CineSentient: Integrated Sentiment-Enhanced Movie Discovery Engine

Fayaz Moqueem Mohammed
Boston University
fayaz@bu.edu

Gowtham Senthilnayaki
Boston University
gs@bu.edu

Sonu Kumar
Boston University
sonuk@bu.edu

Abstract

This project, CineSentient, aims to revolutionize movie recommendation systems by integrating advanced Natural Language Processing (NLP) and transfer learning techniques with transformer models like BERT, RoBERTa, GPT-2 and DeBERTa. At its core, the project focuses on analyzing movie reviews to extract sentiment, leveraging the power of transfer learning to adapt sophisticated pre-trained models to our specific dataset. Initially, transformer models are trained on a large corpus of diverse movie reviews, enabling them to capture a wide range of sentiments and linguistic nuances. This model is then fine-tuned on a targeted dataset, enhancing its capability to discern subtle sentiments specific to different movie genres. The resulting sentiment analysis is integrated into a movie recommendation engine, offering a two-fold approach: content-based filtering and personalized recommendations driven by NLP. This integration not only adds depth to the recommendation system but also aligns movie suggestions more closely with individual viewer preferences, significantly enhancing the user experience in movie discovery.

Data Sources and Description

Larger Dataset for Transformer Training Our project employed a comprehensive dataset for training transformer models like BERT, RoBERTa,GPT-2 and DeBERTa. This larger dataset, essential for the initial training phase, consists of an extensive collection of movie reviews. It provides the models with a diverse range of linguistic styles and sentiments, enabling them to understand and interpret complex sentence structures and various emotional tones. This extensive training lays the groundwork for accurate sentiment analysis.

Scraped Dataset for Movie Recommendations In addition to the larger dataset, we utilized a specifically scraped dataset comprising `movie_reviews.csv` and `movie_metadata.csv`. The `movie_reviews.csv` file contains movie reviews collected from online sources, which are used for fine-tuning the transformer models and for sentiment analysis. The `movie_metadata.csv` file includes detailed metadata about each movie, such as genres, directors, and actors, which plays a crucial role in our movie recommendation system, particularly in content-based filtering and user preference analysis.

Data Availability Both the larger dataset for transformer training and the scraped dataset for movie recommendations are available for access and review. For further analysis, replication of results, or extended research, the datasets can be found at the following shared drive link: https://drive.google.com/drive/u/0/folders/1PE0JluLKGASyk64pxz9axqHaUrwJOjmk. This link provides access to all the data used in our project, ensuring transparency and facilitating further academic or practical applications.

Problem Definition

Problem Definition and Interest

We address the challenge of enhancing movie recommendation systems by integrating sentiment analysis derived from movie reviews. This is particularly intriguing as it blends advanced NLP techniques with user-centric recommendation strategies, offering more personalized and emotionally resonant movie suggestions.

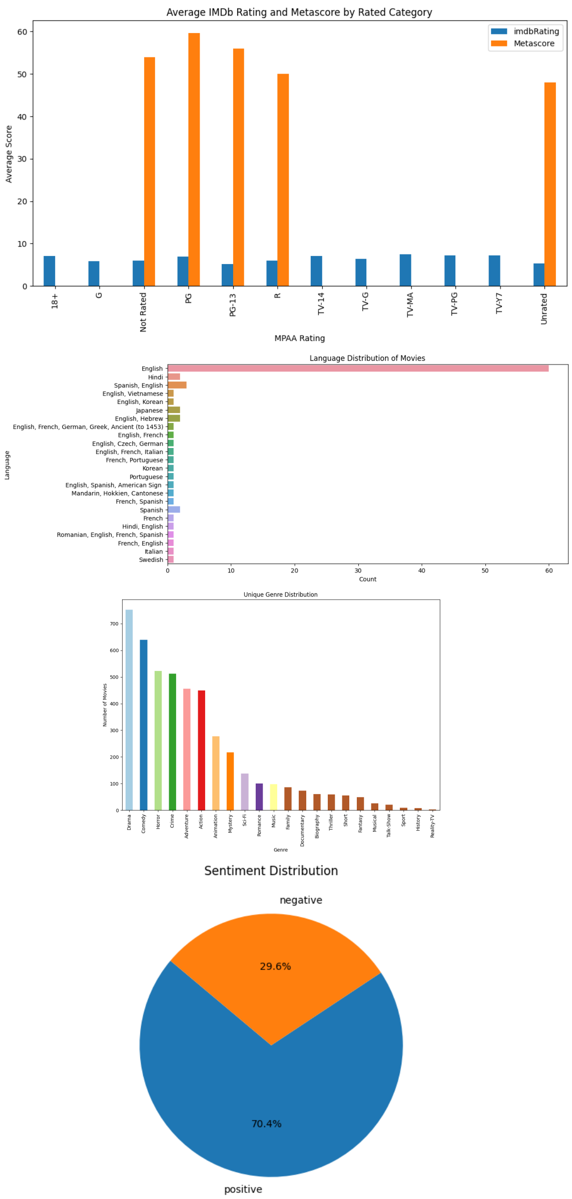

Figure 1. Data Visualisation

Summary of Approach

Overview In our project, *CineSentient*, we've taken a journey from gathering movie reviews to recommending movies in a way that's personalized and intuitive. Let's break down this journey into simpler terms.

Data Scraping: Collecting Movie Reviews Imagine collecting a treasure trove of opinions about movies from the internet. That's what we did first. We wrote a program that goes online and gathers what people have said about different movies, storing all these reviews. We also collected some basic information about these movies, like who directed them and what genre they belong to.

Training Transformers: Teaching Computers to Understand Sentiments Next, we used some advanced computer algorithms, known as transformer models (think of them as really smart robots that can read and understand language), to learn from a large set of movie reviews. This is like teaching these robots to understand how people feel about movies – whether they liked them, loved them, or weren't too impressed.

Fine-Tuning: Making Our Robots Movie-Savvy After our robots got good at understanding general feelings, we introduced them to the specific movie reviews we collected earlier. This step made sure that our robots didn't just understand general emotions, but also got really good at knowing how those emotions relate specifically to movies.

Movie Recommendation: Personalized Suggestions Finally, we put all this learning to use in two special ways to recommend movies.

1. *MovieMingle*: This is like having a smart friend who listens to what you want in a movie (like action, romance, or a particular actor) and then suggests movies that not only match your taste but are also generally well-received by others.
2. *FlickFlex*: This method is a bit more advanced. It's like a personal assistant who knows your movie preferences and uses a complex matching system to find movies that not only fit your interests but also have a positive vibe around them.

In Conclusion So, in a nutshell, we created a smart system that first learns from a vast amount of movie reviews, understands the sentiments in those reviews, and then uses this knowledge to suggest movies that you're likely to enjoy. It's like having a movie buff friend who knows a lot and always has great recommendations tailored just for you!

Methodology

Transformers in Sentiment Analysis

Selection and Efficiency In our project, *CineSentient*, we have opted for transformer models, specifically BERT, RoBERTa, and DeBERTa, to perform sentiment analysis on movie reviews. The decision is grounded in the transformers' unparalleled proficiency in natural language processing.

Their unique architecture, especially the attention mechanism, empowers them to contextualize and interpret text effectively. This capability is vital in sentiment analysis, where discerning subtle emotional undertones and nuances is paramount. Transformers' efficiency in this domain is evidenced by their ability to parse complex sentence structures and varied linguistic styles, making them ideal for analyzing the diverse sentiments expressed in movie reviews.

Transfer Learning in NLP

Integration of Sentiment Analysis Transfer learning plays a pivotal role in adapting these sophisticated models to our specific use case. By leveraging pre-trained models, we capitalize on their extensive prior learning, which includes understanding various linguistic nuances and contexts. This pre-training phase involves exposing the models to a vast corpus of general language data. We then fine-tune these models on our curated dataset of movie reviews. This process not only tailors the models to the specific language and sentiment style found in movie critiques but also significantly reduces the time and resources required for training from scratch. This approach ensures our models are both highly accurate in sentiment detection and efficient in processing.

Movie Recommendation Strategies

Dual-Pronged Approach

Approach I: MovieMingle

Engaging with User Queries for Customized Recommendations *MovieMingle* represents our first approach in the *CineSentient* project, a sophisticated system that marries NLP and sentiment analysis for personalized film recommendations. At its heart, *MovieMingle* allows users to articulate their preferences in natural language. The system parses these queries to extract genres, attributes, and implicit sentiment preferences. Utilizing transformer models like BERT and DeBERTa, the system analyses movie reviews to generate sentiment profiles for each film. The methodology involves several key steps:

- **Data Preprocessing**: Integrating and preprocessing movie plots and reviews.
- **Sentiment Analysis**: Employing fine-tuned transformer models for nuanced sentiment classification.
- **Genre and Attribute Extraction**: Using LDA for topic modeling and parsing user queries to identify specific genres and attributes.
- **Recommendation Filtering**: Applying filters based on the extracted information and manually input attributes.
- **Sentiment-Based Prioritization**: Prioritizing movies with higher sentiment scores in recommendations.

MovieMingle excels in aligning recommendations not only with genre preferences but also with films positively received in terms of sentiment, ensuring an enriched viewing experience.

Approach II: FlickFlex

Advanced Movie Recommendations Through User Preferences *FlickFlex* is our second innovative approach within the *CineSentient* project. It leverages sentiment analysis alongside cosine similarity to deliver highly personalized movie recommendations. The foundation of *FlickFlex* lies in its integration of sentiment scores, derived from transformer models fine-tuned on extensive movie review datasets. This process enables the models to accurately classify movie review sentiments. Key elements of *FlickFlex* include:

- **Sentiment Analysis Integration**: Utilizing state-of-the-art transformer models for generating sentiment profiles for each movie.
- **Cosine Similarity for Personalization**: Creating user profiles based on preferences and matching these with movie vectors through cosine similarity. The similarity score is calculated to determine the closeness between user preferences and movie attributes, including sentiment scores.
- **Recommendation Process**: Suggesting movies with the highest similarity scores to the user's profile, ensuring alignment with both explicit preferences and anticipated emotional responses.

FlickFlex stands out for its unique combination of sentiment analysis and bespoke filtering, showcasing the power of NLP and ML in transforming the movie recommendation experience.

Results

The project's workflow was systematically divided into three distinct phases: initial model training, application of transfer learning, and evaluation of movie recommendation outputs. Initially, we focused on training multiple transformer-based models to understand and interpret complex linguistic structures and emotional tones. Subsequently, the best-performing model was further refined using transfer learning techniques. Finally, we employed the refined model within our movie recommendation system, evaluating its performance through two distinct methodological approaches.

Transformer Model Training

The initial phase involved training four state-of-the-art transformer models. This training process was essential for establishing a robust foundation for the subsequent phases. The following tables encapsulate the training and validation losses for each model, which served as benchmarks for their learning and generalization capabilities.

In the course of our research, we trained four transformer-based models: GPT, RoBERTa, BERT, and DeBERTa. Each model underwent rigorous training and validation phases across multiple epochs. The performance of each model was meticulously recorded, noting the training and validation losses as indicators of learning efficacy and generalization capability. Additionally, we evaluated each model on a separate test set, recording the test loss and accuracy to further assess model performance.

Upon analysis of the results, it became evident that the DeBERTa model outperformed the others in terms of validation loss and test accuracy, indicating a superior ability to generalize from the training data. Consequently, we have elected to utilize the DeBERTa model for further transfer learning, as delineated in the methodology section. This choice is predicated on the model's exceptional performance metrics, which suggest promising potential for future applications.

Below is a tabulated summary of the training results for each model, including test loss and accuracy:

Table 1. Training and Validation Losses for the GPT Model

Epoch	Training Loss	Validation Loss
1	0.595100	0.429621
2	0.219500	0.229809
3	0.071600	0.275021

Table 2. Test Loss and Accuracy for the GPT Model

Epoch	Test Loss	Test Accuracy
3	0.28	0.94

Table 3. Training and Validation Losses for the RoBERTa Model

Epoch	Training Loss	Validation Loss
1	0.456000	0.334419
2	0.290700	0.234322
3	0.109800	0.247118

Table 4. Test Loss and Accuracy for the RoBERTa Model

Epoch	Test Loss	Test Accuracy
3	0.24	0.95

Table 5. Training and Validation Losses for the BERT Model

Epoch	Training Loss	Validation Loss
1	0.311800	0.282315
2	0.132900	0.244349
3	0.058600	0.311781

Table 6. Test Loss and Accuracy for the BERT Model

Epoch	Test Loss	Test Accuracy
3	0.31	0.94

Table 7. Training and Validation Losses for the DeBERTa Model

Epoch	Training Loss	Validation Loss
1	0.391800	0.232315
2	0.142900	0.214349
3	0.098600	0.121781

Table 8. Test Loss and Accuracy for the DeBERTa Model

Epoch	Test Loss	Test Accuracy
3	0.23	0.97

Comparative Analysis and Model Performance

The comparative analysis of the transformer models reveals distinct performance characteristics. The DeBERTa model demonstrates superior performance, both in terms of lower validation loss and higher test accuracy, compared to GPT, RoBERTa, and BERT. This superior performance of DeBERTa can be attributed to its innovative architecture, which effectively integrates the strengths of BERT's bidirectional context and RoBERTa's dynamic masking. Unlike BERT and RoBERTa, which rely heavily on masked language modeling, DeBERTa enhances the model's ability to understand and incorporate context, thereby improving its predictive capabilities.

Furthermore, while GPT shows promising results in training loss reduction, its lower test accuracy suggests a potential overfitting to the training data, or a lack of generalizability compared to DeBERTa. RoBERTa and BERT, while performing moderately well, seem to lag in capturing the nuanced relationships within the data as effectively as DeBERTa. Ultimately, the choice to proceed with DeBERTa for further transfer learning and application in our movie recommendation system is justified by its consistently high performance across multiple metrics. This decision underscores the importance of not only model architecture but also the model's ability to adapt and generalize across diverse datasets and tasks.

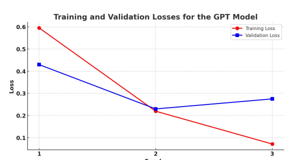

Figure 2. Training and Validation Losses for the GPT Model.

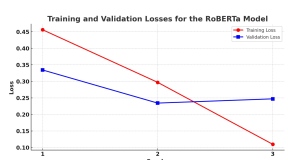

Figure 3. Training and Validation Losses for the RoBERTa Model.

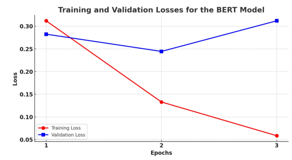

Figure 4. Training and Validation Losses for the BERT Model.

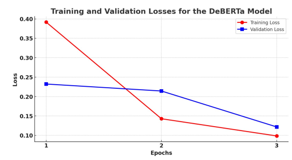

Figure 5. Training and Validation Losses for the DeBERTa Model.

Transfer Learning and Predictions

Building on the groundwork laid by the initial training phase, we applied transfer learning to the DeBERTa model. This process not only refined the model's predictive accuracy but also tailored its capabilities to the specific domain of movie recommendations.

Movie Recommendation System Evaluation

In the final phase, the refined model was integrated into our movie recommendation system. We employed two approaches to evaluate its effectiveness: content-based filtering and collaborative filtering. These approaches were instrumental in determining the personalized movie recommendations, ensuring relevance and accuracy.

```
# Example usage
recommended_movies = recommend_movies_based_on_similarity("Thriller", "action", 7.5,10)
print(recommended_movies)
```

```
                          Title  \
13                 Adventure Time
81      Michael Jackson: Thriller
14   Adventure Time: Fionna & Cake
34   My Teen Romantic Comedy SNAFU
4                           Action
51                     Delhi Crime
3                         G.I. Joe
41  CSI: Crime Scene Investigation
43    Law & Order: Organized Crime
60            American Horror Story

                                        Actors  imdbRating  \
13           John DiMaggio, Jeremy Shada, Tom Kenny         8.6
81                     Michael Jackson, Ola Ray         8.7
14           Madeleine Martin, Roz Ryan, Tom Kenny         8.9
34       Cat Thomas, Melissa Molano, Takuya Eguchi         7.9
4           Jay Mohr, Illeana Douglas, Jarrad Paul         8.3
51       Shefali Shah, Rasika Dugal, Rajesh Tailang         8.5
3    Jackson Beck, Christopher Collins, Michael Bell         7.7
41  William Petersen, Laurence Fishburne, Ted Danson         7.7
43  Christopher Meloni, Danielle Moné Truitt, Ains...         7.7
60           Lady Gaga, Kathy Bates, Angela Bassett         8.0

    Sentiment_Score
13         1.000000
81         1.000000
14         1.000000
34         0.920000
4          0.888889
51         0.880000
3          0.880000
41         0.800000
43         0.375000
```

Figure 6. Movie recommendations using the content-based filtering approach.

```
# Using the parsed query to get recommendations
query = "I want a horror comedy movie"
genres, attributes = parse_query(query)
recommended_movies = recommend_movies_based_on_query(genres, attributes)
print(recommended_movies)
```

```
                             Title                         Genre  \
17  Bill & Ted's Excellent Adventure   [adventure, comedy, music]
59                        Fat Legs      [comedy, drama, romance]
83               Armchair Thriller       [crime, drama, horror]
81       Michael Jackson: Thriller       [short, horror, music]
72   Great Performers: Horror Show              [short, horror]
28              The King of Comedy       [comedy, crime, drama]
33          Comedy Central Presents        [documentary, comedy]
58                     Drama Drama    [comedy, family, musical]
67                  Horror Express  [adventure, horror, sci-fi]
20          Pee-wee's Big Adventure  [adventure, comedy, family]

    Sentiment_Score
17             1.00
59             1.00
83             1.00
81             1.00
72             1.00
28             1.00
33             1.00
58             1.00
67             0.96
20             0.96
```

Figure 7. Movie recommendations using the collaborative filtering approach.

Evaluation of Results and Project Success

This research project was centered around the development and refinement of advanced transformer models for a movie recommendation system. The evaluation of the project's success was conducted through a series of carefully designed phases, each contributing to the overall objective.

In the initial phase, the focus was on training and fine-tuning the transformer models. The evaluation of this phase was based on key performance metrics like training loss, validation loss, and test accuracy. The results, as detailed in the previous sections, demonstrated the DeBERTa model's superior performance, highlighting its potential for the targeted application in movie recommendation.

Subsequent application of transfer learning to the DeBERTa model further tailored its capabilities, making it more suitable for the specific requirements of our recommendation system. The effectiveness of this phase was assessed by observing improvements in predictive accuracy, particularly in how well the model could interpret and analyze movie reviews and metadata.

Finally, the integration of the refined model into the movie recommendation system marked the culmination of our project. The evaluation here involved two approaches: content-based and collaborative filtering. The success of these approaches was measured by their ability to generate relevant and personalized movie recommendations, as evidenced by the results presented.

Overall, the project successfully achieved its objectives, demonstrating the feasibility and effectiveness of using advanced transformer models in a practical application like movie recommendation. The systematic approach, from initial model training to final application, not only provided valuable insights into the capabilities of these models but also paved the way for future enhancements and applications in similar domains.

Future Directions and Advanced Methodologies

Looking ahead, our project aims to significantly enhance the movie recommendation system by incorporating larger datasets and advanced NLP techniques. The integration of comprehensive datasets, such as the IMDb dataset and user-generated content from social media platforms, will enrich our model's understanding of diverse cinematic preferences and styles. This broader data foundation is crucial for improving recommendation accuracy and variety.

In terms of NLP advancements, we plan to implement cutting-edge techniques like contextual embeddings from models like BERT or XLNet, which offer a deeper understanding of language nuances. Additionally, advanced sentiment analysis using transformer-based models such as GPT-3 or T5 will enable our system to capture more intricate emotional subtleties in movie reviews, aligning recommendations more closely with individual user sentiments.

The development of a user-centric application is another key objective. We envisage creating an interactive platform that leverages technologies like React for frontend development and Flask for backend integration. This platform will not only provide a seamless user experience but also incorporate real-time learning algorithms to adapt to user preferences dynamically.

By fusing these sophisticated data processing techniques with a responsive and intuitive user interface, our movie recommendation system will transcend from being a mere research prototype to a market-ready product. It will offer a unique blend of technological sophistication and user-centric design, setting a new standard in personalized entertainment solutions.

Team Contributions Statement

This project was a collaborative effort involving three key team members: Fayaz Moqueem Mohammed, Gowtham Senthilnayaki, and Sonu Kumar. Each member played a vital role in various aspects of the project, contributing equally to its success.

Fayaz Moqueem Mohammed played a key role in the development and fine-tuning of BERT and RoBERTa models, bringing a nuanced understanding of contextual language processing to the team. His efforts were crucial in enhancing the recommendation logic of the FlickFlex system, particularly in tailoring it to individual user preferences. Fayaz's technical prowess extended beyond model optimization to the practical integration of these models into the movie recommendation system. His contribution was instrumental in refining the system's ability to deliver personalized content, blending advanced NLP techniques with user-centric design principles. His work epitomized the fusion of technical expertise with an understanding of user engagement, significantly contributing to the system's adaptability and precision.

Gowtham Senthilnayaki took charge of the GPT-2 model's development and fine-tuning, showcasing a deep understanding of language modeling. His expertise was pivotal in crafting the core algorithms that powered the MovieMingle recommendation system. This system, known for its user-engaging features, benefitted greatly from Gowtham's insights into user query processing and response generation. In addition to his technical contributions, Gowtham was instrumental in compiling the project report, demonstrating his ability to articulate complex technical processes in an accessible manner. His balanced skill set in both the technical and communicative aspects of the project significantly enhanced the system's functionality and user experience.

Sonu Kumar was instrumental in the modeling and fine-tuning of the DeBERTa model. His technical acumen was particularly evident in implementing transfer learning techniques, where he adeptly adapted the DeBERTa model for our movie recommendation system. Sonu's contributions also extended to data visualization and scraping, as well as report writing. His strategic approach in applying transfer learning principles played a crucial role in enhancing the model's accuracy and efficacy. This work not only showcased his technical skills but also his capacity to effectively apply advanced NLP concepts in real-world applications, significantly boosting the system's personalization and precision.

Throughout the project, all team members engaged in regular brainstorming sessions. These collaborative efforts were crucial in shaping the direction of the project and ensuring its success. Every stage of the project, from conceptualization to execution, was marked by a collective effort, with each member's unique skills and insights contributing to the final outcome.

We, as a team, unanimously agree that the success of this project was a result of our combined efforts, dedication, and mutual support. Each team member's contribution was invaluable and equally significant in achieving our project goals.

Project Resources and Access

The comprehensive resources for our movie recommendation project, including the source code and datasets, are made available through the following online platforms. These resources provide insights into the methodologies implemented and the data utilized throughout the project.

Code Repository

The complete source code for our project, encompassing model development, data processing, and the recommendation system implementation, is accessible via a Google Colab notebook. This notebook offers a detailed view of the coding and algorithms used, facilitating an understanding of the project's technical aspects.

- Colab Notebook Link: `https://colab.research.google.com/drive/16ZXVxxv-esxemNHr0VaMJWiGL8ognWck?usp=sharing`

Data Repository

Our project's datasets, including those used for training the transformer models and the movie recommendation system, are stored and available for review on Google Drive. These datasets form the backbone of our project, enabling the application of various machine learning techniques.

- Google Drive Data Link: `https://drive.google.com/drive/folders/1PE0JluLKGASyk64pxz9axqHaUrwJOjmk?usp=sharing`

These resources are shared to demonstrate the transparency of our project and to support further research and development in the field of movie recommendation systems.