Sapphire: An NLP based YouTube video scoring model.

Srinidhi Srujan Murthy

President, AFD Enterprises

This whitepaper introduces Sapphire, a novel ranking model designed to evaluate YouTube videos based on the comprehensive analysis of their transcripts' corpus. By employing regex operations and identifying the most unique and significant keywords throughout the video content, Sapphire offers a more analytical approach to evaluation, considering the relative importance of individual terms. The primary objective of Sapphire is to address the challenges associated with ranking transcripts based on their rigor, independently of video viewership, which is the conventional approach adopted by the YouTube Watch Time algorithm[1] . Additionally, Sapphire includes transcription based on unique identifier keyword weighting strategies. This paper details Sapphire, exploring key components such as YouTube transcription, text preprocessing, Term Frequency-Inverse Document Frequency (TF-IDF) evaluators, and score assessments.

1 Introduction

1.1 Market Background

The education and research industry has a market capitalization of $800 Billion that relies on access to high-quality information. According to a CTE, Jenny Arledge, "Technology is the 'wings' that enable education to soar higher and faster than ever before, transforming the learning landscape if we fully embrace its potential"[2]. However, finding the best learning materials can be time-consuming and inefficient.

Srinidhi Srujan Murthy: srujanm@afd.enterprises

1.2 Enhancing Content Evaluation

Sapphire recognizes that information rigor is a crucial factor in facilitating effective learning. Unlike traditional approaches that rely primarily on viewership metrics[3], Sapphire employs a sophisticated algorithm to evaluate the quality and relevance of YouTube videos through token frequency distributions and TF-IDF scoring. By doing so, Sapphire enables users to identify the resources with the highest amount of content rigor.

1.3 Tailored for Specific Situations

While algorithms like the watch time algorithm are well-suited for video-sharing platforms optimizing for user retention, they lack the ability to source videos based on the depth of the information[4].

1.4 Growing Market Potential

The education sector is witnessing a remarkable rise in demand for advanced information ranking models like Sapphire, driven by the shift towards digital learning resources[5]. Inspired by the successes of algorithms such as Google's PageRank[6] and YouTube's recommendation system, these models are becoming increasingly vital in facilitating personalized learning experiences.

2 Problems

2.0.1 Limitations of YouTube's watch time algorithm

YouTube's watch time algorithm, a critical determinant of video visibility and success on the platform, exhibits noteworthy limitations that warrant consideration in the context of content creation and consumption. The algorithm tends to put poor emphasis on the content covered, focus on the main topic, and place quantity over quality in viewership statistics[4].

2.0.2 Emphasis on quantity over content quality

A primary concern revolves around the algorithm's predisposition towards quantity, potentially overshadowing the importance of content quality. With an inherent bias towards videos with longer watch times, content creators may be inclined to prioritize video length or employ clickbait strategies to artificially inflate views[7].

2.0.3 Compromising content depth for metrics

Consequently, this emphasis on metrics may inadvertently foster an environment where the pursuit of quantity overshadows a commitment to crafting substantive, valuable content. The allure of longer videos and attention-grabbing titles, even at the expense of content depth, may compromise the overall quality of user experiences on YouTube[8].

2.1 Solution

An algorithm for determining the weight of each term used in a corpus, excluding stopwords, would provide a more rational and content-rigor based approach to ranking and scoring. By incorporating TF-IDF, lexical analysis, and tokenization, this algorithm can score content rigor[9].

Legend:
S - Sapphire score
C - corpus text
w - Token in corpus
n - number of tokens
f - word Frequency

$$S(c) = \frac{\sum_{i=0}^{n_f} f(w_c) * TF - IDF((w_c)_i)}{n_c} \tag{1}$$

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) \times \log\left(\frac{N}{\text{df}(t, D)}\right) \tag{2}$$

$$P(w_n|w_1^{n-1}) = \frac{C(w_1^n)}{C(w_1^{n-1})} \tag{3}$$

3 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) scoring is a numerical representation used in information retrieval and text mining. It helps to determine the importance of a term in a document or a corpus. TF-IDF scoring considers a term's term frequency (TF) and inverse document frequency (IDF). TF measures how often a term appears in a document, while IDF measures how important a term is across a corpus. The TF-IDF score of a term increases with its frequency in a document and decreases with its frequency across the entire corpus. This helps identify terms specific to a document and can be used for indexing and ranking[10].

3.0.1 Lexical Analysis

Lexical analysis involves breaking down a sequence of characters or words into smaller units called tokens. These tokens can be words, phrases, or even individual characters. On the other hand, N-gram models focus on analyzing sequences of N words and their probabilities of occurrence[11].

These are the essential components to creating a mathematical entity out of every word recognized in a given transcript. It allows us to capture the emphasis and contextual significance of every word mentioned relative to the others in the provided bag of words[12].

4 Requirements

4.0.1 YouTube transcription

In Sapphire, the utilization of YouTube Transcription plays a crucial role as it relies on the transcription of a video to gain insights into its content. By analyzing the words and phrases present in the transcription, Sapphire gains a deeper understanding of the topic and subject matter of the video[13].

Furthermore, YouTube Transcription helps Sapphire assess the content used within a video. This information is vital in determining the overall rigor, indicating whether viewers find the video informative.

4.0.2 Text pre-processing

Text preprocessing is a step in natural language processing (NLP) that involves cleaning and transforming unstructured text data to prepare it for analysis[14]. The main techniques used in text preprocessing include:

Lowercasing: Converting all text to lowercase helps ensure consistency and avoids duplication of words due to case differences[15].

Removing Special Characters: Special characters, such as punctuation marks and symbols, are often removed to eliminate noise and simplify the text[16].

Tokenization: Tokenization involves splitting the text into individual words or phrases, known as tokens. This step is required for further analysis, such as TF-IDF analysis[17].

Removing Stopwords: Stopwords are common words that do not carry significant meaning, such as "and," "the," and "is." Removing stopwords helps reduce noise and focus on important words[18].

Stemming and Lemmatization: Stemming and lemmatization are techniques used to reduce words to their base or root form. This helps consolidate similar words and reduce the vocabulary size[19].

These preprocessing techniques ensure that the text is cleaned, standardized, and ready for analysis.

4.0.3 Importance of Text Preprocessing

Text pre-processing is crucial in various NLP tasks, including sentiment analysis, text classification, and information retrieval. By cleaning and transforming the text data, it becomes easier to extract meaningful insights and patterns from the data [20].

4.0.4 Selection of Relevant Transcript Portion

The length of the video's transcript should be considered to determine the portion of the video that should be used for evaluation. Optimal bounds are provided to identify the most critical part of the transcript. When the video's time length exceeds 5 hours, it scrapes the middle portion of the transcript to extract the most insightful content of the video. This ensures that the analysis focuses on the relevant content and avoids unnecessary information.

4.0.5 Vocabulary Weighting

After selecting the relevant portion of the transcript, vocabulary weighting is applied to assess the significance and impact of the text[21]. Vocabulary weighting techniques, such as TF-IDF analysis, assign weights to words based on their frequency and importance in the document or corpus. This helps identify key terms and their relevance to the overall content.

4.1 Example of TF-IDF Calculation

To illustrate the TF-IDF calculation process, let's consider an example. Suppose we have a YouTube video about "Introduction to Machine Learning." The transcript of the video is preprocessed and tokenized into sentences. The algorithm then analyzes the words within these sentences and assigns TF-IDF scores to each word based on their frequency and rarity within the transcript. For instance, the word "machine" appears frequently throughout the transcript, indicating its high term frequency (TF). However, if the word "learning" appears less frequently in the transcript, it will have a lower TF. On the other hand, if the word "machine" appears in many other videos' transcripts, it will have a lower inverse document frequency (IDF), indicating its commonality. Conversely, if the word "learning" appears in fewer videos' transcripts, it will have a higher IDF, indicating its rarity. By combining the TF and IDF scores, the algorithm assigns a TF-IDF score to each word. Words with higher TF-IDF scores, such as "learning," are considered more important and relevant to the content of the video, while words with lower scores, such as "machine," are considered less significant.

5 Score assessment

The assessor's primary responsibility lies in filtering words based on their TF-IDF score.

The assessor then establishes an average standard for the word weight score, acting as a threshold to distinguish valuable words from unnecessary ones. Words scoring higher than the average are retained for further analysis, while those equal to or below the average are discarded.

The Keyword Qualifier plays a critical role in determining the average score. The Keyword Qualifier ensures that only words meeting or surpassing the average score are considered for further analysis, while words below the average are eliminated from consideration.

This process eliminates superfluous words, enabling the scoring process to focus exclusively on words with value and relevance.

References

[1] Ruohan Zhan, Changhua Pei, Qiang Su, Jianfeng Wen, Xueliang Wang, Guanyu Mu, Dong Zheng, and Peng Jiang. Deconfounding duration bias in watch-time prediction for video recommendation, 2022. [Online; accessed 2023-04-09].

[2] Michelle C. Singh. Technology creates diverse learning opportunities for all students. *LinkdIn*, 2023.

[3] Lori Nishiura Mackenzie. Why most performance evaluations are biased, and how to fix them, Jan 11 2019. [Online; accessed 2024-03-10].

[4] Uttaran Samaddar. Youtube algorithm hack: Chase watch time and engagement — not views. *VidIQ*, Sep 25 2023. [Online; accessed 2024-01-23].

[5] Eric Roberts. The google pagerank algorithm, Nov 9 2016. [Online; accessed 2022-11-15].

[6] Sergey Brin and Larry Page. The pagerank citation ranking:bringing order to the web, Jan 29 1998. [Online; accessed 2023-11-15].

[7] Matthew Fyfeld. Navigating four billion videos: teacher search strategies and the youtube algorithm, Jun 09 2020. [Online; accessed 2024-01-27].

[8] Sasha Lerman. What is more important: the quality or quantity of content on youtube? *Prodvigate*, Dec 21 2023. [Online; accessed 2024-01-27].

[9] Liu Cai-zhi, Sheng Yan-xiu, Wei Zhi-qiang, and Yang Yong-Quan. Research of text classification based on improved tf-idf algorithm, Aug 24-27 2018. [Online; accessed 2023-06-19].

[10] Anirudha Simha. Understanding tf-idf for machine learning. *CapitalOne*, Oct 6 2021. [Online; accessed 2024-01-27].

[11] Daniel Jurafsky and James H. Martin. N-gram language models. Whitepaper, Jan 7 2023. [Online; accessed 2023-05-8].

[12] V. Sanju Farhanaaz. An exploration on lexical analysis, March 5 2016. [Online; accessed 2024-01-27].

[13] Yanmeng Liu. Exploring a corpus-based approach to assessing interpreting quality, April 11 2021. [Online; accessed 2024-01-27].

[14] Deepanshi. Text preprocessing in nlp with python codes. *Analytics Vidhya*, April 26 2023. [Online; accessed 2024-01-27].

[15] Max Ved. Text preprocessing for nlp and machine learning tasks. *Experfy*, Nov 17 2023. [Online; accessed 2024-01-27].

[16] Chetna Khanna. Text pre-processing: Stop words removal using different libraries. *Towards Data Science*, Feb 10 2021. [Online; accessed 2024-01-27].

[17] Abid Ali Awan. What is tokenization? *Data Camp*, Sep 2023. [Online; accessed 2024-01-27].

[18] Jacob Murel. What are stemming and lemmatization?, Dec 10 2023. [Online; accessed 2024-01-27].

[19] T. Nishanthy S. Arudchutha and R.G. Ragel. String matching with multicore cpus: Performing better with the aho-corasick algorithm. Whitepaper, March 5 2014. [Online; accessed 2023-05-28].

[20] Indeed. What is data preprocessing? (with importance and examples), March 12 2023. [Online; accessed 2024-03-10].

[21] Duan Longjiang. Test of english vocabulary recognition based on natural language processing and corpus system, April 21 2021. [Online; accessed 2024-01-23].