

Development Track: A YouTube Extension with Dynamic Keyword Generation

Zongxian Feng(zfeng13), Chenxin Wang(cw110), Andres Suazo(suazo2)

Functions and Users

Our project aims to develop a specialized Chrome browser extension designed to revolutionize the way users discover content. It targets YouTube as its sole platform in order to demonstrate the concept in action. This tool dynamically generates contextual keyword recommendations by analyzing user interactions, such as scrolling through comments and video recommendations, alongside provided metadata like descriptions, transcripts, and subtitles. This approach ensures that recommendations are tailored to the user's current engagement and interests.

Our target users can be categorized into two main groups:

Content Viewers: Casual users of the platform who find themselves navigating through the immense amount of content available. They would benefit from a more tailored experience by having the tool recommend content that matches the viewer's interests or even aid in the discovery of new areas that might be of interest.

Content Creators: Owners of the content that is being distributed by the platform. Insights into viewer preferences and tendencies provide creators with a better way to engage with their audience as they would produce new content more aligned with the current interests of the audience.

Significance

Our extension addresses a critical gap in YouTube's content discovery process by providing personalized keyword recommendations that adapt in real-time to user behavior. This dynamic approach is crucial in a digital landscape where users expect highly personalized experiences. It not only enhances user engagement by making content discovery more intuitive but also increases the visibility of creators' content by aligning recommendations with viewers' immediate interests.

Unique Feature: Dynamic, Behavior-Based Keyword Recommendations

What sets our extension apart is its ability to dynamically adjust keyword recommendations based on real-time analysis of user interactions, particularly focusing on how users engage with comments. By prioritizing keywords from the user's current viewport, we hypothesize that our tool will significantly enhance the relevance of search suggestions, closely aligning with users' immediate interests.

Approach

To implement this innovative feature, we will:

1. Employ advanced Natural Language Processing (NLP) techniques, with a focus on Attention mechanisms, to analyze text content that users interact with in real-time.
2. Develop a behavior tracking mechanism within the extension that identifies and analyzes key user interactions, such as scrolling through the comments section.

3. Ensure that the extension's operations respect user privacy, with all data analysis performed locally on the user's device.

Our development stack will include Python for backend processing, utilizing Huggingface for NLP tasks and JavaScript for the extension's frontend and behavior tracking functionalities.

Evaluation

We plan on using a combination of methods in order to better evaluate the effectiveness of the tool. Our primary method will be A/B testing with the goal of determining if our tool resulted in increased user interactions with the search and higher engagement with recommended content compared to YouTube's standard recommendations. We will randomly divide a set of users, and have one set use the platform as is, while the other uses the platform with our extension integrated into it. The main indicator we will monitor will be click through rates on recommended content, overall time spent on the platform and user interactions with the content (likes, comments, etc). An optional additional indicator we might include in our evaluation could be the diversity of content in videos seen by the user to determine if exploration was facilitated by the tool.

Timeline

Week 1: Initialization

- Define project scope and MVP features.
- Set up development environments for NLP (Python, TensorFlow) and the extension (JavaScript).

Week 2: NLP Model Development

- Develop a basic NLP model to extract keywords from YouTube metadata and comments.
- Begin model training with initial datasets.

Week 3: Extension Development

- Create the browser extension frontend to display keywords.
- Integrate the NLP model with the extension backend.

Week 4: Testing and Refinement

- Conduct tests and gather feedback.
- Refine the demo for core functionality and usability.

Task Divison

Zongxian: Develops and trains the NLP model for keyword extraction.

Chenxin: Handles frontend development of the browser extension.

Andres: Manages backend integration and connects the NLP model with the frontend.