Semantic video content search and recommendation

Mark David Longar*
Jožef Stefan Institute
Ljubljana, Slovenia

Jakob Fir*
University of Ljubljana
Ljubljana, Slovenia

Bor Pangeršič*
University of Ljubljana
Ljubljana, Slovenia

Abstract

The rapid growth of video streaming platforms has intensified the demand for personalized content recommendations. However, current solutions often rely on historical user data, leading to challenges like the cold start problem and overlooking users' immediate preferences. We present a conversational recommendation system that leverages large language models (LLMs) to generate keyword-based content and query descriptions. By integrating Retrieval-Augmented Generation (RAG), our system efficiently retrieves relevant content, independent of prior user interactions, and ensures consistent performance across languages. Preliminary testing shows our system outperforms the RAG baseline by up to 24% in less descriptive queries and demonstrates consistent performance across three languages. While the results are promising, further evaluation focusing on user interaction and satisfaction is necessary. Our approach can potentially be extended to other recommendation systems, offering broader applicability and enhanced content personalization.

Keywords

large language models, recommendation system, search system, retrieval augmented generation

1 Introduction

The surge in video streaming platforms has accelerated the demand for personalized content recommendations. As these platforms expand their libraries and user bases, the challenge of delivering precise, user-specific recommendations intensifies. In this dynamic environment, streaming services must quickly adapt to provide accurate recommendations, which are crucial for maintaining user engagement and ensuring satisfaction.

Existing recommendation systems primarily rely on historical user interaction data, such as viewing history and ratings. This dependence leads to significant challenges, such as the cold start problem, where new users or newly added content lack sufficient data for accurate recommendations. Additionally, these systems often fail to account for users' immediate preferences, which can change dynamically due to various factors such as mood, viewing context (e.g., watching alone or with a group), or recent events in the user's life. This gap highlights the need for more adaptive and responsive recommendation mechanisms.

Recent advancements in Large Language Models (LLMs) present an opportunity to address these limitations. LLMs offer significant potential due to their emergent reasoning abilities, their capacity to extract high-quality representations of textual features, and their ability to leverage the vast external knowledge encoded within them [10], [7]. By harnessing LLMs, it is possible to create a recommendation system that interacts with users to capture their immediate preferences, thereby overcoming the cold start problem and enhancing the relevance of recommendations. Additionally, ensuring consistency in the quality of recommendations across different languages is increasingly important as many streaming services operate globally.

Our approach utilizes LLMs to generate keyword descriptions for both content and user queries. These keywords serve as the basis for recommendations, with a Retrieval-Augmented Generation (RAG) [6] model efficiently retrieving relevant content. By crafting query keywords using LLMs, the system adapts to user preferences in real time, providing relevant and language-consistent recommendations.

This paper makes the following contributions: **(1) Development of a Keyword-Based Recommendation System:** We introduce a novel approach that utilizes LLMs to generate keyword-based descriptions for content and user queries, enabling more personalized and adaptive recommendations. **(2) Exploration of Two User Interaction Models:** We propose and evaluate two distinct interfaces for user interaction—a conversational chat-based model and a structured question-answering model, where the system refines recommendations through a series of targeted yes/no questions generated by the LLM. **(3) Comprehensive Evaluation Strategy:** We outline a detailed plan for evaluating the system's performance in a production environment, focusing on its ability to deliver consistent, high-quality recommendations across different languages and user contexts.

2 Related Work

Recommender systems have progressed from techniques such as collaborative filtering and matrix factorization to more complex models that incorporate deep learning. The advent of large language models (LLMs) has enabled innovative methods for interacting with these systems [11], particularly when combined with retrieval techniques [9]. One of the most promising advancements in this area is the use of Retrieval-Augmented Generation (RAG) models, which integrate the powerful text generation capabilities of LLMs with retrieval-based methods to improve recommendation accuracy and relevance [6].

Recent advancements in conversational recommender systems have focused primarily on integrating LLMs with traditional recommender systems or fine-tuning LLMs using user-item interaction data [9], [10], e.g., [8], [4], and [5]. These approaches, while effective, often rely heavily on historical user data, leading to challenges such as the cold start problem. This reliance underscores the need for novel methods that reduce dependency on past interactions and leverage real-time retrieval mechanisms to enhance content recommendations [2].

To address these challenges, recent work by Di Palma et al. (2023) [2] introduced a Retrieval-Augmented Recommender System, which combines the strengths of LLMs and retrieval-based methods. Their approach employs LLMs both at the conversational layer and the backend retrieval process, thereby improving recommendation relevance, particularly in scenarios with sparse data or new users. Their experimental results demonstrated that

*All authors have contributed equally.

Information Society 2024, 7–11 October 2024, Ljubljana, Slovenia

https://doi.org/10.70314/is.2024.sikdd.10

this RAG-based framework performs comparably to state-of-the-art systems, even in zero-shot scenarios, underscoring the potential of such an approach to mitigate cold start and hallucination problems inherent in LLMs.

Our approach builds on the strengths of RAG-based models by introducing a keyword-based recommendation system that operates within a RAG framework. This system ensures consistent performance across multiple languages and adapts to real-time user preferences without relying on historical user data.

3 Data

The data used in this study was provided by our partner United Cloud, who operate a multinational streaming service in the Balkan region, EON TV[1]. The EON platform encompasses a variety of content, such as video-on-demand (VOD) movies and TV shows, as well as live TV channels. We focused exclusively on VOD movie data, although our approach is capable of accommodating multiple content types.

The VOD movies data set comprises nearly 5000 movies in various languages. Each movie is accompanied by a brief description averaging around 460 characters (5-6 sentences) in multiple languages. In cases where multiple translations were available, we opted for the original language of the movie; otherwise, we chose the first available translation.

4 Methodology

4.1 Recommendation Mechanism

The core of our recommendation system is the generation of textual representations of content. Instead of using movie descriptions directly, we employ the LLM to generate a set of English keywords and related movies. This approach prevents the model from overemphasizing less relevant details, such as specific plot points, that may not be central to the user's query. User queries follow a similar approach, where the LLM generates a set of relevant keywords, as well as any possibly relevant movies.

One of the key advantages of this method is its ability to abstract core concepts from user queries using the LLM, aligning better with the keywords generated from movie descriptions. The LLM-generated keywords from both the movie descriptions and user queries are designed to encapsulate the essential topics and themes. By aligning the keywords generated from movie descriptions with those derived from user queries, our system enhances the relevance of the recommendations. This alignment is crucial in ensuring that the retrieved movies resonate with the user's expressed interests, even when these interests are not articulated well. Furthermore, the use of in-context learning allows the system to maintain its performance without extensive fine-tuning [3], making it both efficient and effective.

The rest of the recommendation system follows the Retrieval-Augmented Generation (RAG) [6] pipeline (see Figure 1). The RAG pipeline operates by first generating textual representations of movies, which are then embedded into a vector space. These embeddings are stored in a vector database, allowing for efficient similarity searches. When a user submits a query, the system generates a corresponding representation, embeds it into the same vector space, and retrieves the top k most similar movie embeddings from the database. This process ensures that the recommendations are both contextually relevant and semantically aligned with the user's input.

[1]No EON user data was used.

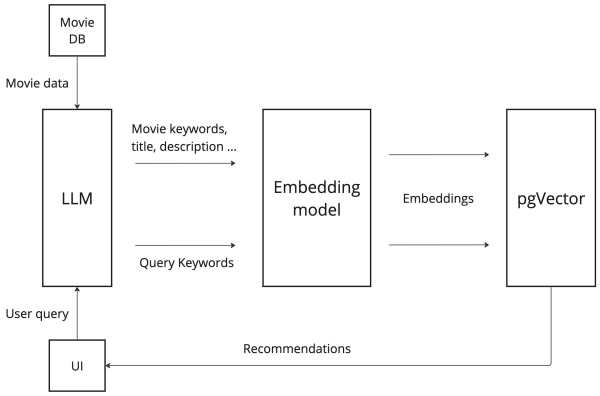

Figure 1: Overview of the Recommendation Pipeline.

4.2 User Interface

Our proposed user interface designs (see Figure 2) offer two main ways for users to interact with our recommendation core. Besides a direct search, where the user submits a query and receives recommendations in a single step, we propose: **(a)** A chatbot, which assists users in narrowing down their options through a conversational interface. The chatbot provides recommendations at each response, allowing for a multi-step interaction that refines the search results progressively. **(b)** An inquisitive method, where an agent asks the user a series of Yes/No questions to narrow down the search. Keywords are generated based on the user's responses, making it particularly useful for users who are uncertain about what they want to watch. This approach shifts the burden of knowing what to query from the user to the system, streamlining the recommendation process.

Each of these designs aims to enhance user engagement and satisfaction by providing tailored interactions that cater to different user preferences and needs.

5 Evaluation

We have developed a twofold approach for addressing the evaluation of our model:

First, to gauge the effectiveness of our keyword-based approach for recommendation, we curated a small multilingual evaluation dataset to test our core recommendation mechanism. This dataset includes queries in various languages along with their expected recommendations. We compared the performance of our mechanism with a baseline RAG system that directly embedded user queries and movie descriptions.

Second, to assess the efficiency and user satisfaction of our system in real-world situations, we have devised an evaluation plan to test our system in production. This strategy utilizes a structured A/B testing framework to conduct precise comparisons between our semantic recommendation system and conventional search, addressing distinct aspects of user experience and system performance.

5.1 Evaluation dataset

To create our evaluation dataset, we carefully selected 25 movies across multiple languages, including both well-known and lesser-known titles. For each movie, we formulated two types of queries to assess the system's retrieval accuracy: *Descriptive* and *General* queries.

The *Descriptive* queries were designed to simulate scenarios where the user knows exactly what they are looking for. For instance, a query for the movie *Messi (2014)* might be, "I am looking for inspirational documentaries about famous athletes, such as Lionel Messi and his rise through football." In contrast, the *General* queries were intended to test situations where the user has only a rough idea of what they want to watch, which is likely more common in real-world environments. An example of a general query for the same movie might be, "soccer movies that will inspire me."

To evaluate the system's performance across different linguistic contexts, we manually translated these queries into English, Serbian, and Slovenian. We then compared the performance of our keyword-based retrieval mechanism against a baseline RAG model that directly used user queries and movie descriptions without generating keywords.

5.2 Experiment Design

We have divided our user base into four distinct groups to facilitate a detailed comparative analysis, aligned with our proposed user interface designs:

Baseline Group: This control group doesn't use our system, but instead finds movies and receives recommendations based on the traditional recommendation methods, a common practice in the industry.

Direct Semantic Search Group: This control group interacts with a straightforward search interface. Users submit a query and receive recommendations in a single step. This approach provides immediate suggestions based on the user's input, mimicking traditional full-text search practices.

Chatbot Group: Participants in this treatment group use a conversational interface (interface **a**), where a chatbot assists in narrowing down options. The chatbot provides recommendations at each response, enabling a multi-step interaction that progressively refines the search results. This design enhances engagement by simulating a natural conversation.

Inquisitive Method Group: Users in this group engage with an agent that asks a series of Yes/No questions to narrow down the search (interface **b**). Keywords are generated based on the user's responses.

The evaluation will be conducted continuously, starting with a focused initial phase over the first month post-implementation to address immediate usability and performance issues, followed by ongoing monitoring to capture long-term user engagement and satisfaction.

By implementing this structured evaluation framework, we aim to comprehensively understand the impact and effectiveness of our semantic recommendation system, guiding further refinements and ensuring that the system meets user needs and expectations.

5.2.1 Metrics We would like to measure how users interact with our system in two main ways: First, we would like to know how engaged and satisfied they are with our recommendations, i.e., do users find our system frustrating to navigate, and whether they watch movies recommended by our system. The second set of metrics will aim to capture how different demographics interact with our system, as a major goal is to remove any biases such as language or age.

Engagement and Satisfaction Metrics: These include Click-Through Rate (CTR), which measures the percentage of clicked recommendation links, and Watch Time to gauge the duration users engage with recommended content. Additionally, immediate user reactions are captured through Like/Dislike Ratios, while more detailed user feedback is collected via surveys administered after interactions.

Behavioral Metrics: We analyze User Interaction Patterns, such as search frequency and refinement actions, and System Usage Frequency to determine how different demographics utilize the system and to identify any potential biases in system engagement. We also record the search time and number of queries needed for a decision.

6 Results

The outcomes presented in Table 1 showcase the performance of both models in various query types and languages, as measured by accuracy at the top 5 and top 10 recommendations.

The results reveal that the baseline model surpasses (or matches) the performance of the keyword mechanism in the case of *Descriptive* queries, particularly in terms of Accuracy@5. However, in terms of Accuracy@10, the two models demonstrate relatively similar performance. Conversely, the keyword model shows significant performance enhancements for *General* queries, particularly in Accuracy@10, indicating its capacity to adapt to non-specific content descriptions. Additionally, the keywords model consistently performs well across different languages, whereas the baseline model shows fluctuations of up to 28% across languages.

In summary, the keywords model allows for more general and multilingual queries, while the baseline model excels at retrieving very specific content.

Table 1: Evaluation results on the descriptions and general queries data sets. LLM embeddings were generated using OpenAI's *text-embedding-3-large* model. The Keywords model used *GPT-4o*.

	Accuracy@5		Accuracy@10	
	Keywords	Baseline	Keywords	Baseline
Descriptive Queries				
English	60%	**64%**	**68%**	**68%**
Serbian	56%	**80%**	72%	**84%**
Slovenian	56%	**80%**	72%	**84%**
General Queries				
English	**44%**	28%	**68%**	44%
Serbian	44%	**52%**	**68%**	52%
Slovenian	44%	**56%**	**72%**	56%

6.1 User Interface Implementation

We implemented our proposed interface design using Flutter, which guarantees functionality across a variety of devices, including iOS, Android, Windows, and web browsers. This cross-device compatibility is crucial as it ensures that all users, regardless of their preferred platform, have access to our application. The support for mobile devices is particularly useful in our interrogation design, where users can easily navigate through options by swiping cards left or right.

Additionally, we integrated Tipko [1], a Slovenian transcription service, to facilitate voice-to-text capabilities. This feature

enhances user convenience by enabling voice communication with our chat bot, removing the necessity for typing.

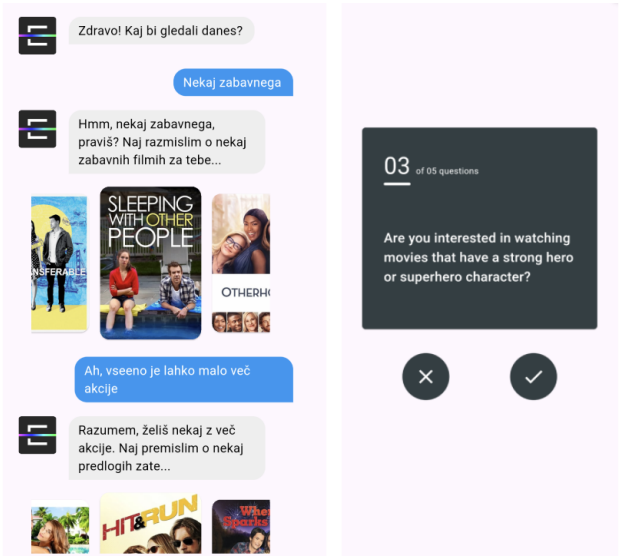

Figure 2: Implementations of our *(a) Chatbot* (left) and *(b) Inquisitive* (right) user interface designs.

7 Discussion

This report introduces a new content recommendation mechanism and three ways to interact with it. Table 1 demonstrates the success of our keyword retrieval model in understanding general user preferences while still performing well when searching for specific content. Moreover, its consistency across languages and its ability to retrieve content using specific descriptions as well as general themes make it well-suited for a diverse user base. Additionally, the keyword model allows seamless integration with both the *Chatbot* and *Inquisitive* methods. Moreover, our system could be extended to dynamically adjust keyword generation based on user-specific factors such as viewing history, local time, weather, and current mood indicators. This personalization ensures that the recommendations are not only relevant to the content but also tailored to the user's immediate context and preferences.

Our approach has some limitations, including the cost per query, which is higher than traditional search, although not exorbitant. Furthermore, our model's performance is commendable given our limited knowledge about the movie content but relies on the assumption that the language model may have more information about a movie than our dataset. It's worth noting that, in the short term, it appears that models are continually improving, becoming faster, more knowledgeable, and more cost-effective.

Lastly, as with any chat application that involves user inputs, security is a crucial consideration. While improvements can be made through better prompting and fine-tuning, ongoing monitoring is essential when the system is in production.

8 Future work

In future work, we plan to further explore methods for improving user experience and personalization. Our initial experiments have involved incorporating the user's time, location, and weather to enhance results. Moving forward, we aim to explore additional integrations, such as the user's calendar. We also intend to expand our user interface by introducing new forms of interaction, such as movie trailers and multiple-choice questions.

To overcome the limitations of our movie information, we are interested in delving deeper into the content by analyzing subtitles using a local language model. Additionally, we aim to broaden our database to include other types of content, such as live channel content and special time-limited events like Eurovision, Eurobasket, and the FIFA World Cup.

Finally, we are interested in the integration of a traditional recommendation models that utilize historical watch data or ratings to re-rank our recommendations.

Acknowledgments

This project was made in collaboration with United.Cloud and In516ht for the 2024 Data Science Competition, organized by The Faculty of Computer and Information Science at the University of Ljubljana. We thank our advisors Slavko Žitnik, Aljaž Košmerlj, Klementina Pirc, and Rebeka Merhar for their contributions.

References

[1] Primož Bratanič. *Transkript app | Samodejna transkripcija slovenskega govora*. May 2024. URL: https://transkript.si/.

[2] Dario Di Palma. "Retrieval-augmented recommender system: Enhancing recommender systems with large language models". In: *Proceedings of the 17th ACM Conference on Recommender Systems*. 2023, pp. 1369–1373.

[3] Elnara Galimzhanova et al. "Rewriting Conversational Utterances with Instructed Large Language Models". In: (Oct. 2023). DOI: 10.1109/wi-iat59888.2023.00014. (Visited on 05/22/2024).

[4] Yunfan Gao et al. "Chat-rec: Towards interactive and explainable llms-augmented recommender system". In: *arXiv preprint arXiv:2303.14524* (2023).

[5] Xu Huang et al. "Recommender ai agent: Integrating large language models for interactive recommendations". In: *arXiv preprint arXiv:2308.16505* (2023).

[6] Patrick Lewis et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 9459–9474.

[7] Peng Liu, Lemei Zhang, and Jon Atle Gulla. "Pre-train, Prompt, and Recommendation: A Comprehensive Survey of Language Modeling Paradigm Adaptations in Recommender Systems". In: *Transactions of the Association for Computational Linguistics* 11 (2023), pp. 1553–1571.

[8] Zihan Liu et al. "ChatQA: Building GPT-4 Level Conversational QA Models". In: *arXiv preprint arXiv:2401.10225* (2024).

[9] Arpita Vats et al. "Exploring the Impact of Large Language Models on Recommender Systems: An Extensive Review". In: *arXiv preprint arXiv:2402.18590* (2024).

[10] Likang Wu et al. "A survey on large language models for recommendation". In: *World Wide Web* 27.5 (2024), p. 60.

[11] Bowen Zheng et al. "Adapting Large Language Models by Integrating Collaborative Semantics for Recommendation". In: *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. 2024, pp. 1435–1448. DOI: 10.1109/ICDE60146.2024.00118.