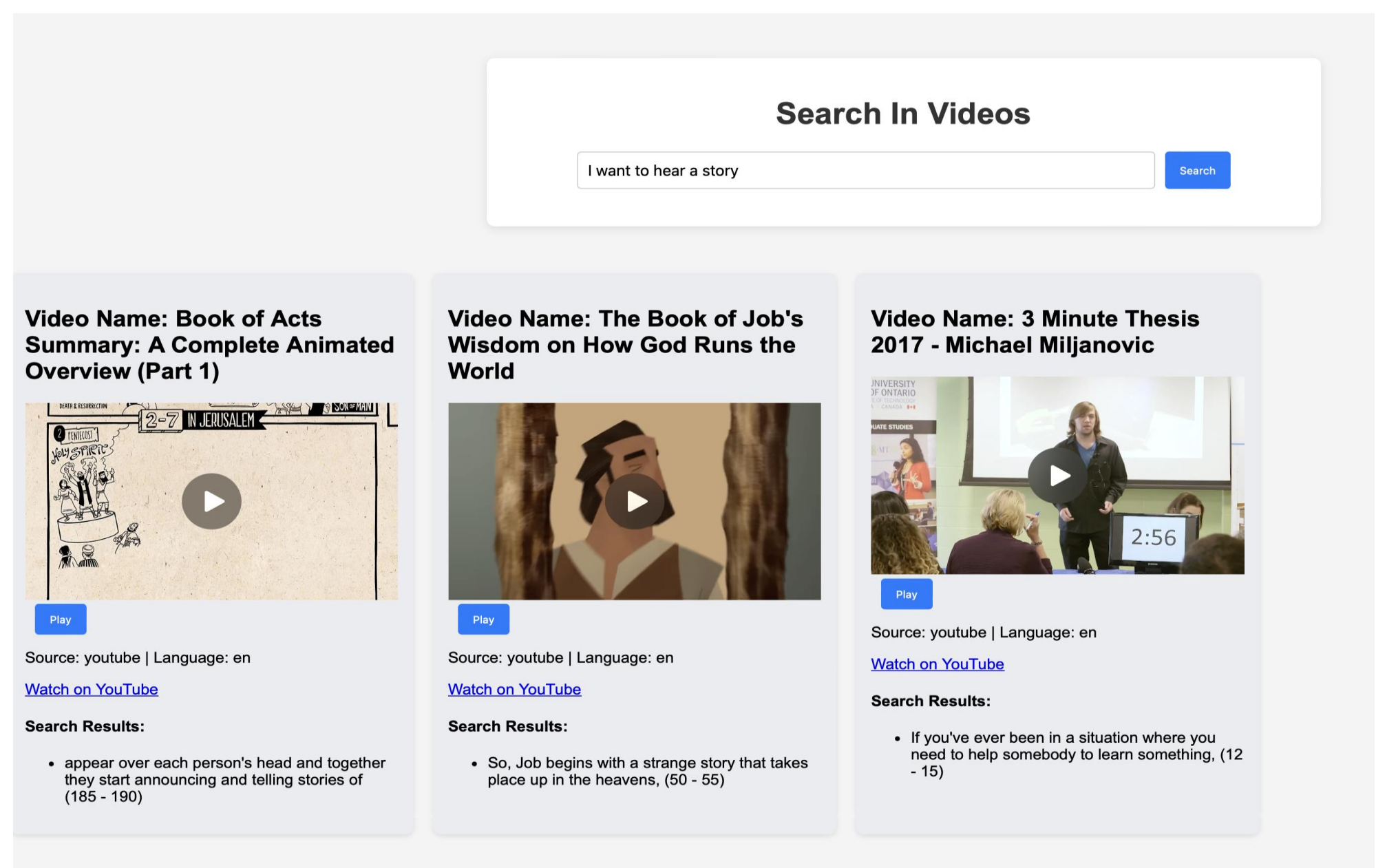


Text Embedding and Indexing: The JSON file is sent into a LLM (large language model) model to generate text embeddings using TTXAI library. TTXAI transforms each sentence into a vector representation. Later, the system compares the similarity between user's query and the database to retrieve semantic results that match.

Query Processing, Retrieval & Search Results:



Results



Search In Videos

I want to hear a story [Search]

Video Name: Book of Acts Summary: A Complete Animated Overview (Part 1)

Source: youtube | Language: en
Watch on YouTube

Search Results:

- appear over each person's head and together they start announcing and telling stories of (165 - 190)

Video Name: The Book of Job's Wisdom on How God Runs the World

Source: youtube | Language: en
Watch on YouTube

Search Results:

- So, Job begins with a strange story that takes place up in the heavens, (50 - 55)

Video Name: 3 Minute Thesis 2017 - Michael Miljanovic

Source: youtube | Language: en
Watch on YouTube

Search Results:

- If you've ever been in a situation where you need to help somebody to learn something, (12 - 15)

After performing the methodology, the following result is achieved: On the user's Natural Language query, the system converts the query into a vector embedding, then searches similar embeddings within the embedding database and returns semantically matched results and shows the user a set of videos started at the desired time.

Conclusion

The project demonstrates the successful retrieval of semantic results based on the user's natural language queries. The system utilizes large language models (LLMs) and an embedding database to store the transcripts in a vector format, which are later used to retrieve results that match the user's query. However, the system currently doesn't capture all the semantic nuances of the video, as it is limited to audio data only. Future improvements to this project include implementing real-time video indexing to return semantically matched results as new content is added. Another enhancement would be to implement image captioning on randomly selected frames of the video to capture additional contextual data. Additionally, storing action or object detection data in the embedding database could further improve the accuracy and relevance of the semantic results.

References

- [1]: J. Sivic and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," *Robotics Research Group, Department of Engineering Science, University of Oxford, United Kingdom*. Available: <https://www.robots.ox.ac.uk/~vgg/publications/2003/Sivic03/sivic03.pdf> (Accessed Oct. 26, 2024).
- [2]: D. Lin, S. Fidler, C. Kong, and R. Urtasun, "Visual Semantic Search: Retrieving Videos via Complex Textual Queries," *TTI Chicago, University of Toronto, Tsinghua University*. Available: https://www.researchgate.net/publication/264975659_Visual_Semantic_Search_Retrieving_Videos_via_Complex_Textual_Queries (Accessed Oct. 26, 2024).
- [3]: L. Jiang, S.-I. Yu, D. Meng, T. Mitamura, and A. G. Hauptmann, "Text-to-Video: A Semantic Search Engine for Internet Videos," *ResearchGate*. Available: https://www.researchgate.net/publication/288074169_Text-to-video_a_semantic_search_engine_for_internet_videos (Accessed Oct. 26, 2024).
- [4]: J. Louradour, "Whisper-timestamped," GitHub repository, 2023. Available: <https://github.com/linto-ai/whisper-timestamped>
- [5]: A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," "arXiv preprint arXiv:2212.04356", 2022.
- [6]: T. Giorgino, "Computing and visualizing dynamic time warping alignments in R: The dtw package," "Journal of Statistical Software", vol. 31, no. 7, 2009, doi: 10.18637/jss.v031.i07.
- [7]: D. Mezzetti, "txtai," GitHub repository. Available: <https://github.com/neuml/txtai>