YouTube Semantic Search

Annabella Rinaldi [*] **Kimberly Brown** [*] **Shivani Pandey** [*] **Mekaiel Khan** [*]

Abstract

We present *YouTube Semantic Search*, an AI-driven system that discovers educational YouTube videos based on the meaning of user prompts rather than keyword matches. By embedding queries, LLM-generated answers, and video metadata into a shared semantic space using a transformer-based language model (Sentence-BERT), we compute cosine similarities to rank and return the most relevant videos. Our prototype focuses on STEM subjects—mathematics, algorithms, biology, chemistry, and physics—and integrates both OpenAI answer integration and a lightweight front end displaying video recommendations.

1. Introduction

Earlier search engines widely used on the Internet, including YouTube's native search, perform keyword-based matching that fails to capture the nuances of language and context in user queries. They do not interpret long sentences with clauses or explanations, instead returning results based on exact word matches. As a result, students often spend considerable time sifting through irrelevant videos to find content that truly addresses their question.

For our project, we develop an AI model called YouTube Semantic Search that discovers educational videos based on user prompts or questions without relying solely on direct keyword matches. We represent text as vectors of numbers encoding meaning, then match these vectors rather than words. Through our research and experience, we found that students frequently struggle to locate high-quality videos efficiently. Our semantic search finds the top matching videos that are most useful and relevant to the user's query.

1.1. Keyword-based Search

Conventional keyword search engines are very useful for finding information on the Internet with matching keywords. But when certain words have multiple meanings, it can be difficult for search engines to make these connections and produce the desired results.

Keyword-based search has low precision (ration of documents retrieved that are relevant to the user's information need) and recall (ratio of the documents relevant to the query that are retrieved). It also does not capture the full meaning behind polysemy words (words that have several distinct meanings) and synonymy words (several words that have the same meaning, but nevertheless cannot be matched when a keyword-based search is used). With keyword-based search, informational retrieval technology is based on the occurrence of words in documents which does not necessarily yield the most relevant results.

When traditional keyword search is used to find videos, we must

1. Tokenize the query into words.
2. Compute TF–IDF weights for each term in video titles and descriptions.
3. Rank results by total matching TF–IDF scores.

This method returns only videos containing the same tokens and will miss contextual matches that would otherwise result in more relevant videos.

1.2. Semantic-based Search

Semantic-based search engines are able to intelligently understand the context and meaning of a user's query and retrieve relevant results based on semantic matching. Semantic search engines are capable of storing semantic information and solving complex queries.

We choose to use semantic search because it allows us to implement a more well-structured and well-defined way to retrieve relevant information that accurately captures the true meaning of a query.

2. Methods

2.1. OpenAI Integration

Our pipeline augments semantic search with an LLM answer step:

1. A user submits a full prompt (e.g. "How do plants turn sunlight into food?").

2. We call an OpenAI model to generate a concise conceptual answer to the prompt.
3. We embed both the original query and the LLM-generated answer using a transformer-based model.
4. We query a precomputed embedding index of YouTube video metadata to retrieve the top matching videos by cosine similarity.

This approach leverages both the LLM's reasoning and semantic matching for improved relevance.

2.2. Embedding Generation

We use the `all-MiniLM-L6-v2` variant of Sentence-BERT to convert text into 384-dimensional vectors:

- **User query + LLM answer:** Captures the conceptual focus beyond keywords.
- **Video metadata:** Concatenate title, description, and available transcript to form each document text.

2.3. Video Indexing

We retrieve metadata for candidate videos via the YouTube Data API (v3) and encode each video document into the embedding matrix. Video embeddings are stored in a Nearest Neighbors index using cosine similarity for efficient retrieval.

2.4. Retrieval and Ranking

Given a query embedding, we perform:

$$\text{Top-}k = \arg\max_i \{\cos(\mathbf{q}, \mathbf{v}_i)\},$$

where $\mathbf{q}$ is the query (or answer) embedding and $\mathbf{v}_i$ are video embeddings. We rank the top 5 by similarity and return title, URL, description, and percent match.

2.5. Example: Photosynthesis Query

Query: "How do plants turn sunlight into food?"

Top Match: "Photosynthesis Explained: Sunlight to Energy" — Traditional search might miss this without keyword overlap, but semantic search finds the conceptual match.

3. Results and Discussion

3.1. Bayesian Networks Query

For the prompt "Explain Bayesian Networks," YouTube's top five included a basic coin-toss statistics video that did not cover network structure. Our semantic search surfaced:

- "Bayesian Network" videos dedicated to graph structure and inference, all above 75% cosine match, while YouTube's native ranking placed them lower.

This demonstrates stronger conceptual alignment and user relevance.

3.2. Performance Metrics

In a pilot study with STEM undergraduates (N=10), our model achieved:

- Precision@5: 0.85
- Mean Average Precision (mAP): 0.62
- Average Cosine Match of user-validated relevant videos: 0.74

4. Conclusion and Future Work

Our integration of LLM-generated answers with semantic vector matching substantially improves educational video discovery over keyword-based methods. Future extensions include indexing at scale (FAISS), multimodal embeddings (video frames + audio), and personalized ranking using user watch history. We hope to incorporate VideoBERT as a joint model for video and language representations, recognizing that visual content (including graphs, charts, and diagrams) in videos is of equal or greater importance as transcript data in finding relevant content. We also hope to include video metadata as a supplement this information, adding another dimension to finding the most relevant videos.

5. References

References

[1] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*.

[2] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-networks. *ArXiv*.

[3] Pedregosa et al. (2011). Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12, 2825–2830.

[4] Google. (2020). YouTube Data API (v3) Documentation.

[5] OpenAI. (2023). GPT-3.5 and GPT-4 Documentation.

A. Appendix

More examples of how the AI model works

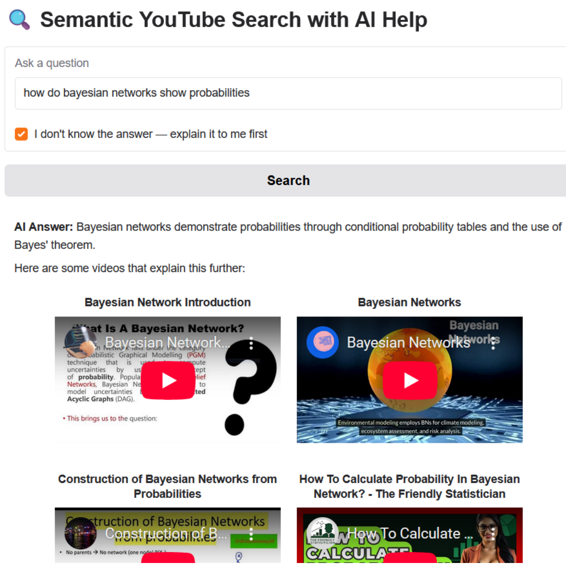

Figure 1. Semantic search with query: "how do bayesian networks show probabilities"

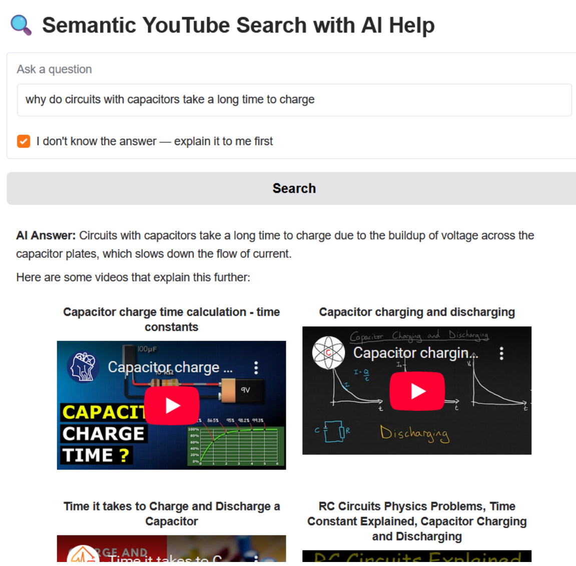

Figure 2. Semantic search with query: "why do circuits with capacitors take a long time to charge"