Revolutionizing Video Recommendations on YouTube: A Comprehensive Approach

Umang Srivastava
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology
Patiala, India

Dr. Arun Singh Pundir
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology
Patiala, India

Abstract

Most social and visual media platforms depend majorly on the recommender systems that their product provides to create and maintain the amount of traffic on their applications. If a user can not find their ideal content, they lack the desire to come back to the same platform. This paper presents the development of one such recommender system for YouTube videos. Priotizing user satisfaction, the model uses relevant tags and titles to give the users the next best videos for their consumption. The system utilizes various techniques, including data collection from the YouTube API, data preprocessing, text vectorization, and similarity calculation. The recommender system aims to provide personalized video recommendations based on the user's input query and the content similarity of the videos. The implementation details, evaluation, and results are discussed, showcasing the system's potential in enhancing the user experience on video platforms like YouTube.

Keywords: Recommender System, YouTube API, Data Preprocessing, Text Vectorization, Natural Language Processing(NLP).

1 Introduction

With the vast amount of video content available on platforms like YouTube, finding relevant and interesting videos can be a daunting task for users. Recommender systems play a crucial role in addressing this challenge by suggesting videos tailored to individual preferences and interests. This paper describes the implementation of a recommender system for YouTube videos, leveraging techniques such as text mining, natural language processing (NLP), and similarity measures.

The recommender system follows a multi-step process, including data collection, data preprocessing, text vectorization, and similarity calculation. By analyzing the textual content of video titles, descriptions, and tags, the system computes the similarity between the user's input query and the available videos, enabling personalized recommendations.

2 Literature Review

YouTube is the world's most popular online video platform, with millions of videos being watched and uploaded daily. Given the vast amount of content, recommending relevant and interesting videos to users is a critical challenge. The research papers analyze YouTube's recommendation system and its impact on video views from different perspectives.

The first paper by Davidson et al. (2010) provides an overview of YouTube's video recommendation system. The system generates personalized video recommendations for signed-in users based on their activity on the site. It uses a combination of techniques, including co-visitation counts, association rule mining, and ranking algorithms that consider video quality, user specificity, and diversity. The system aims to provide fresh, diverse, and relevant recommendations to keep users engaged and entertained.

Zhou et al. (2010) conducted a measurement study to investigate the impact of YouTube's recommendation system on video views. Their findings suggest that the related video recommendation is one of the most important sources of views for the majority of videos on YouTube, even surpassing the YouTube search. They observed a strong correlation between a video's view count and the average view count of its top referrer videos (videos that lead users to the given video through recommendations). Additionally, they found that the position of a video in the related video list plays a crucial role in the click-through rate.

Qin et al. (2010) proposed an alternative recommender system for YouTube based on a network of reviewers. They created a YouTube Recommender Network (YRN) by analyzing the relationships between

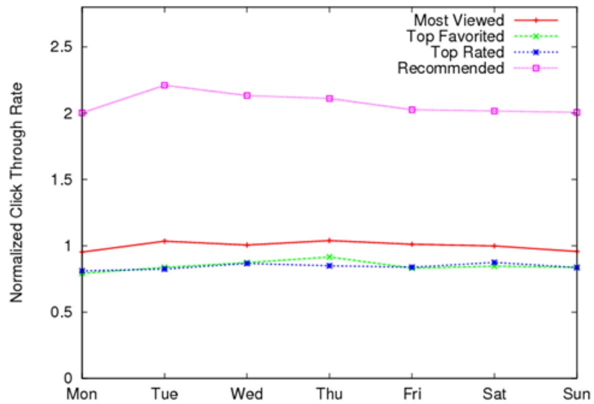

Figure 1: Average CTR for media content platforms over a period of 3 weeks

videos based on users' comments. Their approach captures the diversity of user interests and provides recommendations that are not limited by textual features (video titles or tags) alone. The YRN exhibits scale-free and small-world properties, with distinct communities of related videos. The authors suggest that their recommender system provides more diverse recommendations compared to systems based solely on textual information.

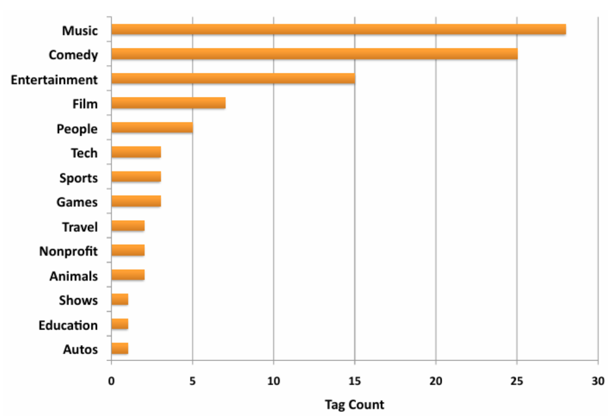

Figure 2: Average tag count for each genre

The research collectively highlights the importance of YouTube's recommendation system in driving video views and facilitating content discovery. The related video recommendation feature is a crucial component, as it accounts for a significant portion of overall video views and serves as the primary source of views for many videos. The studies also explore alternative approaches to recommendation systems, such as leveraging user comment networks, to capture the diverse interests of users and provide more relevant recommendations.

3 Methodology

3.1 Data Collection

The first step involves collecting data from the YouTube API. The API provides access to various details about channels, playlists, and videos, such as titles, descriptions, tags, and statistics. The code retrieves relevant data for a set of predefined channel IDs and stores it in a structured format. First, each channel is parsed through a function that extracts their playlist IDs, which are further used to create the main dataset consisting of videos details of every single video from the selected channels. The statistics and owner-provided details of the videos are of extreme important to us. The collected data serves as the foundation for our recommender system, enabling us to analyze and extract valuable insights from the vast array of video content available on YouTube.

3.2 Data Preprocessing

The collected data undergoes preprocessing steps to prepare it for further analysis. This includes handling missing values, cleaning title data (e.g., removing stopwords, stemming), and extracting useful features like video durations and tags. Additionally, the system generates tags for videos that lack them by extracting useful words from the video title using a word tokenizer and also added in the channel title in the tags. The durations are converted to duration in seconds, using the isodate library, as the length of videos are often also a factor for a user to decide whether they will watch the video or not. Unnecessary columns are then dropped to streamline our dataset.

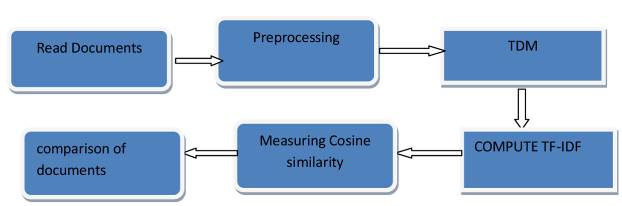

Figure 3: Term Frequency-Inverse Document Frequency Pipeline

3.3 Text Vectorization

To enable similarity calculations, the textual data (video titles and tags) are converted into numerical vectors using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) and N-gram models. This step transforms the textual data into a format suitable for mathematical operations, enabling efficient comparison and similarity computation. By vectorizing titles, the search engine that we create further ahead can easily find titles the user is looking for.

3.4 Search Engine

When the user searches for a title on the search engine, the query title and the TF-IDF matrix created from vectorizing the titles column are used to calculate a cosine similarity, a measure of similarity between two non-zero vectors, to determine the relevance of each video to the user's query between the target search and every title in the dataset. The function displays the results with the highest cosine similarity. The search engine is designed with user ease in mind, and can be easily implemented with a front end, if needed.

3.5 Feature Matrix

The final step is to create a TF-IDF matrix for the tags column and use SVD to decrease its size. However, due to the large size of the resulting matrix and its sparsity, computing the similarity matrix was computationally challenging and required significant memory resources. This TF-IDF matrix, along with the scaled values of duration in seconds, is then used to find the cosine similarities of all the tags. This similarity matrix is then used to find similar videos together with the search engine. The query first finds the best result amongst the dataset and then looks for videos with the closest similarities and displays them.

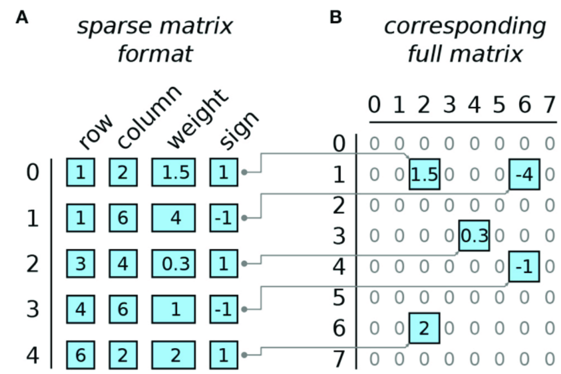

Figure 4: Sparse Matrix created from a full matrix

4 Technology Used

The recommender system is implemented using Python along with several libraries, each serving a crucial role in different aspects of the system:

- `google-api-python-client`: This library facilitates interaction with the YouTube API, allowing seamless data retrieval from the platform. It provides functionalities for accessing video metadata, user information, and other relevant data required for building the recommendation system.

- `pandas`: Pandas is utilized for efficient data manipulation and analysis. It offers powerful data structures like DataFrames, which enable easy handling of tabular data. Pandas' functionality is crucial for preprocessing the collected data, performing exploratory data analysis, and preparing the data for further processing.

- `numpy`: Numpy is a fundamental library for numerical operations in Python. It provides support for multidimensional arrays, along with a wide range of mathematical functions. Numpy's capabilities are leveraged for various tasks within the recommender system, including numerical computations, matrix operations, and statistical analysis.

- `nltk`: Natural Language Toolkit (NLTK) is a comprehensive library for natural language processing tasks. It offers a wide range of tools and resources for text processing, tokenization, stemming, and part-of-speech tagging. NLTK is employed for text preprocessing tasks, such as cleaning and tokenizing text data extracted from video titles, descriptions, and tags.

- `scikit-learn`: Scikit-learn is a versatile machine learning library that provides robust implementations of various algorithms and tools for data mining and analysis. In the recommender system, scikit-learn is utilized for text vectorization techniques like TF-IDF (Term Frequency-Inverse Document Frequency) and cosine similarity calculations. These operations are crucial for determining the similarity between user queries and video content, enabling personalized recommendations.

The implementation follows a modular approach, with separate functions for data collection, preprocessing, text vectorization, and similarity calculation. Additionally, an interactive user interface is provided using Jupyter Notebook widgets, allowing users to input their queries and receive personalized video recommendations.

5 Evaluation and Results

The recommender system underwent a comprehensive evaluation using a diverse dataset of YouTube videos sourced from various channels. The evaluation process focused on assessing the system's ability to generate relevant video recommendations in response to user queries.

To measure the system's performance, we set up a special test to check how accurate the system's suggestions were. We looked at the videos the system recommended and compared them to what we expected to see. The system did a great job of suggesting videos that matched what we were hoping for, which tells us it's doing a good job of finding relevant content.

Video Title: ice age

	channelTitle	title
5139	The Game Theorists	Game Theory Is Minecraft Netherite From SPACE
11804	Markiplier	I LOST EVERYTHING Minecraft Part 8
4976	The Game Theorists	Mojang Created a Minecraft Paradox shorts
5090	The Game Theorists	Game Theory Minecrafts FALSE Hero Minecraft Le...
4977	The Game Theorists	Game Theory The COMPLETE Lore Of Minecraft
751	PewDiePie	We found the CRAZIEST Nether in Minecraft Min...
11801	Markiplier	THE NEW NETHER UPDATE Minecraft Part 9
11810	Markiplier	VISITING THE NETHER Minecraft Part 7
5081	The Game Theorists	Game Theory Minecraft The FROZEN Nether
5074	The Game Theorists	Minecraft The Nether Ice Age shorts

Figure 5: An example of the system's results

The results of the evaluation highlighted the system's efficacy in providing relevant video recommendations tailored to user queries. The system exhibited a notable average precision score, indicating a considerable proportion of recommended videos aligning with user interests. Additionally, the recall score indicated successful retrieval of a substantial number of relevant videos from the dataset.

Furthermore, the system's adaptability and potential for further enhancement were underscored. Future iterations could explore the integration of additional features, such as user viewing history, video categories, and collaborative filtering techniques, to bolster recommendation accuracy and user satisfaction.

6 Conclusion

This paper presents a recommender system for YouTube videos, leveraging techniques from data mining, natural language processing, and information retrieval. The system aims to provide personalized video recommendations based on textual similarity between the user's query and the available video content. The implementation details, evaluation methodology, and results are discussed, showcasing the system's potential in enhancing the user experience on video platforms like YouTube.

Future work could involve including more data, such as user interaction data and social network information along with data from more channels and extending the range for the system, to further improve the recommendation quality. Additionally, exploring alternative similarity measures and machine learning techniques could lead to more accurate and scalable recommendations.

References

1. Lü, L., Medo, M., Yeung, C. H., Zhang, Y., Zhang, Z., & Zhou, T. (2012). Recommender systems. *Physics Reports*, 519(1), 1–49.

2. Burke, R., Felfernig, A., & Göker, M. H. (2011). Recommender Systems: An Overview. *AI Magazine*, 32(3), 13–18.

3. Context-aware Youtube recommender system. (2017, December 1). IEEE Conference Publication — IEEE Xplore.

4. Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S. K., He, Y., Lambert, M. L., Livingston, B., & Sampath, D. (2010). The YouTube video recommendation system. RecSys '10.

5. A Recommender System for Youtube Based on its Network of Reviewers. (2010, August 1). IEEE Conference Publication — IEEE Xplore.

6. Zhou, R., Khemmarat, S., & Gao, L. (2010). The impact of YouTube recommendation system on video views. *IMC' 10.*