

Lost in Plot: Contrastive Learning for Tip-of-the-Tongue Movie Retrieval

Obed Junias

Department of Computer Science
University of Colorado Boulder
Boulder, CO, USA
obed.junias@colorado.edu

Abstract

Humans often recall a movie by fragments usually using bits of plot, emotion, or striking visuals rather than its exact title. Traditional search engines and keyword-based retrievers struggle with such vague queries, while generative models can hallucinate plausible but incorrect titles. In this work, we propose a dense retrieval approach that leverages contrastive learning to align user descriptions and movie metadata in a shared semantic space. We evaluate this system on a public dataset derived from TMDb and IMDb and compare it with two strong baselines: few-shot prompting of large language models and the vanilla encoder before contrastive fine-tuning. Our experiments demonstrate that the contrastively fine-tuned model outperforms the GPT-4 few-shot baseline on both Recall@K and MRR, but falls short over the base encoder.

1 Introduction

The ability to identify a specific movie from vague natural language descriptions represents an intersection of *Natural Language Processing (NLP)*, *Information Retrieval (IR)*, and *semantic understanding*. We often struggle to recall the name of a movie we once saw. Instead of exact titles, we remember fleeting details: a character’s tearful goodbye, a tragic love story with time travel, and this is closely related to the phenomenon of “tip-of-the-tongue” retrieval ([Arguello et al., 2021](#))

In practice there’s a big gap between how we naturally describe a movie and how it’s stored in a database. We remember movies as “that comic-horror about a haunted house at Christmas,” or “the one where Brad Pitt plays Death,” not as neat keywords. These vague descriptions leave keyword-search embeddings often struggling. This problem isn’t just limited to movies: people describe songs as “that track about flying and loss” or photos as “the waterfall at sunrise.”

To tackle this gap, we investigate whether contrastive learning ([Izacard et al., 2022](#)) can enable better semantic retrieval of movies from vague user descriptions.

So, we frame our research around the following questions:

1. How effective is contrastive learning-based dense retrieval for retrieving movies from vague user descriptions?
2. How does this fine-tuned model compare to few-shot prompting methods using GPT-4 ([OpenAI, 2023](#)) and base encoder model without any fine-tuning?

To address these:

1. We build a dense retrieval system that pulls together semantically similar movies and pushes apart dissimilar ones via a contrastive loss based on latent themes.
2. We create a synthetic evaluation dataset of 3000 vague descriptions using GPT-4 few-shot prompting.
3. We index 100k movies in FAISS and compare retrieval performance against GPT-4 few-shot and vanilla BERT using Recall@1/5/10/25 and MRR.

Our results show clear gains over GPT-4 few-shot prompting, but fail to even match the vanilla BERT encoder on Recall@K and MRR.

Related Work

Tip-of-the-tongue retrieval tasks and their connection to NLU have received growing attention due to their real-world applications. Arguello et al. ([Arguello et al., 2021](#)) in their study, highlight the need for query understanding systems that can interpret incomplete queries efficiently.

Lin et al. (Lin et al., 2023) introduced a method to decompose complex user queries for tip-of-the-tongue retrieval. Their approach highlights the challenges in handling long, semantically rich queries, an aspect we address using dense retrieval methods and contrastive embeddings.

Izacard et al. (Izacard et al., 2022) proposed an unsupervised contrastive learning approach (Contriever) for dense information retrieval. While their method shows strong performance in zero-shot retrieval, it is domain-agnostic. Our approach adapts contrastive learning to the movie domain for enhanced semantic embeddings along with knowledge-graph-based disambiguation.

Zhang et al. (Zhang et al., 2022) propose LaCon, a supervised contrastive framework that uses class labels to mine both hard positives and negatives. LaCon integrates smoothly with pre-trained transformers and yields up to 4.1% improvement on GLUE and 9.4% on FewGLUE benchmarks. This work demonstrates leveraging label semantics directly in contrastive objectives, which inspired me to use theme, genre, and decade labels to sample positives and guide auxiliary classification in movie retrieval.

We use BERTopic (Grootendorst, 2022) to find latent “themes” from movie overviews. BERTopic first embeds documents with pre-trained BERT, then applies a class-based TF-IDF procedure to surface coherent topic clusters. In our current work, this provides a lightweight, unsupervised signal that we integrate as a third contrastive head alongside genre and decade.

To assess retrieval quality, we use Recall@K and Mean Reciprocal Rank (MRR) (Manning et al., 2008). Recall@K measures the fraction of queries whose correct movie appears in the top-K results, while MRR averages the inverse rank of the first correct hit. These metrics are widely used in closed-set retrieval evaluations and offer a clear view of both coverage (Recall) and ranking precision (MRR).

2 Methodology

The main aim is to learn a joint embedding space where these vague user descriptions and movie metadata are close together, so that cosine-similarity ranking retrieves the correct film from any type of natural language queries.

In this section, we describe how we train a dense retrieval model that aligns vague user descriptions

and movie metadata in a shared semantic space.

2.1 Data Preparation

We start by combining the TMDb/IMDb metadata (plot, titles, keywords). We encode genres as a multi-hot vector and map each release year to its decade. To get a latent “theme” signal, we run BERTopic over all plot summaries and treat each movie’s topic ID as its theme label. These three signals guide both the triplet sampling and the auxiliary classification heads.

2.2 Model Architecture

We use a pre-trained BERT encoder (Devlin et al., 2019) to obtain contextualized embeddings, apply mean pooling to produce a fixed-length vector, and then project it into a lower-dimensional retrieval space. Concretely, for an input text sequence text with attention mask \mathbf{m} , we then compute

$$\mathbf{h} = \text{MPool}(\text{BERT}(\text{text}), \mathbf{m}) \in \mathbb{R}^H,$$

where MPool denotes element-wise mean pooling over the non-padded tokens. We then apply a learned linear projection and ℓ_2 -normalization to get a resultant dense vector $\hat{\mathbf{z}}$ which serves as the retrieval embedding. In addition, we attach three classification heads for genre, decade, and theme classification, each implemented as a single linear layer on $\hat{\mathbf{z}}$.

2.3 Anchor-Positive Sampling

To teach the model to pull semantically similar movies closer and push dissimilar ones apart, we sample the training data into $(a_i \text{ and } p_i)$ where:

- a_i is the “anchor”: a movie’s combined metadata (plot, titles, keywords) sampled belonging to a particular theme.
- p_i is a “positive” example: a different movie sharing the same BERTopic theme as the anchor.

All the remaining examples in the batch serve as implicit negatives for anchor a_i in our retrieval loss.

2.4 Multi-Task Loss

We combine a contrastive retrieval loss with three auxiliary classification losses. Denoted by $\hat{\mathbf{z}}_i^a, \hat{\mathbf{z}}_i^p \in \mathbb{R}^D$ the embeddings of anchor and positive i , we define the InfoNCE retrieval loss as:

$$\mathcal{L}_{\text{retr}} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\hat{\mathbf{z}}_i^a \cdot \hat{\mathbf{z}}_i^p / \tau)}{\sum_{j=1}^N \exp(\hat{\mathbf{z}}_i^a \cdot \hat{\mathbf{z}}_j^p / \tau)}$$

where τ is a temperature hyperparameter. For each anchor, we also predict:

- A multi-label *genre* vector via binary cross-entropy.
- A *decade* label using cross-entropy.
- A *theme* label using cross-entropy.

Let $\mathcal{L}_{\text{genre}}$, $\mathcal{L}_{\text{year}}$, and $\mathcal{L}_{\text{theme}}$ denote these classification losses. We weight and sum them to form the overall training objective:

$$L = w_{\text{retr}} \mathcal{L}_{\text{retr}} + w_{\text{genre}} \mathcal{L}_{\text{genre}} + w_{\text{year}} \mathcal{L}_{\text{year}} + w_{\text{theme}} \mathcal{L}_{\text{theme}}.$$

2.5 Training Details

We fine-tune the entire model (encoder, projection, and heads) end-to-end using AdamW with weight decay. Key hyperparameters include:

- **Batch size:** 16
- **Learning rate:** 2×10^{-5} for all parameters
- **Scheduler:** Cosine decay with 10% warm-up
- **Epochs:** 8 (early stopping on MRR)
- **Evaluation:** Encode all 100K movies into FAISS index, measure Recall@1/5/10/25 and MRR on held-out synthetic queries

3 Experiments

We evaluate our proposed contrastive retrieval model on a large-scale movie corpus and compare it against two strong baselines. This section describes the dataset, baselines, metrics, inference details, and the results of our experiments.

3.1 Dataset

We use a public movie metadata collection drawn from TMDb and IMDb¹ and then derive three key resources:

- **Corpus:** A collection of 100,000+ movies, each represented by its title, plot overview, genre, release year, and tagline.
- **Latent themes:** BERTopic is run on all the plot overviews to produce latent themes.
- **Synthetic queries:** We then use GPT-4 to generate 3000 “vague descriptions” via the few-shot prompting technique (e.g. “a sci-fi romance about time loops”).

¹<https://www.kaggle.com/datasets/alanvourch/tmdb-movies-daily-updates>

We hold out 1,500 queries for validation and report final metrics on the remaining 1,500 queries.

3.2 Baselines

We compare three retrieval methods:

1. **GPT-4 few-shot prompting:** Given two example pairs, we prompt GPT-4 to generate 25 movie titles for each query.
2. **Vanilla encoder:** Pre-trained BERT-base encodes both queries and all 100 k movies; movie embeddings go into FAISS index, and we retrieve by cosine similarity.
3. **Contrastive model:** Similarly, fine-tuned BERT-base with projection and multi-task heads encodes both queries and movies metadata; retrieval is again via FAISS over its normalized embeddings.

3.3 Evaluation Metrics

We measure the retrieval quality using:

- **Recall@K:** fraction of queries for which the correct movie appears in the top K results.
- **Mean Reciprocal Rank (MRR):**

$$\text{MRR} = \frac{1}{Q} \sum_{i=1}^Q \frac{1}{\text{rank}_i},$$

where rank_i is the position of the correct movie for query i .

3.4 Inference Details

- **Indexing:** We encode all 100K movies into a FAISS IndexFlatIP index using 512-dim normalized embeddings.
- **Querying:** Each query embedding is matched against the index to retrieve top-25 candidates.

3.5 Results

Table 1 summarizes the main findings. The contrastively fine-tuned model outperforms GPT-4 by a large margin but still lags behind the vanilla encoder, indicating room to improve the architecture.

From the above table 1, we make three key observations:

- **Vanilla BERT** handles vague inputs surprisingly well, with a recall@1 of 13.93 % and MRR of 0.1883.

Model	R@1	R@5	R@10	R@25	MRR
Vanilla BERT	0.1393	0.2447	0.2933	0.3713	0.1883
Contrastive Fine-Tuned BERT	0.1140	0.1960	0.2427	0.3107	0.1540
GPT-4 Few-Shot Prompting	0.0647	0.0827	0.0893	0.0913	0.0724

Table 1: Retrieval metrics on 1500 synthetic queries.

- **Contrastive fine-tuning** yields a slight drop (recall@1 = 11.40 %, MRR = 0.1540). This suggests that our current positives and multi-task weighting need further tuning to outperform the base encoder.
- **GPT-4 few-shot prompting** performs worst (recall@1 = 0.0647 %, MRR = 0.0724), confirming that even an under-performing domain-specific dense retriever beats the general state-of-the-art LLM on this task.

In summary, vanilla BERT sets a strong baseline, and GPT-4’s predictions generated using few-shot prompting struggles on sparse, vague queries. Our contrastively fine-tuned model narrows the gap to vanilla BERT but still requires improvements (e.g. harder negatives, adjusted loss weights) to consistently outperform it.

4 Discussion

In this work, we set out to bridge the gap between how users describe a movie and how movies are represented in an encoder’s embedding space using contrastive learning. Our contrastively tuned model yields clear gains over GPT-4’s predictions (Table 1). However, it still falls slightly behind the vanilla encoder on Recall@K and MRR. This mixed result points to both promise and challenges in contrastive learning for movie retrieval.

Why Contrastive Fine-Tuning outperforms GPT-4 Few-Shot The results clearly show that GPT-4 often hallucinates plausible titles instead of grounding its answers. However, a lightly fine-tuned dense encoder uses cosine similarity on pre-computed embeddings, making it much more reliable for retrieving the correct film.

Why It Lags Behind Vanilla BERT We believe the drop in performance points to a few issues:

- **Very large theme space:** A large number of themes can introduce noisy or unrepresentative positives, as some themes might only have a handful of movies.

- **Theme granularity is coarse:** Movies sharing the same high-level topic may still be semantically distant (e.g., two “sci-fi” films with very different plots).
- **Positive sampling is weak:** By relying only on shared themes, we ignore stronger signals like overlapping keywords, cast, or directors.
- **Competing classification heads:** The genre, decade and theme heads can pull the model in different directions. Balancing multiple losses and three auxiliary objectives is delicate, and the classification tasks may overwhelm the core retrieval signal.

5 Conclusion

In this work, we have presented a contrastive learning framework for retrieving movies from vague, “tip-of-the-tongue” descriptions. Starting from a pre-trained BERT encoder, we added lightweight projection and classification heads. And we fine-tuned the model with a contrastive + auxiliary loss. In evaluation against GPT-4 few-shot prompting and the vanilla BERT encoder, our fine-tuned model clearly beats the GPT-4 few-shot baseline on Recall@K and MRR, but still falls short of the vanilla BERT encoder’s performance.

Despite these gains, there is still a lot of room left to improve. Future work will explore hard negative mining, partial fine-tuning of only the top transformer layers, and evaluation on real user queries rather than synthetic descriptions. Incorporating richer metadata signals (cast, directors) into the contrastive objective also promises to strengthen performance.

Overall, this study demonstrates that even modest contrastive fine-tuning of a vanilla encoder can yield practical benefits in a reliable movie retrieval, making way for more robust searchers when exact keywords fail.

6 Limitations

Although this study offers a clear proof of concept, there are a few limitations to keep in mind:

1. **Synthetic queries.** Our evaluation uses GPT-4-generated descriptions, which may not fully reflect real user phrasing or distribution of “tip-of-the-tongue” queries.
2. **No hard negatives.** We did not incorporate hard negative mining, which can sharply boost

contrastive retrieval as per prior work.

3. **Full-model fine-tuning.** We fine-tuned all BERT layers uniformly. Partial fine-tuning (e.g., only the top 2–4 transformer blocks) might preserve more of the base encoder’s generic semantics while adapting to movie retrieval.
4. **Compute constraints.** Memory limits (P100 with 16 GB VRAM) forced small batch sizes and downsampled training, potentially underutilizing the contrastive signal.
5. **Single dataset.** Results are reported on publicly available TMDb/IMDb split. Generalization to other languages, genres, or closed-caption descriptions remain untested.

Despite these limitations, our approach offers a lightweight and scalable path to improve dense movie retrieval, and the insights gathered above will help us build stronger models in the future.

7 Ethical Considerations

This work relies on publicly available movie metadata from TMDb and IMDb. While the dataset is large and diverse, it might still reflect some historical biases which could affect retrieval fairness. Additionally, we use GPT-4 to generate evaluation queries; these synthetic descriptions may not capture the full variability of real user queries.

References

- Jaime Arguello, Adam Ferguson, Emery Fine, Bhaskar Mitra, Hamed Zamani, and Fernando Diaz. 2021. [Tip of the tongue known-item retrieval: A case study in movie identification](#). In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, CHIIR ’21, page 5–14, New York, NY, USA. Association for Computing Machinery.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Maarten Grootendorst. 2022. [Bertopic: Neural topic modeling with a class-based tf-idf procedure](#). *Preprint*, arXiv:2203.05794.
- Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sébastien Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. [Unsupervised dense information retrieval with contrastive learning](#). *Transactions on Machine Learning Research*.
- Kevin Lin, Kyle Lo, Joseph Gonzalez, and Dan Klein. 2023. [Decomposing complex queries for tip-of-the-tongue retrieval](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5521–5533, Singapore. Association for Computational Linguistics.
- Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. [Introduction to Information Retrieval](#). Cambridge University Press, Cambridge, UK.
- OpenAI. 2023. Gpt-4 technical report. <https://openai.com/research/gpt-4>.
- Zhenyu Zhang, Yuming Zhao, Meng Chen, and Xiaodong He. 2022. [Label anchored contrastive learning for language understanding](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1437–1449, Seattle, United States. Association for Computational Linguistics.