DS675-101-Machine Learning  
Final Project Report

Netflix Recommendation System using BERT

Team-

Harshith Gade (hg355)

Sai Nikhil Dunuka (sd2279)

Sritanvi Thirunagari (st868)

Link to video presentation:

<https://drive.google.com/file/d/1RP_HvHqj5ZUQBlSgEqcin5I9VeYYJf-t/view?usp=drive_link>

Link to the code:

<https://colab.research.google.com/drive/1QJgWvOXCXQnIq_2YvkBtMy6SywtSbwci#scrollTo=6ff46539>

### 

### **Netflix Recommendation System using BERT**

**Introduction:**

Netflix Inc. is an American media company based in Los Gatos, California. Founded in 1997 by Reed Hastings and Marc Randolph in Scotts Valley, California, it operates the over-the-top subscription video on-demand service Netflix brand, which includes original films and television series commissioned or acquired by the company, and third-party content licensed from other distributors.

**About this Dataset:** [Netflix](https://en.wikipedia.org/wiki/Netflix) is one of the most popular media and video streaming platforms. They have over 8000 movies and TV shows available on their platform, as of mid-2021, they have over 200M subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration, etc.

Descriptive statistics were computed for each variable as part of the analysis, and visualizations were made to investigate the relationships between the various variables. We created a number of graphs, such as the scatterplot, distplot, count plot, bar plot, pair plot, heatmap, pie plot, and box plot to gain insight from the dataset.

Link of the dataset: <https://www.kaggle.com/datasets/shivamb/netflix-shows>

## 

## 

## 

## **Data Description**

* **show\_id:**Unique identifier for each Movie/TV Show. It's a distinct number or code assigned to every entry in the dataset, helping to distinguish between different shows or movies.
* **type:**Identifies whether the entry is a Movie or a TV Show. It's a categorical variable indicating the format of the content.
* **title:**The name of the Movie or TV Show, serving as a human-readable identifier for the content.
* **director:**The person responsible for directing the movie or TV show. This attribute identifies the individual leading the creative and artistic aspects of the production.
* **cast:**Lists the actors involved in the movie or TV show. It provides information about the key performers in the production.
* **country:**Specifies the country where the movie or show was produced. It indicates the origin or location of the production.
* **date\_added:**The date when the content was added to Netflix. This attribute helps in understanding the timeline of when the content became available on the platform.
* **release\_year:**Represents the actual release year of the movie or TV show. It indicates when the content was originally released to the public.
* **rating:**TV rating assigned to the movie or show. It provides information about the target audience age group and suitability.
* **duration:**Specifies the total duration of the content, either in minutes (for movies) or the number of seasons (for TV shows). It gives an idea of the time investment required to watch the content.
* **listed\_in:**Describes the genre or genres to which the movie or show belongs. It categorizes the content based on its thematic elements.
* **description:**A brief summary or description of the movie or TV show. It provides a concise overview of the content's plot or theme.

Few potential correlations between the attributes:

* **Director and Rating:**

Check if there's a correlation between the director of a movie/show and its overall rating. Some directors consistently produce highly-rated content.

* **Release Year and Viewer Ratings:**

Explore if there's a correlation between the release year and the viewer ratings. It could indicate whether newer or older content tends to be more well-received.

* **Cast Popularity and Viewer Ratings:**

Analyze if there's a correlation between the popularity of the cast and the overall rating. A star-studded cast might attract more viewers and potentially lead to higher ratings.

* **Country and Genre:**

Investigate if there's a correlation between the country of production and the genre of the movie/show. Certain countries might be known for producing specific genres.

* **Date Added and Viewer Engagement:**

Check if there's a correlation between the date a movie/show is added to Netflix and its subsequent popularity or rating. This could provide insights into viewer engagement patterns.

**What is the recommendation system using BERT+MLP like?**

A movie/series recommendation system using BERT and MLP is a type of recommendation system that utilizes two powerful artificial intelligence techniques:

BERT:

* Bidirectional Encoder Representations from Transformers (BERT) is a pre-trained deep learning model that excels at understanding the nuances of natural language. In the context of movie/series recommendations, BERT can process and extract meaningful features from text data such as reviews, descriptions, and dialogue. This allows the model to better understand the content and themes of each movie/series, and consequently, make more accurate recommendations.

MLP:

* Multilayer Perceptron (MLP) is a type of artificial neural network with multiple layers of interconnected nodes. MLPs are adept at learning complex relationships between features and can be trained to predict specific outcomes. In a recommendation system, an MLP can learn the relationship between user preferences (e.g., past watch history, ratings) and movie/series features extracted by BERT. This allows the system to predict which movies/series a particular user is likely to enjoy.

**Why a recommendation system using BERT+MLP?**

There are several reasons why we use a recommendation system using BERT and MLP:

1. Increased Accuracy:

* BERT's natural language understanding: BERT excels at understanding the nuances of text, capturing the meaning and sentiment of reviews, descriptions, and dialogue. This allows the system to extract more relevant features than traditional methods, leading to more accurate recommendations that match users' interests and tastes.
* MLP's learning capabilities: MLPs are powerful tools for learning complex relationships between features. In this case, the MLP can learn the intricate connections between extracted features and user preferences, leading to personalized recommendations that cater to individual needs.

2. Enhanced Personalization:

* Tailoring recommendations: BERT and MLP enable the system to move beyond simple collaborative filtering approaches. Instead of solely relying on other users' ratings, the system can consider each user's unique preferences and watch history, providing recommendations that are truly relevant and enjoyable for each individual.
* Understanding user intent: BERT's ability to analyze text allows the system to understand user intent beyond explicit ratings. For example, it can analyze reviews and identify subtle cues that reveal what aspects of a movie/series a user might enjoy or dislike. This allows the system to make nuanced recommendations that align with individual preferences.

3. Improved Scalability:

* Handling large datasets: BERT and MLP models can be efficiently trained on massive datasets containing millions of movies/series and users. This makes them suitable for real-world applications where dealing with large volumes of data is essential.
* Continuous learning and adaptation: These models can continuously learn and adapt as new data becomes available. This ensures that the recommendations remain relevant and up-to-date, reflecting the evolving preferences and interests of users.

4. Addressing limitations of traditional approaches:

* Traditional recommendation systems often suffer from the cold start problem: When new users or items lack sufficient data, the system struggles to make accurate recommendations. BERT and MLP, by leveraging their understanding of content and user preferences, can overcome this challenge and provide personalized recommendations even for new users and items.
* Traditional systems might rely on explicit ratings, which can be biased or incomplete. BERT and MLP, by analyzing textual data, can capture implicit preferences and interests, leading to a more comprehensive and accurate understanding of user preferences.

In summary, using BERT and MLP in a recommendation system offers several advantages over traditional methods. It significantly improves the accuracy and personalization of recommendations, making it a powerful tool for enhancing user experience and engagement with movies/series.

**How does the recommendation system using BERT+MLP works?**

A movie/series recommendation system using BERT and MLP works through a series of steps:

1. Data Preprocessing:

* The first step involves gathering and preparing the necessary data. This includes:
  + Movie/series information: This includes titles, descriptions, genres, directors, actors, user ratings, and reviews.
  + User data: This includes user profiles, viewing history, ratings, and preferences.

2. BERT Feature Extraction:

* Preprocessed text data (movie/series descriptions and reviews) is fed into a pre-trained BERT model.
* BERT analyzes the text and extracts meaningful features that capture the content, themes, sentiment, and other relevant information about each movie/series.
* These features are essentially numerical representations of the text, encapsulating the important information for recommendation.

3. User Preference Model:

* User data is used to train an MLP model.
* The MLP learns the complex relationships between user features (e.g., demographics, viewing history, ratings) and their preferences for different types of movies/series.
* This allows the model to create a personalized profile for each user, capturing their individual tastes and interests.

4. Recommendation Generation:

* The system combines the extracted features from BERT and the user profiles learned by the MLP.
* This information is used by a prediction model (often another MLP) to predict which movies/series a particular user is likely to enjoy.
* The prediction model essentially compares the extracted features of each movie/series to the user's profile, identifying those that best match their preferences.

5. Refinement and Ranking:

* The initial recommendations generated by the prediction model might be further refined based on additional factors:
  + Popularity: Popular movies/series with high ratings might be prioritized.
  + Novelty: Recommending new and interesting content can be beneficial to avoid boredom.
  + Diversity: Presenting a diverse range of options based on genre, theme etc can broaden the user's horizons.

6. Recommendation Delivery:

* The system presents the final recommendations to the user through various interfaces, such as personalized recommendation lists, streaming service interfaces, or online platforms.

This is a simplified overview, and the specific implementation details might vary depending on the chosen tools and algorithms. However, the core principles remain the same: using BERT to understand content and extract features, using MLP to learn user preferences, and combining these insights to generate personalized recommendations for each user.

**STEPS INVOLVED TO ACHIEVE THIS**:

There are several steps involved in making a movie/series recommendation system using BERT and MLP:

1. Data Collection:

* Movie/series data:
  + Obtain movie/series information from online databases like TMDB, IMDB, or OpenSubtitles.
  + Collect data such as titles, descriptions, genres, directors, actors, release dates, and user ratings.
  + Scrape reviews from websites or utilize public review datasets.
* User data:
  + Collect user demographics, viewing history, ratings, and preferences through surveys, user profiles, or tracking user activity on the system.

2. Data Preprocessing:

* Clean and filter data: Remove duplicates, missing values, and irrelevant information.
* Preprocess text data: Apply techniques like tokenization, stemming, and lemmatization to prepare text for BERT processing.
* Encode categorical features: Convert categorical data like genres and directors into numerical representations.
* Feature engineering: Create additional features based on domain knowledge or user behavior analysis.

3. BERT Feature Extraction:

* Train or fine-tune a pre-trained BERT model on the text data (movie/series descriptions and reviews).
* Extract features from the BERT model that capture the content, themes, sentiment, and other relevant information about each movie/series.
* These features can be word embeddings, sentence embeddings, or pooled representations generated by the BERT model.

4. User Preference Modeling:

* Train an MLP model on the user data, including features like demographics, viewing history, and ratings.
* The MLP learns the relationships between these features and user preferences for different types of movies/series.
* This model creates a personalized profile for each user, capturing their individual tastes and interests.

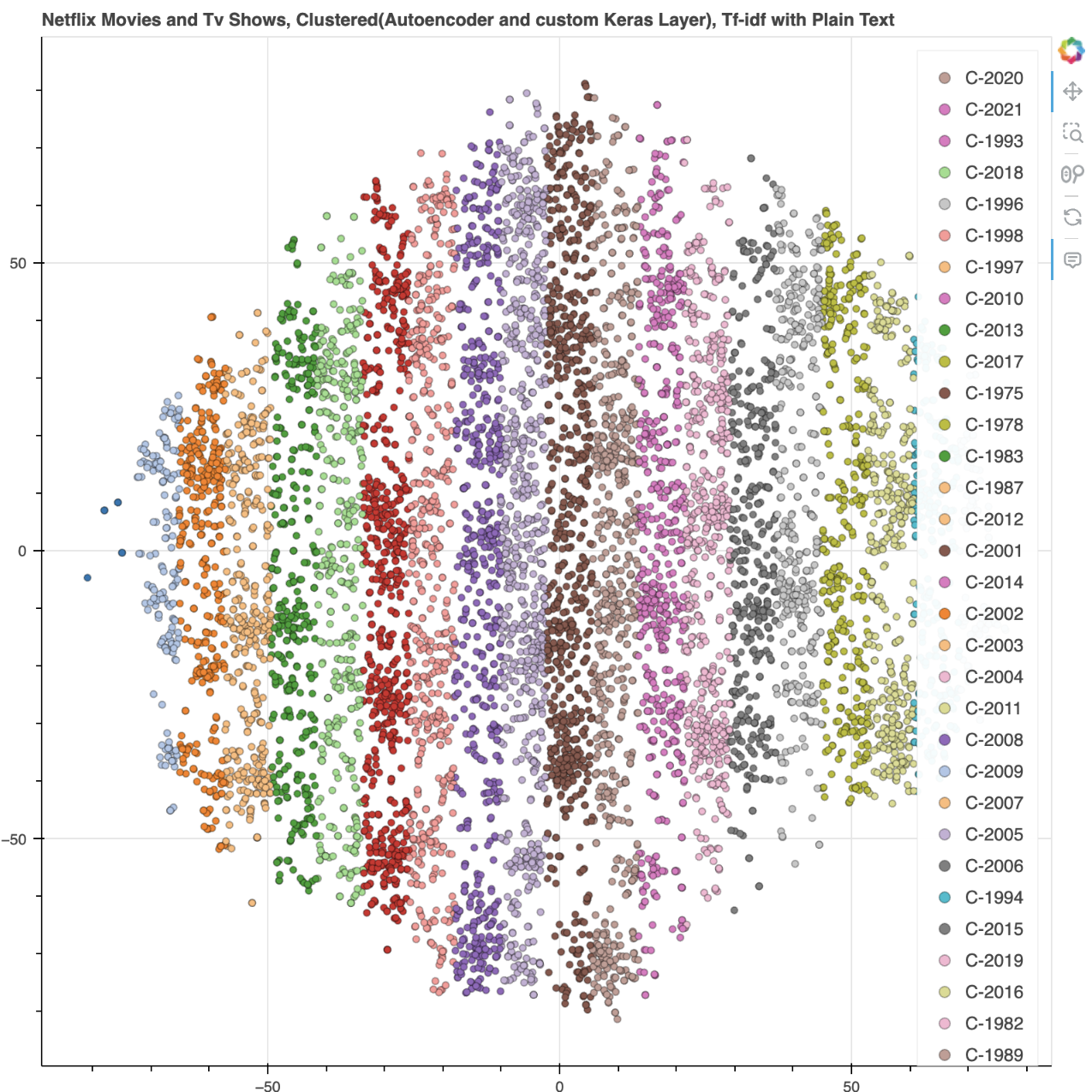
5. Recommendation Generation:

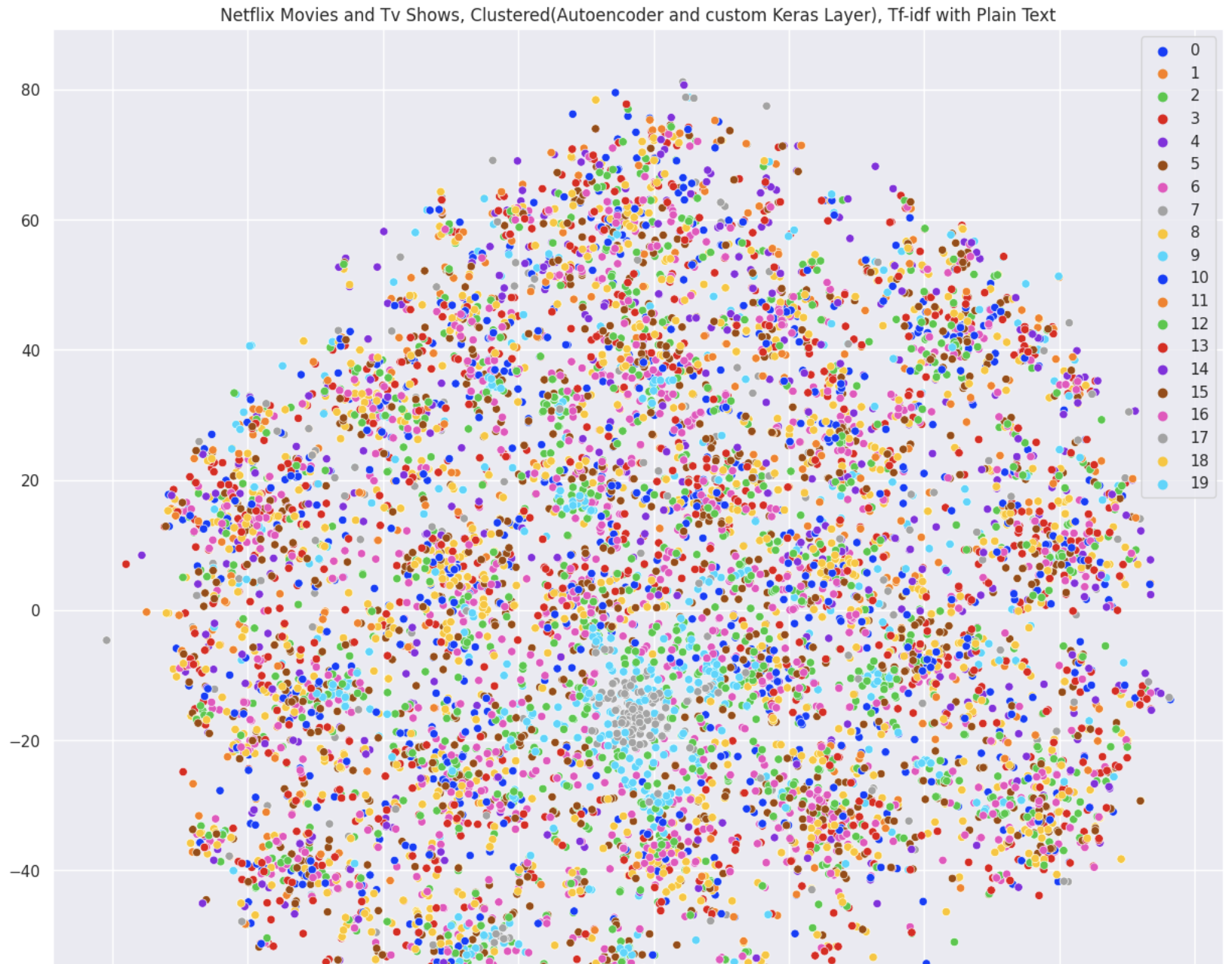
* Combine the extracted features from BERT with the user profiles learned by the MLP.
* Train a prediction model on this combined data.
* The prediction model learns to predict which movies/series a particular user is likely to enjoy, comparing the extracted features to the user's profile.

**CLUSTERING:**

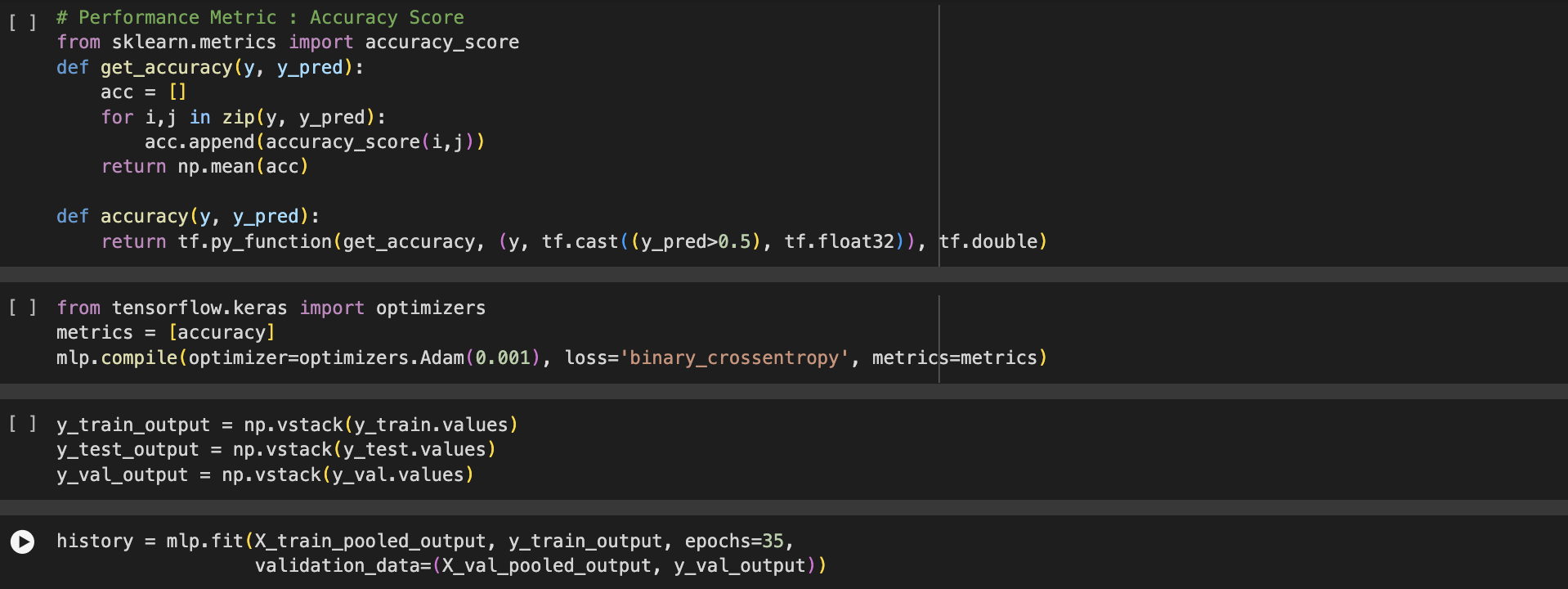
Clustering is an unsupervised learning technique that groups similar data points together based on their features or similarities. It's used to discover natural patterns or structures within data without needing predefined categories. This method is valuable in various fields such as customer segmentation, anomaly detection, and image segmentation. Clustering algorithms like K-means, hierarchical clustering, and DBSCAN are used to partition datasets into clusters, enabling better understanding and organization of data.

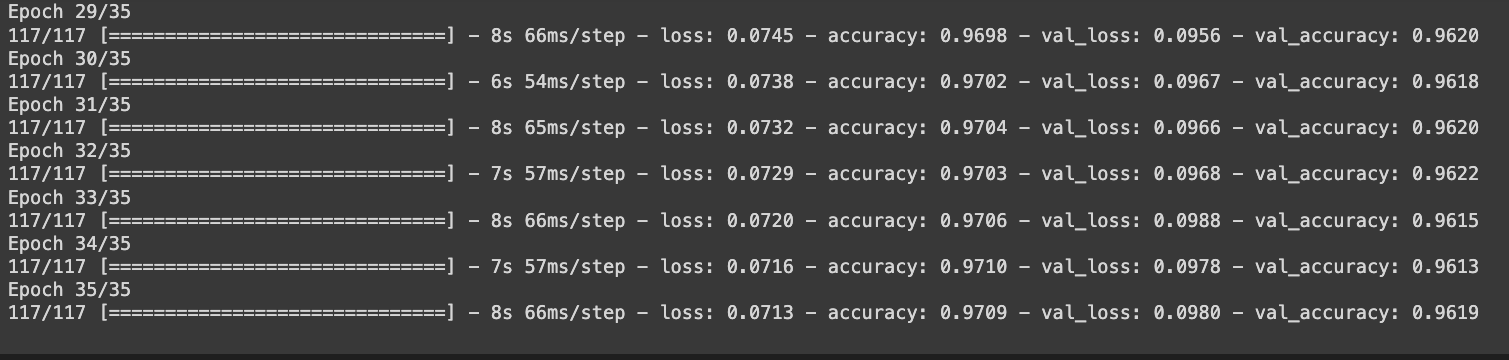
We created a scatterplot which contains title, cast, directors and description. We clustered the dataset based on the release years attributes and tried it with various epochs and iterations.

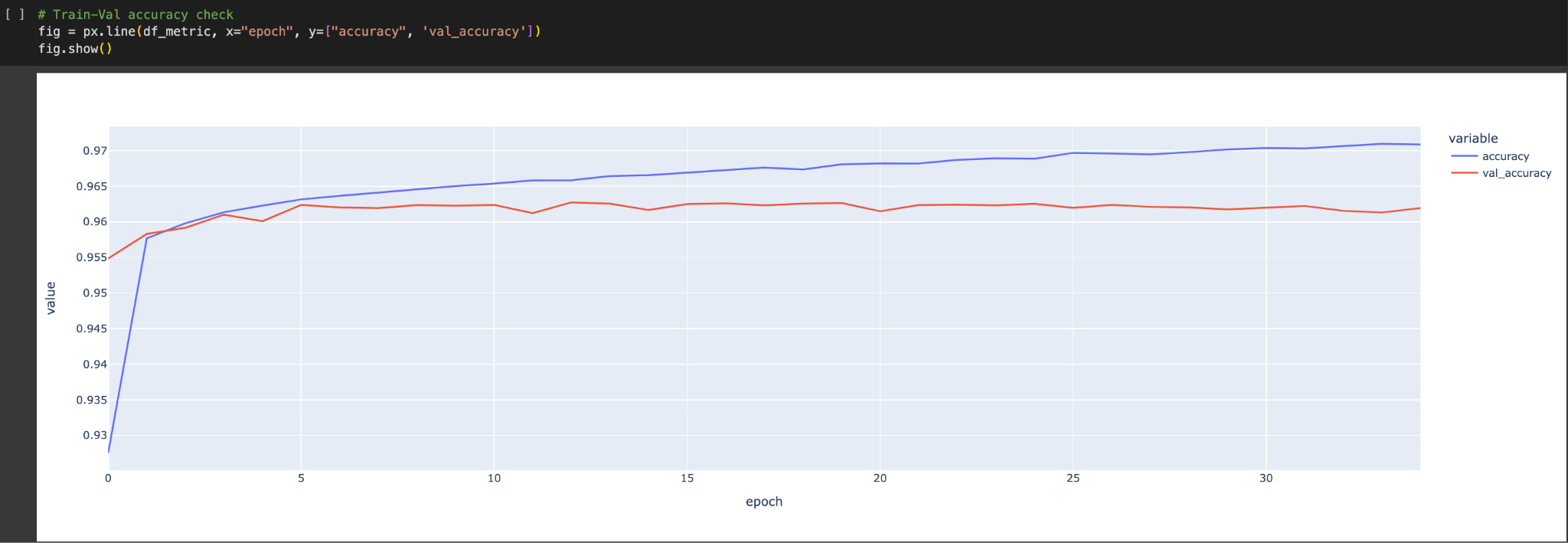


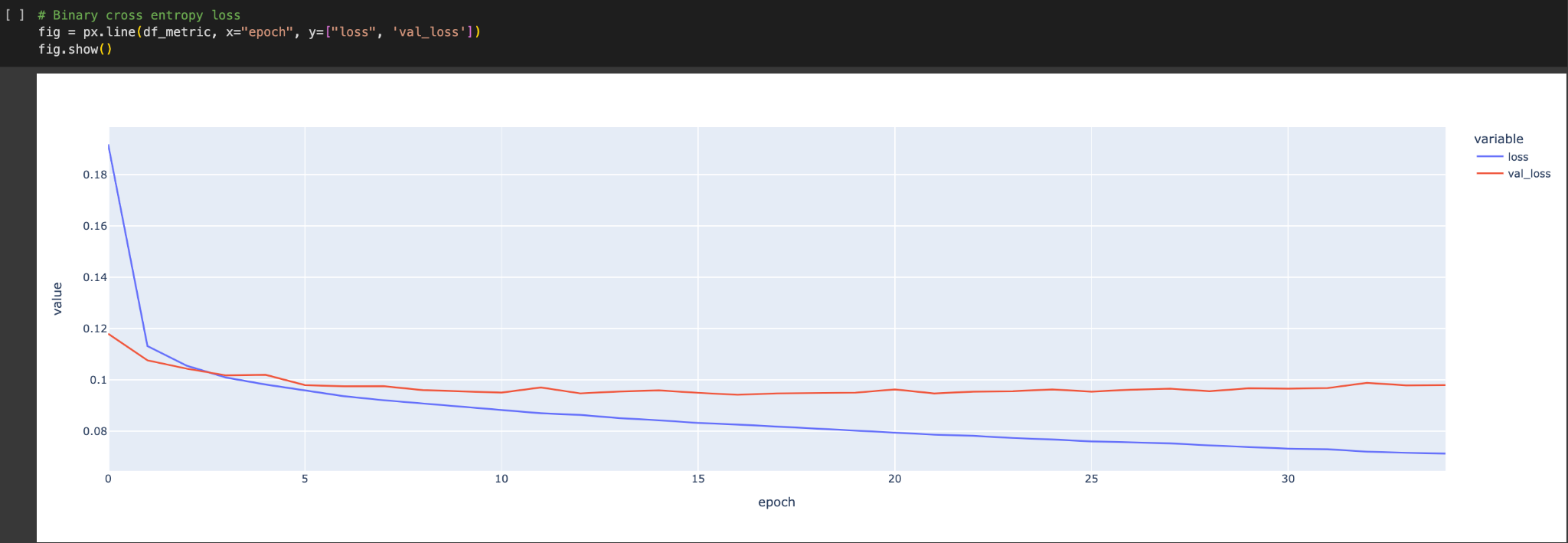


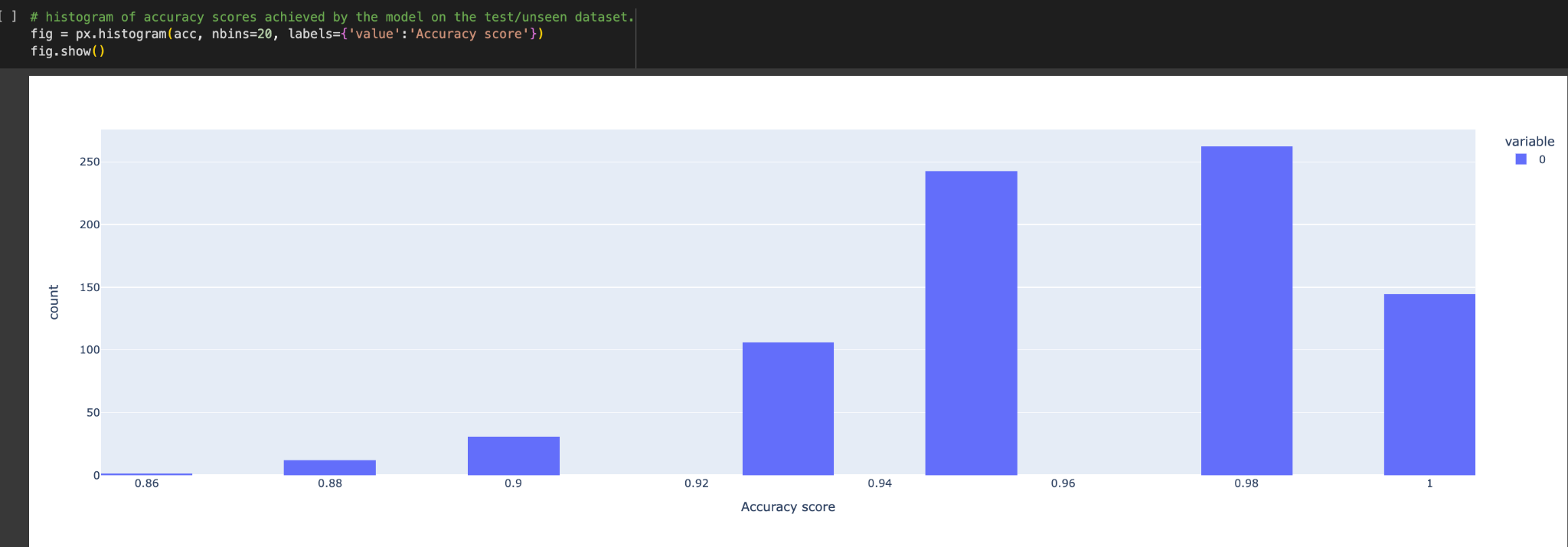
**OUTPUT OF OUR ML MODEL:**

****

****

****

****

****

**Few tasks/ideas we tried in this project:**

* Explored the Netflix-shows dataset through visualizations and graphs using matplotlib and seaborn
* Found few correlations between the attributes of the dataset
* Built a multilabel classifier using **BERT and MLP** to predict the Genre of a show using its description
* We added a few dense layers in it and enhanced the accuracy of our model.

**What differentiates our work from Kaggle:**

* Most of the work you find on Kaggle related to this dataset is Data Visualizations, prediction etc
* We on the other hand built a multilabel classifier using BERT and MLP
* We took both the pooled output and sequence outputs and we observed more efficiency in pooled outputs.
* We did a comparison between MLP and clustering and found more accurate results in MLP.
* We added some dense layers with RELU gradient and our accuracy was enhanced. We got accuracy rate around 96-97%

**CONCLUSION:**

A movie/series recommendation system using BERT and MLP offers significant advantages over traditional methods. By combining the power of natural language understanding with user preference learning, it delivers a more accurate, personalized, and scalable recommendation experience

Overall, a BERT+MLP recommendation system provides:

* A deeper understanding of content and user preferences.
* Recommendations that are truly relevant and enjoyable for each individual.
* A more engaging and satisfying movie/series exploration journey.

Looking ahead:

* Further research and development are expected to improve the accuracy, personalization, and efficiency of these systems.
* Integration with other AI technologies like image recognition and sentiment analysis can further enhance the recommendation experience.
* The future of movie/series recommendations lies in personalized, intelligent systems that understand users and their preferences on a deeper level.