Content-Based Filtering for Netflix Title Recommendations

Group 8

Grace Esther (2702305576), Adhi Swasono Aryaning Bawono (2702269620),

Axel Dimas Anugrah (2702274015)

---

1. Introduction

This project presents the design and implementation of a content-based filtering recommendation system applied to Netflix titles. The primary objective is to suggest similar titles based on intrinsic item features rather than user interaction history. By relying exclusively on the metadata of content items, such as textual descriptions, genres, cast, directors, and country of origin, the system provides item-to-item recommendations that are interpretable and free from cold-start problems typically associated with collaborative approaches.

The recommendation problem addressed in this work centers on the challenge of identifying relevant titles for users when explicit user-item interaction data is either sparse or unavailable. Unlike collaborative filtering methods that depend heavily on historical user preferences and often suffer from cold-start limitations, content-based filtering leverages item attributes to generate recommendations. This approach not only enhances interpretability but also allows recommendations for newly added titles without prior user feedback.

The dataset used in this project consists of 8,807 Netflix titles, encompassing both movies and TV shows, with metadata fields such as director, cast, country, release year, rating, duration, genres (listed_in), and textual descriptions. The dataset exhibits some missing values, particularly in the director and cast fields, which were addressed through imputation and preprocessing strategies. The release years range from 1925 to 2021, with a mean release year around 2014, indicating a broad temporal coverage of content.

Initial data analysis revealed a heterogeneous distribution of content types and metadata completeness, necessitating robust preprocessing steps including normalization, tokenization, lemmatization, and stopword removal to prepare textual features for vectorization. Feature weighting was applied to emphasize more descriptive metadata, such as descriptions and genres, to improve recommendation quality.

The development process is modularized, encompassing a notebook-based prototype for data exploration and model building, object-oriented scripts (oop.py and infer.py) for reusable logic and inference, and a Streamlit interface (app.py) which facilitates interaction with the system through a deployed web application.

2. Methodology

2.1 Dataset and Preprocessing

The dataset used is netflix_titles.csv, a publicly available collection containing metadata for films and television shows available on the Netflix platform. Relevant fields for modeling include title, description, director, cast, genres, country, rating, and release_year.

Textual preprocessing was conducted using tools from the NLTK library. The process involved normalization through lowercasing and removal of special characters, followed by tokenization and part-of-speech tagging. Lemmatization was performed using WordNetLemmatizer, informed by POS tags to ensure grammatical accuracy. Stopwords and tokens shorter than three characters were excluded. This process was applied to each text-based feature to produce a clean corpus suitable for vectorization.

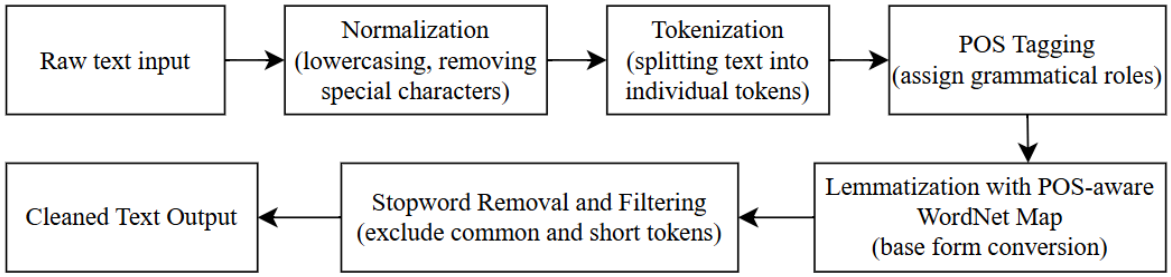

Fig. 1. Text preprocessing pipeline

2.2 Feature Engineering and Vectorization

Each cleaned textual field was vectorized using Term Frequency–Inverse Document Frequency (TF-IDF). Independent vectorizers were fitted for description, genres, director, cast, and country, with a specified maximum number of features to limit dimensionality.

To reflect the varying semantic importance of different metadata fields, feature vectors were scaled by domain-informed weights. The description field was assigned the highest weight, followed by genres, director, cast, and country.

Metadata Field	Features	Weight
Description	3000	1.0
Genres	500	0.8
Director	300	0.5
Cast	500	0.4
Country	200	0.2

Table 1. TF-IDF feature allocation summary

After weighting and horizontal stacking, the feature matrix was reduced in dimensionality using Truncated Singular Value Decomposition (SVD) with 200 components. This transformation preserved essential semantic variance while improving computational efficiency. Finally, the resulting vectors were normalized using L2 norm to ensure compatibility with cosine similarity measures.

2.3 Similarity Computation and Recommendation Logic

For scalability, cosine-based nearest-neighbor search was implemented using NearestNeighbors from Scikit-learn. If this module was unavailable, pairwise cosine similarity was computed directly. For each query, the system identifies the top-N most similar items by vector proximity.

If the input title is present in the dataset, its corresponding vector is used to retrieve the nearest neighbors. If the title is not found, the system invokes a cold-start fallback strategy. This approach selects recent titles from the dataset and performs similarity search based on genre vectors, ensuring that thematically relevant results are still returned.

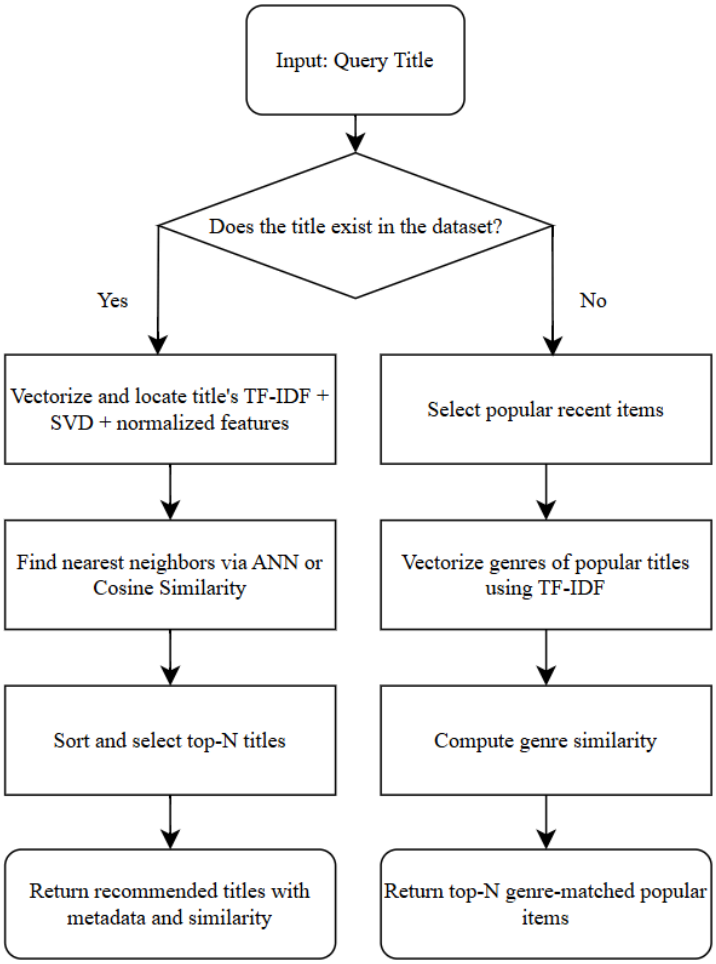

Fig. 2. Flowchart of the recommendation decision logic

3. Results and Analysis

3.1 Qualitative Evaluation

In the absence of user interaction data or explicit ratings, the system was evaluated qualitatively by examining the relevance of its output to a variety of input queries. For well-known titles such as Stranger Things, the recommendations included shows with supernatural themes, adolescent characters, and science fiction elements. This supports the claim that the model effectively captures thematic proximity in the vector space.

When the system was queried with a non-existent title, the genre-based fallback successfully returned content with similar classification labels, although without the precision achieved for known titles. This behavior demonstrates the system's robustness in handling unknown inputs.

```
[ ] recommend("Stranger Things")
```

	title	director	cast	genres	country	release_year	rating	description	similarity
3187	Nightflyers	unknown	eoin macken david ajala jodie turnersmith angu...	horror mystery scifi fantasy	united state	2018	TV-MA	humankind future stake group scientist powerfu...	0.837727
6953	Helix	unknown	billy campbell hiroyuki sanada kyra zagorsky m...	horror mystery scifi fantasy	united state canada	2015	TV-MA	investigate possible outbreak arctic research ...	0.791089
1473	Chilling Adventures of Sabrina	unknown	kiernan shipka ross lynch miranda otto lucy da...	horror mystery scifi fantasy	united state	2020	TV-14	magic mischief collide halfhuman halfwitch sab...	0.773480
5287	The Vampire Diaries	unknown	nina dobrev paul wesley ian somerhalder steven...	drama mystery scifi fantasy	united state	2016	TV-14	trap adolescent body feud vampire brother stef...	0.737511
2303	Warrior Nun	unknown	alba baptista toya turner lorena andrea kristi...	action adventure mystery scifi fantasy	united state	2020	TV-MA	wake morgue orphan teen discovers possess supe...	0.721433

```
[ ] recommend("Non-Existent Movie")
Title 'non-existent movie' not found. Recommending popular items by genre...
```

	title	director	cast	genres	country	release_year	rating	description	similarity
1378	Penguin Bloom	glendyn ivin	naomi watt andrew lincoln jacki weaver griffin...	child family movie dramas	australia united state	2021	TV-14	mom cop aftermath harrowing accident find insp...	1.000000
1372	June & Kopi	noviandra santosa	acha septriasa ryan delon makayla rise hilli	child family movie dramas international movie	indonesia	2021	TV-PG	street dog take young couple family pit become...	0.938086
80	Firedrake the Silver Dragon	tomer eshed	thomas brodiesangster felicity jones freddie h...	child family movie	unknown	2021	TV-Y7	home threaten human young dragon summons coura...	0.898603
64	Nightbooks	david yarovesky	winslow fegley lidya jewett krysten ritter	child family movie	unknown	2021	TV-PG	scary story fan alex must tell spinetingling t...	0.898603
23	Go! Go! Cory Carson: Chrissy Takes the Wheel	alex woo stanley moore	maisie benson paul killam kerry gudjohnsen lim	child family movie	unknown	2021	TV-Y	arcade game sled day hiccup cure cory carson c...	0.898603

Fig. 3. Example recommendation results for selected queries, including both recognized and unrecognized titles

3.2 Feature Impact Analysis

Field-wise weighting played a critical role in enhancing the system’s accuracy. Informal ablation tests revealed that omitting the description field significantly reduced semantic relevance, while removing country or cast had relatively minor effects. This validates the chosen weighting scheme.

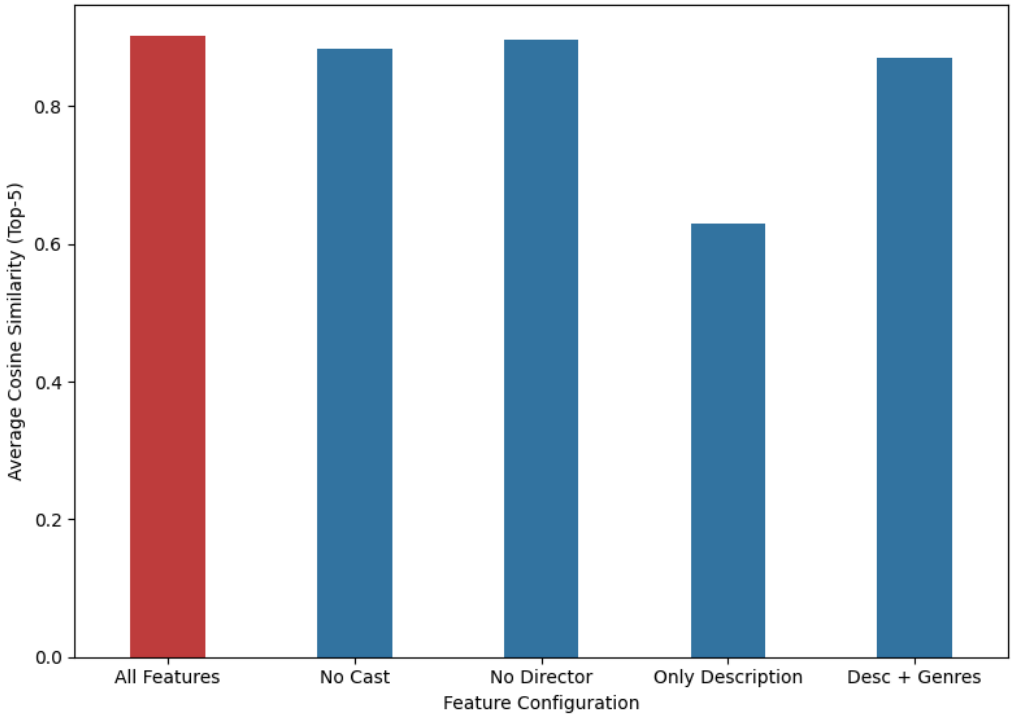

Fig. 4. Impact of feature combinations on recommendation similarity

4. Deployment Process

4.1 Software Architecture

The system was developed using a layered architecture. Initial modeling was performed in a Jupyter notebook to support iterative development. The finalized logic was transferred to modular Python scripts. The file oop.py contains object-oriented classes for preprocessing and feature construction. The file infer.py handles model loading and inference. These modules encapsulate reusable logic and decouple the modeling pipeline from the user interface.

4.2 Streamlit Integration

A user-friendly interface was implemented using the Streamlit framework in the app.py script. The application allows users to input a title and receive a ranked list of recommended content along with relevant metadata and similarity scores. Input validation and error handling are included to address unknown titles via the genre-based fallback mechanism.

4.3 Deployment Outcome

The Streamlit application was successfully deployed to Streamlit Community Cloud, making it publicly accessible. The interface is responsive and delivers real-time recommendations,

effectively bridging the gap between technical implementation and user experience. The final application is publicly accessible at: https://netflix-hybrid-recommender.streamlit.app/

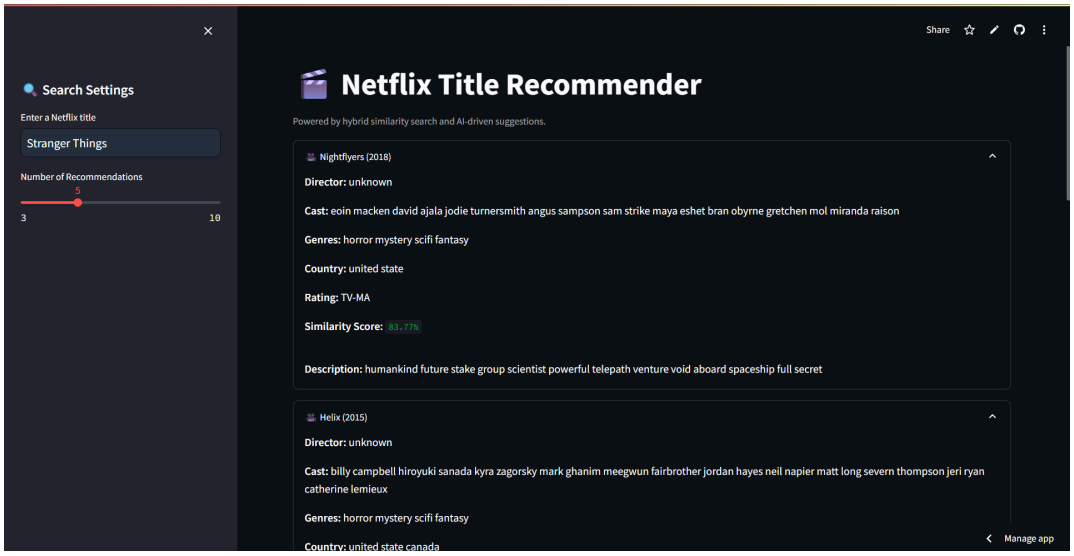

Fig. 5. Streamlit app interface showing recommendations for a known movie in the dataset

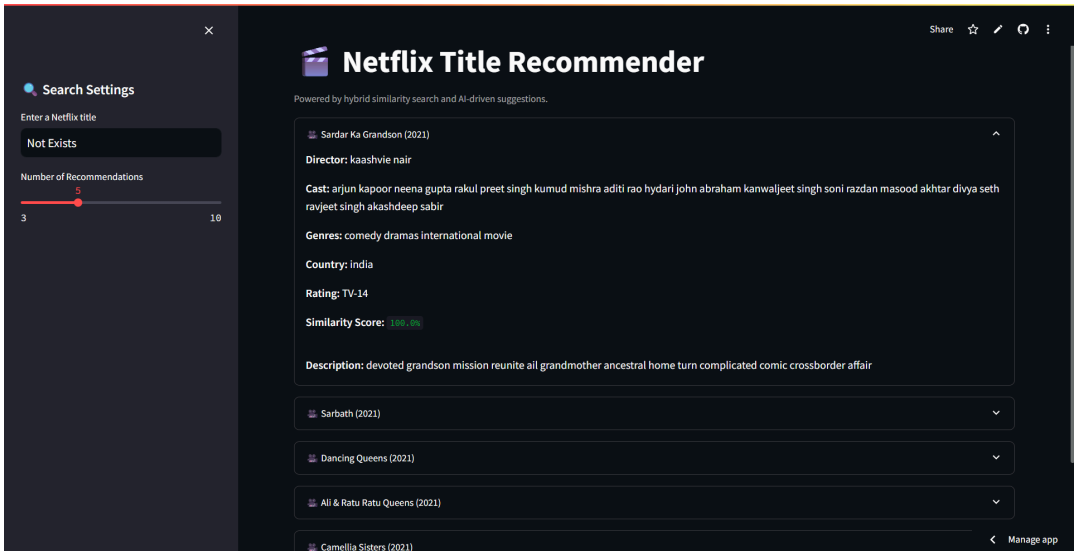

Fig. 6. Streamlit app interface showing fallback recommendations for a non-existent movie title

5. Conclusion

This work demonstrates the viability of content-based filtering as a strategy for media recommendation in the absence of user-specific data. By leveraging item metadata and natural language processing techniques, the model constructs a meaningful representation of media

items in a high-dimensional vector space. Recommendations are generated based on proximity in this space, yielding thematically coherent results.

The project also illustrates the value of modular design and modern deployment frameworks. The system is fully functional, scalable, and interpretable, with a working web interface that enhances accessibility for end users. Future enhancements may include the integration of collaborative filtering components, support for multilingual content, and deployment through containerized infrastructure to ensure reproducibility and scalability.