CSCI 4364/6364 Machine Learning
Fall 2022
Section 81

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Movie Recommendation System using the BERT Model[1]

Team Name: Cine Hunters

Project Team Members (1-5 members):

Name	**GWID**
Cora Sula	G23959463
Anulekha Boddu	G40397446

[1] Template derived from Bock, P. (2001) Getting it Right: R&D Methods for Science and Engineering, Academic Press

1 Analysis

Due to the rise of new and more streaming services, users who wish to unwind with a movie can rarely find one on the same platform. As a result, people have to spend a lot of time switching between streaming sites and scrolling endlessly to end up not watching a movie or rewatching one they have already seen. Our model hopes to solve the "mindless scrolling" and time waste by using the BERT model to recommend users a movie based on their preferences in movies and thus reducing the time spent in a day searching for a suitable movie to watch. This approach uses a content-based filtering method to provide users with recommendations similar to a movie they specify. We aim to use the BERT model to analyze movie similarities in the text of the movie overviews and then use the cosine similarity to assign a score to the movies most similar to the one the user wanted to get a recommendation on by giving them the top 3 most similar movies.

Most existing literature utilizes either one of the three main filtering techniques (content-based, collaborative, or hybrid) or a BERT model with collaborative filtering. However, a BERT model with a content-based filtering technique for movie recommendations has yet to be explored in depth to accommodate multiple streaming platforms.

1.1 Problem Description:

A study in 2021[2] showed that people spend about 23 minutes a day on average deciding what movie to watch by scrolling through different streaming platforms, which is approximately about 1.3 years of searching in their lifetime. We would like to use sentence transformers to reduce the amount of time an average user spends a week looking for what movie to watch next by providing them with recommendations. Machine learning will be used to solve this

[2] Alexander, J. (2016) Study: Viewers will spend a total of 1.3 years trying to find something to watch

problem by training the model to suggest a movie to users based on their input movie. The BERT model can also be used alongside the cosine similarity to analyze the similarity in the text of movie overviews and make recommendations based on other movies with similar plots.

Problem Description*:* Can using a sentence transformer model such as BERT improve recommendations by finding movies with similar synopses?

1.2 Performance Criteria:

- Recommendations should give the user 3 movies based on their input
- Give predictions based on user input.
- Provide recommendations from across all streaming platforms
- Recommendations should use content-based filtering to analyze the movie titles, overviews, genre, cast, directors, etc.

1.3 Related Work:

Although there is a large number of existing movie recommendation systems, they are fairly incipient as they are usually found in specific streaming platforms, which in turn will only recommend movies that are available to watch on their platforms. Therefore there is a lot of scope for improvements in the models.

Currently, there are three main approaches or technologies used for recommendation systems: content-based filtering, collaborative filtering, and a hybrid between the two.

1. Content-Based Filtering is arguably the most straightforward system of recommendation, content-based filtering gives importance to the attributes of a movie. Users are recommended movies with similar properties to the ones they like.

2. Collaborative Filtering which is based on the similarity in taste between users. It is more reliant on past behavior. Users are recommended movies that other users with similar interests like.
3. Hybrid Filtering which combines both content-based and collaborative filtering.

BERT (Bidirectional Encoder Representations from Transformers) is an open-sourced language representation model used for Natural Language Processing. It was created by Devlin et. al. in 2018[3]. Language representation models are important for vectorizing and modeling natural language for things such as item descriptions, or in the case of movie recommendations, synopses. BERT is different from other language representation models due to its bidirectional support. It pre-trains deep "bidirectional representations from the unlabeled text by jointly conditioning on both left and right context in all layers." In recommendation systems, BERT is often used to map user behavior sequences.

Cosine similarity measures the cosine angle between two items. The smaller the angle between two vectors, the more similar they are, and vice versa. It is useful in recommendation systems for finding similar products to recommend. In recommendation systems that use content-based filtering, cosine similarity computes the likeness between two items from their descriptions.[4]

Two of the main challenges when designing recommender systems include the cold start problem and data sparsity. The cold start problem occurs if the user is new and therefore user interests are not available. The recommender will struggle to find recommendations. Data sparsity occurs if the user watches a lot of movies but does not rate them.[5]

---

[3] Devlin, Jacob, et al. "Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding." *ArXiv.org*, Cornell University, 24 May 2019, https://arxiv.org/abs/1810.04805.
[4] Singh, Ramni Harbir, et al. "Movie Recommendation System Using Cosine Similarity and KNN." *International Journal of Engineering and Advanced Technology*, vol. 9, no. 5, 2020, pp. 556–559., https://doi.org/10.35940/ijeat.e9666.069520.
[5] Sharma and Dutta (2020) Movie Recommendation Systems: A Brief Overview

Fu and Wang (2020) proposed a method that addresses issues such as the cold start problem or "more of the same" recommended content. They do so by utilizing the BERT model to explore the titles of items and model the relevance between different items. Instead of simply describing an item with a unique number, they used item names as content. They find that their model, when fine-tuned only for the Next Purchase Prediction task, exceeds the baseline precision by 310.9%. [6]

Dang et al. (2021) propose a model that 'integrates sentiment analysis and genre-based similarity in collaborative filtering methods'. They use a BERT model for feature extraction and hybrid deep learning models for sentiment analysis for user reviews. They run their models on music data from existing datasets, MARD and Amazon Movie Reviews.[7]

Channarong et al. (2022) propose a new method they call HybridBERT4Rec, which builds on the existing BERT4Rec model by applying BERT to both content-based filtering and collaborative filtering. They find that their newly proposed model is more accurate than the existing BERT4Rec model.[8]

A study from 2022 proposed a new graph-based movie recommendation based on users' sentiments using BERT, as well as user ratings, and the Kaggle dataset, to recommend users a movie.[9] Their results found that using their model along with emotion sentiment is better than previous models, showing that sentiment analysis improves a movie recommendation system.

---

[6] Yuyangzi Fu and Tian Wang, Item-based Collaborative Filtering with BERT, Proceedings of the 3rd Workshop on e-Commerce and NLP, 2020, pp. 54–58.
[7] Dang et al. (2021) Using Hybrid Deep Learning Models of Sentiment Analysis and Item Genres in Recommender Systems for Streaming Services
[8] Channarong et al. (2022) HybridBERT4Rec: A Hybrid (Content-Based Filtering and Collaborative Filtering) Recommender System Based on BERT
[9] Lee et al. (2022) Improving Graph-Based Movie Recommender System Using Cinematic Experience

In another recent study, Rodríguez-Hernández et al. (2020) evaluated if content-based recommendation systems using linked data and the BERT model can improve the efficiency of recommendation systems.[10] Their proposed model will use an individual user's rating of various items, and then based on their description they will recommend an item they have not yet purchased or consumed.

Penha and Hauff (2021) studied BERT in relation to books, movies, and music and found that 65% of the time BERT selects the most appropriate item to match a user with based on their history. Their research found that BERT has content knowledge about what items are stored, especially if it is about genres.[11]

1.2 Project Objective:

Create a movie recommendation system that can provide users with a movie they would watch based on user input by taking a content-based filtering approach with BERT and the cosine similarity.

2 Hypothesis

2.1 Hypothesis

The recommendation system will produce 3 movie recommendations with the highest similarity to the movie that the user will input.

2.2 Method

We will build a model that takes user input and uses sentence transformers and cosine similarity to suggest something new to watch. We will use BERT to convert the movie synopses into a vector space that we will then use to compute the cosine similarity. The

[10] Rodríguez-Hernández (2020) An Experimental Evaluation of Content-Based Recommendation Systems: Can Linked Data and BERT Help?
[11] Penha, Gustavo, and Claudia Hauff. "What Does Bert Know about Books, Movies and Music? Probing Bert for Conversational Recommendation." *ArXiv.org*, 4 Mar. 2021, https://arxiv.org/abs/2007.15356.

cosine similarity will compare movies through their synopses and find movies that are the most similar in plot.

2.3 Data

We will be using the Kaggle dataset titled "The Movies Dataset." This dataset contains data from 45,000 movies released on or before July 2017 and includes cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, and countries[12]. It also contained 26 million ratings from 270,000 users on popularity, which ranged from 0 to over 400. For pre-processing the data, in order to clean up the data, we chose to remove movies foreign movies, as well as movies that had a popularity score of 3 or under, as they were too obscure and unknown. At the end of our data-cleaning process, we had a little over 10,000 movies left.

2.4 Experiment

We will evaluate the movies generated by the recommender by presenting it with different movies, and then using a manual approach to assess the performance of the model by evaluating the relevancy of each of the given recommendations. We will use movies of different genres, years, age ratings, directors, movies with sequels, etc. to test the recommenders' ability to accurately find the similarity in texts between movie overviews and accurately give the user the most similar ones.

3 Synthesis

We will use the BERT model, The Kaggle "The Movies Dataset", cosine similarity, and a number of python libraries including pandas, scikit-learn, and sentence transformers.

[12] Banik, Rounak. "The Movies Dataset." *Kaggle*, Google, 10 Nov. 2017, https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset.

4 Validation

4.1 Results:

Using a number of trials, we analyzed the accuracy of the recommended movies by seeing if the movies were appropriate and relevant to the user-provided movie based on content, genre, and age-ratings. During the testing phase, we assessed the accuracy of the suggested movies by analyzing the recommendations it gave, to see if the model was adequately functioning and correctly calculating the similarity between the movie overviews.

```
Enter a movie for recommendations: Scream
The top 3 recommendations for Scream are:
Halloween
A Nightmare on Elm Street
Patchwork
Press 1 to enter another movie, 0 to exit: 0
Thank you for using CineHunters!
```

Figure 1. Genre-specific movies using scary movies

In Figure 2. we input the scary movie *Scream*, and the results for recommendations were also all horror. In this instance, the model didn't recommend *Scream 2*, *Scream 3*, and *Scream 4*, but instead gave us variability with the scary movies which is better for the user experience using the system. It can be presumed that a user who has watched and liked *Scream* would know about the sequels, so we appreciated the fact that the model had a larger scope in the recommendations it gave someone, rather than simply returning sequels. Most importantly, it was able to detect the overall genre and theme of the movie based on the analysis of similarities between the movie overviews.

```
Enter a movie for recommendations: Vampire Academy
The top 3 recommendations for Vampire Academy are:
The Twilight Saga: Breaking Dawn - Part 1
BloodRayne
The Twilight Saga: Eclipse
Press 1 to enter another movie, 0 to exit: 1
Enter a movie for recommendations: Twilight
The top 3 recommendations for Twilight are:
Vamp U
Once Bitten
Fright Night 2: New Blood
Press 1 to enter another movie, 0 to exit: 1
Enter a movie for recommendations: What We Do in the Shadows
The top 3 recommendations for What We Do in the Shadows are:
Nadja
My Babysitter's a Vampire
Vamps
Press 1 to enter another movie, 0 to exit: 0
Thank you for using CineHunters!
```

Figure 2. Specific themes and keywords using vampires

In figure 3, we chose a specific trope or symbol to see how the model would perform. We chose 3 different vampire movies, *Twilight, Vampire Academy,* and *What We Do in the Shadows*, to see if the model would pick up on this element group and see if it could recommend only vampire movies. The results show that all the recommendations for all three movies were also other vampire movies across the board, and no other supernatural creatures such as werewolves, zombies, or ghosts. It is also worth noting that all of the recommendations are different and aren't repeated despite each of the inputs having the same specific trope.

```
Enter a movie for recommendations: Shrek
The top 3 recommendations for Shrek are:
The Tale of Despereaux
Tinker Bell and the Legend of the NeverBeast
The Angry Birds Movie
```

Figure 3. Age ratings and contents

Something of great importance in our recommendation was to ensure that a user who inputs a children's movie as their recommendation would not get back adult content that may be inappropriate for their age. When we entered the movie *Shrek* all the recommendations were children's movies.

```
Enter a movie for recommendations: Elf
The top 3 recommendations for Elf are:
The Santa Clause
Ernest Saves Christmas
Arthur Christmas
```

Figure 4. Movie themes

To test the movie recommendation systems' ability to pick up on movie themes, we used a Christmas movie as an example to see if it would also recommend other Christmas movies. All 3 of the movies recommended were also Christmas-themed, just like *Elf.*

The movies we tested our recommendation system with based on age appropriateness, themes, and genres, proved to be successful recommendations that had the same overall content as the movies themselves.

4.2 Limitations:

Our model had limitations that hindered its performance and extent of recommendations. As mentioned, the dataset used only included movies released on or before July 2017. If a user wanted to get a recommendation on a movie they watched and liked released after that, or if we wanted our model to include recent movies in the 3 recommendations, then it would not be able to do that.

It also doesn't consider previous watch history. If a user were to input a movie they wanted to receive recommendations on, then the system could give them back something they may have already watched. Another limitation was that the model was case-sensitive, requiring users to type the movie title exactly as it appears, otherwise, the system would think it doesn't exist. Lastly, this model was computationally expensive. We used GoogleColab and didn't upgrade to PRO, so it took several hours for our model to run, which had to be repeated each time we ran it.

4.3 Future Work:

In the future, we would like to add more user-interface to our model to make it as easy for the user to navigate and follow. We would also like to add many features to enhance their experience with the recommendation system. This includes giving the user the ability to filter movies by different features and add movie logs to track watch history. We also want to specify the streaming platforms that our recommendations are available on to make it more efficient for the users. We also want to give users the ability to like or dislike a recommendation so that the model can improve on its future recommendations.

4.4 Conclusions:

Using a pre-trained Bidirectional Encoder Representation from Transformers and the cosine similarity, we were able to create a model that accurately gives user recommendations based on text analysis using Sentence Transformers, which was able to determine the degree of similarity between a movie the user put in the system against all other movies in the dataset, and give the user the top 3 movies with the highest similarity score.

This model improves on existing models as current recommendation systems are generally found on specific streaming platforms, where they only recommend movies to users based on what is available to watch in their streaming service, as well as based on what others viewed and liked. Our model has no preference for a single streaming site and is not limited in its recommendations on a basis of others as seen in collaborative filtering.

Our recommendation system, with the use of the BERT model and cosine similarity, was able to accurately recommend 3 movies to a user based on the movie they put in the recommendation system. Through a number of trials using different genres, age ratings, and movies that had sequels, we were able to accurately determine if our recommendation system was working and give accurate recommendations to the users based on their movie preferences.

5 Acknowledgments

We would like to thank Professor Sipple for his feedback on our project proposal and guidance throughout the semester in his instruction of the Machine Learning course.