

Names: zahra shariati and parsae esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

Machine Learning Final Project

Abstract:

In this project, first we try to understand the Movies Dataset, clean it and deploy a model for movie recommendation system and improve it. Then we make a user interface and deploy it on Hugging Face Spaces.

Introduction:

The Movies Dataset is a collection of data on 45,000 movies containing 24 features, both numeric and categorical. As shown in figure 1, most of our columns are categorical (20 columns) and only 4 of them are numeric.

#	Column	Non-Null Count	Dtype
0	adult	45466 non-null	object
1	belongs_to_collection	4494 non-null	object
2	budget	45466 non-null	object
3	genres	45466 non-null	object
4	homepage	7782 non-null	object
5	id	45466 non-null	object
6	imdb_id	45449 non-null	object
7	original_language	45455 non-null	object
8	original_title	45466 non-null	object
9	overview	44512 non-null	object
10	popularity	45461 non-null	object
11	poster_path	45080 non-null	object
12	production_companies	45463 non-null	object
13	production_countries	45463 non-null	object
14	release_date	45379 non-null	object
15	revenue	45460 non-null	float64
16	runtime	45203 non-null	float64
17	spoken_languages	45460 non-null	object
18	status	45379 non-null	object
19	tagline	20412 non-null	object
20	title	45460 non-null	object
21	video	45460 non-null	object
22	vote_average	45460 non-null	float64
23	vote_count	45460 non-null	float64

dtypes: float64(4), object(20)

Figure 1: Info for dataset columns

After importing needed libraries and loading the dataset, we need to clean it.

EDA and Cleaning:

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

As it is shown in figure 2, there are lots of null items in most columns. It’s better to get rid of them in the first place. Most movies don’t have a homepage, tagline and poster path, so we remove these columns as they are not useful to us. Moreover, original title and original language are not important, because title is enough and original language is not what matters, the original language of a movie can be English due to the country it's made in, but it might have various versions for some other languages. Furthermore, videos are required here so we drop that column.

Also, there are 3 rows, causing some mismatches. (Figure 3) For example, their overview value is Released which is the value of the status column. And these mismatches rise from the budget column. Those rows with their budget value in NaN, should be removed.

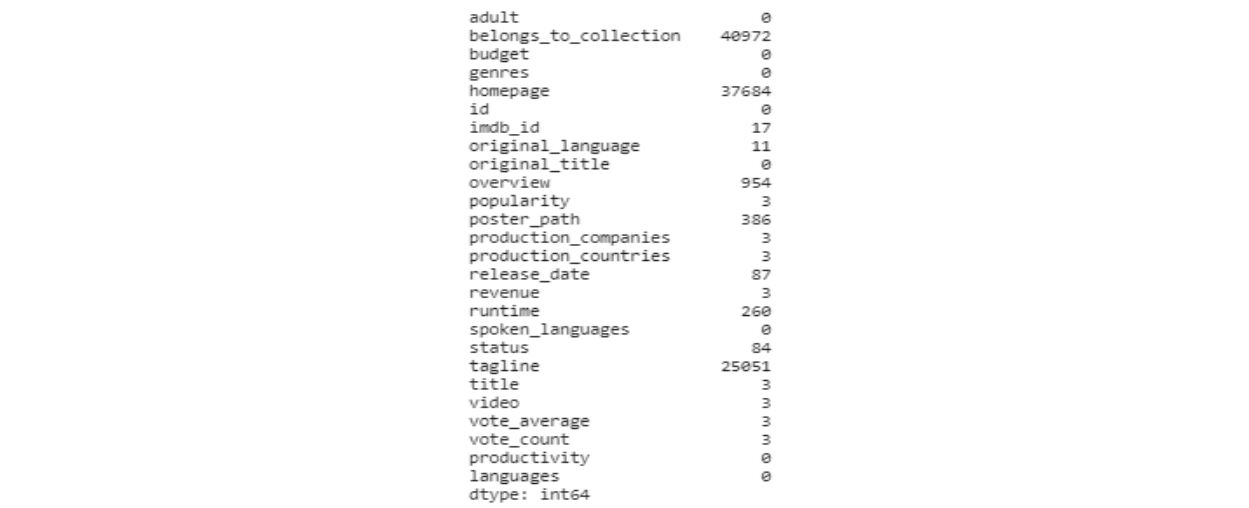


Figure 2: number of null values in each column

	adult	belongs_to_collection	budget	genres	homepage	id	imdb_id	original_language	original_title	overview
19730	- Written by Ørnås	0.065736	NaN	[[{'name': 'Carousel Productions', 'id': 11176}]]	[[{'iso_3166_1': 'CA', 'name': 'Canada', 'iso_3166_1_alpha_2': 'CA'}]]	1997-08-20	0	104.0	[[{'iso_639_1': 'en', 'name': 'English'}]]	Released
29503	Rune Balot goes to a casino connected to the ...	1.931659	NaN	[[{'name': 'Aniplex', 'id': 2883}, {'name': 'Go...'}]]	[[{'iso_3166_1': 'US', 'name': 'United States', 'iso_3166_1_alpha_2': 'US'}]]	2012-09-29	0	68.0	[[{'iso_639_1': 'ja', 'name': '日本語'}]]	Released
35587	Avalanche Sharks tells the story of a bikini ...	2.185485	NaN	[[{'name': 'Odyssey Media', 'id': 17161}, {'name': '...'}]]	[[{'iso_3166_1': 'CA', 'name': 'Canada', 'iso_3166_1_alpha_2': 'CA'}]]	2014-01-01	0	82.0	[[{'iso_639_1': 'en', 'name': 'English'}]]	Released

Figure 3: mismatches

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

After making the categorical budget feature a numeric one, now there are 5 numeric features in general. We first do the cleaning for them. (Figure 4)

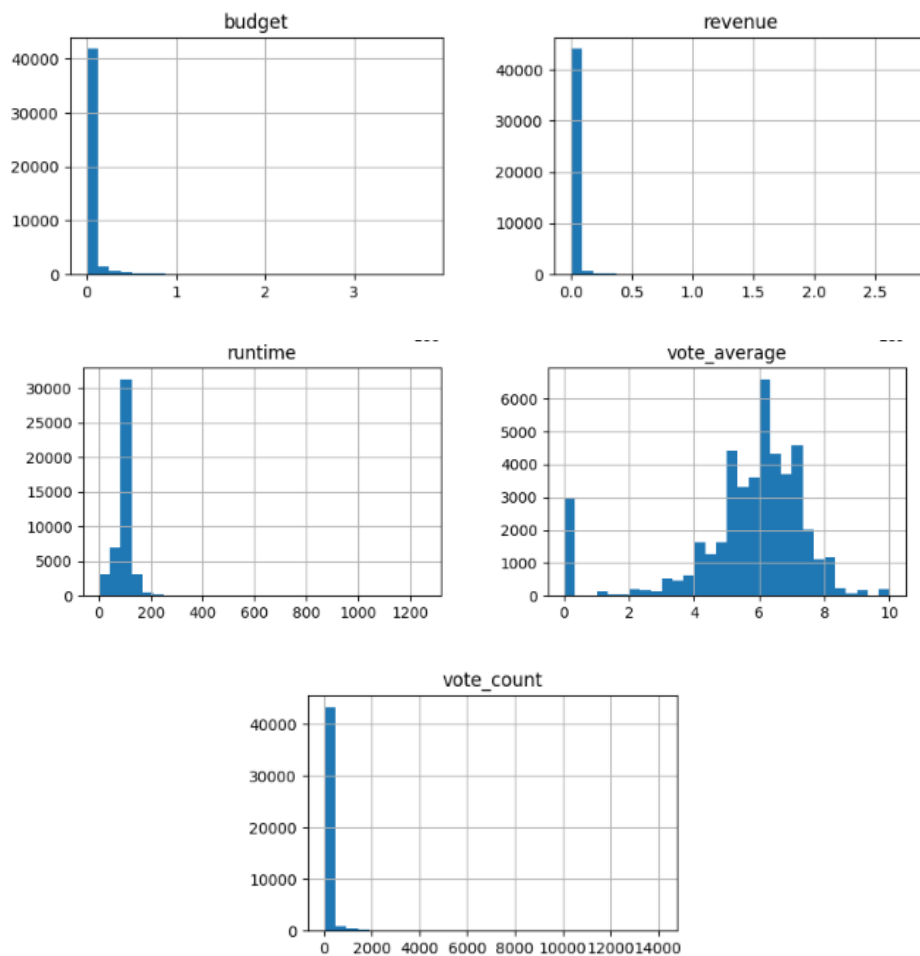


Figure 4: histogram plot of numeric features

Let's start with budget and revenue. These 2 have a majority of zero values. (Shown in figure 5) we make a new column called productivity which is the division of revenue to budget multiplied by 100 and then we remove budget and revenue columns. But as it is seen in Figure 6, production values are dense around 0, due to lots of zero values of either budget or revenue. So, we omit this column too.

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

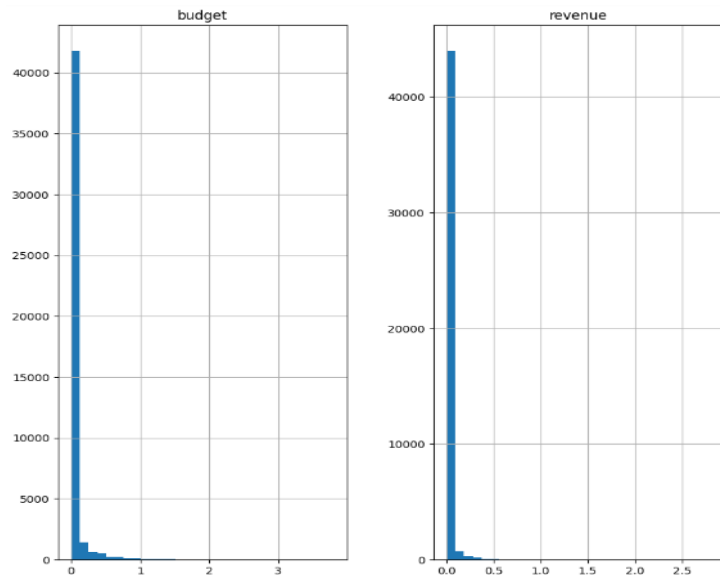


Figure 5: histogram of budget and revenue columns

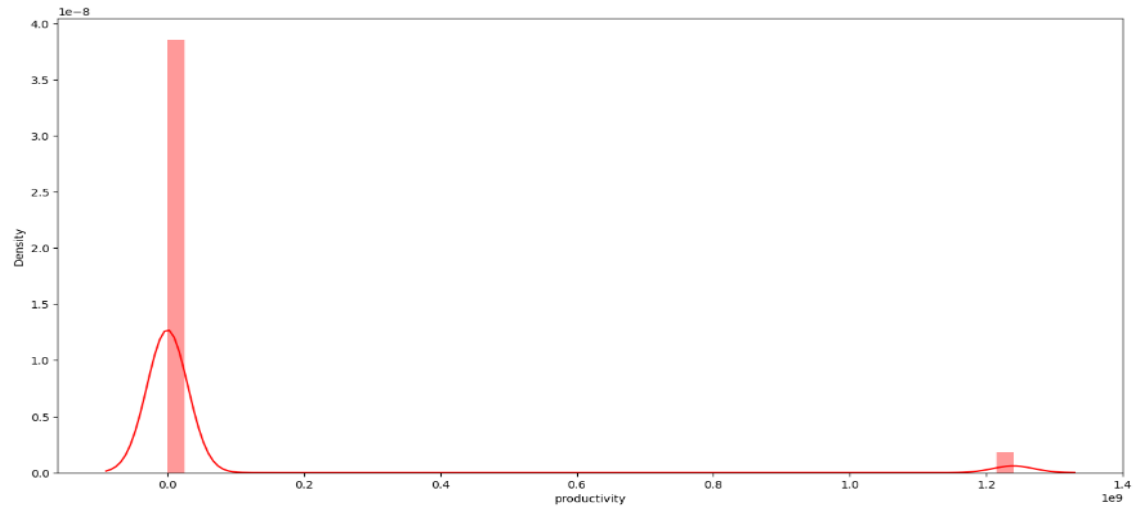


Figure 6: the density of popularity column

As in Figure 4, most movies have less than 5 rates (vote_count). So, all the votes are valuable, and there is no need to make a new feature for the relation of vote average and vote count. We omit the vote count column and continue with the vote average column.

--	--	--

Names: zahra shariati and parsae esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

For the collections, we replace the null values with none, then for those in the same collection we put a name in a new column called collection name. Then in the belong to collection column we put 1 or 0 if they belong to a collection or not.

In the status column, 99% of the movies have the ‘Released’ status. So, it doesn’t make any significant effect on the recommendation and we drop it. (Figure 7)

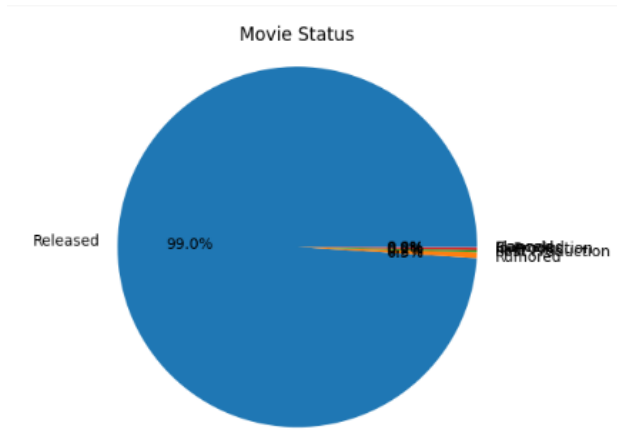


Figure 7

According to Figure 2, there are 3 rows that have no vote average and some other important features so we omit them. Also, for those movies that have null values in the runtime column, we put the mean value of this column and replaced 0 values with none. Now the number of null items is as it's shown in figure 8. Only imdb id and overview and release date columns have null values. One of the most important features is the Release date because there are lots of movies which have exact same names and the only way to clarify them for users is to put the release date alongside. So those movies that don't have this feature value, need to be removed.

adult	0
belongs_to_collection	0
genres	0
id	0
imdb_id	17
overview	954
popularity	0
production_companies	0
production_countries	0
release_date	84
runtime	0
title	0
vote_average	0
vote_count	0
productivity	0

Figure 8

--	--	--

Names: zahra shariati and parsae esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

Now for those remaining release dates, we first convert them to date time. Now, it's better to divide them into year and month, because the full date is not needed for finding similarities, but year is important when a user always watches movies from a specific decade or a specific season or month.

Those movies which don’t have an imdb id are useless to use and need to be removed because we use these feature values in the UI.

The values in the spoken languages are so confusing (Figure 9), so we convert these object type values into strings consisting of 133 unique values like: fa, en, fr , ... and put them in a new column, named languages.

```
array([[{'iso_639_1': 'en', 'name': 'English'}],
      [{"iso_639_1": 'en', 'name': 'English'}, {'iso_639_1': 'fr', 'name': 'Français'}],
      [{"iso_639_1": 'en', 'name': 'English'}, {'iso_639_1': 'es', 'name': 'Español'}],
      ...,
      [{"iso_639_1": 'sv', 'name': 'svenska'}, {'iso_639_1': 'de', 'name': 'Deutsch'}],
      [{"iso_639_1": 'ar', 'name': 'العربية'}, {'iso_639_1': 'pl', 'name': 'Polski'}],
      [{"iso_639_1": 'ff', 'name': 'Fulfulde'}, {'iso_639_1': 'en', 'name': 'English'}]],
      dtype=object)
```

Figure 9

We clean the values in genres, companies and countries columns as what we did for spoken languages. One concerning thing is that we must put (.) between each word in the values of the languages, genres, companies and countries. This helps us later in the embedding and our model. Figure 10 shows the number of null values.

adult	0
belongs_to_collection	0
genres	0
id	0
imdb_id	0
overview	939
popularity	0
production_companies	0
production_countries	0
release_date	0
runtime	0
title	0
vote_average	0
vote_count	0
productivty	0
languages	0
collection_name	0
is_adult	0
popularity_cat	0
release_year	0
release_month	0

Figure 10

Now we have a group of numeric features and a group of categorical features. We put all our categorical features in a new column named stringy features and omit all categorical features from before. We must be aware that id and imdb id are not included.

--	--	--

Names: zahra shariati and parsae esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

Embedding

We need to embed our categorical features first.

In scikit-learn, `fit_transform` is a convenience method that combines two steps: fitting a model or transformer to the data and then transforming the data using the learned model.

Fitting the model or transformer:

Before transforming the data, the model or transformer needs to be fitted to the data. This step involves learning or estimating the necessary parameters from the given data.

In the case of `CountVectorizer`, fitting the model means creating a vocabulary based on the text data. The vocabulary consists of all the unique words present in the data, and their indices are assigned.

During the fitting process, the `CountVectorizer` analyzes the text data to identify the vocabulary, including filtering out any specified stop words, applying tokenization, and determining the frequency of each word.

The `fit_transform` method performs this fitting process as the first step.

Transforming the data:

Once the model or transformer is fitted to the data, the second step is to transform the data using the learned parameters.

In the case of `CountVectorizer`, transforming the data means converting the text documents into a numerical representation based on the learned vocabulary.

The `fit_transform` method applies this transformation to the input data using the fitted `CountVectorizer` object.

The resulting transformed representation could be a sparse matrix, where each row represents a document, and each column represents a word in the vocabulary. The values in the matrix indicate the frequency or count of each word in each document.

The `toarray()` method is called after `fit_transform` to convert the sparse matrix into a dense NumPy array for easier manipulation and inspection.

By combining the fitting and transformation steps into a single method call, `fit_transform` simplifies the process and is often more efficient than calling `fit` and `transform` separately. It ensures that the same vocabulary and learned parameters are used for both fitting and transforming the data consistently.

`max_features` specifies the maximum number of features (unique words) to be included in the vocabulary. In this case, it is set to 12000, meaning only the 12000 most frequent words will be used.

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

Now that we have embedded the stringy features, we are ready to make our model and train it.

Models

We make 3 methods (for numeric variables, for embedded features, and for combining numeric and embedded).

RecommenderModel:

This module represents the main recommender model and inherits from the nn.Module base class.

It takes several input parameters: numeric_input_size, embedding_size, hidden_size, fusion_hidden_size, and num_items.

The numeric_input_size is the size of the numeric input features, embedding_size is the size of the embedded string features, hidden_size is the size of the hidden layers, fusion_hidden_size is the size of the hidden layer in the fusion section, and num_items is the number of items to be recommended.

The *init* method initializes the parameters and defines the layers for processing numeric features, fusion, and the output layer.

In the forward method, the numeric features and embedded string features are processed separately.

The numeric features are passed through the numeric layers, which consist of a linear layer and a ReLU activation function.

The numeric output is then concatenated with the embedded string features.

The fused features are passed through the fusion layers, which also consist of a linear layer and a ReLU activation function.

Finally, the fused output is passed through the output layer, which produces the recommendation output.

NumericModel:

This module is responsible for processing the numeric input features.

It takes two input parameters: input_size and hidden_size.

The input_size is the size of the input features, and hidden_size is the size of the hidden layers.

In the *init* method, the parameters are initialized, and the layers for processing numeric features are defined.

The forward method takes the numeric features as input and passes them through the defined layers, which include a linear layer and a ReLU activation function.

The output of the forward method is the result of processing the numeric features.

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

StringyModel:

This module is responsible for processing the embedded string features.

It takes two input parameters: `embedding_size` and `hidden_size`.

The `embedding_size` is the size of the embedded string features, and `hidden_size` is the size of the hidden layers.

In the `init` method, the parameters are initialized, and the layers for processing the embedded string features are defined.

The forward method takes the embedded string features as input and passes them through the defined layers, which include a linear layer and a ReLU activation function.

The output of the forward method is the result of processing the embedded string features.

After training these models we understand that the best model is the stringy model that uses embedding. So, now we must make all numerical features into strings.

To belong to the collection we convert those 0 and 1 integers into 0 and 1 in string type. For release year we convert the integers into strings, and for release month we write down the name of the months. We add an `isadult` column instead of `adult` and the values are `kid` and `adult`.

According to Figure 4 and Figure 11, the duration of most movies is between 100 and 200 minutes. So, we set `long` for those under 60 minutes and `long` for upper 60 minutes.

```
count    45362.000000
mean      97.497955
std       33.934187
min        1.000000
25%       87.000000
50%       96.000000
75%      107.000000
max      1256.000000
```

Figure 11

Now we need to deal with popularity and vote average, to decide which movies are more popular, we must choose between one of them. According to Figure 12, vote average describes better and gives more information. So, regarding the votes we set 4 values for a new column named `popularity_cat` (not popular, almost popular, very popular, most popular). Then we drop the popularity and vote average columns.

--	--	--

Names: zahra shariati and parsae esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

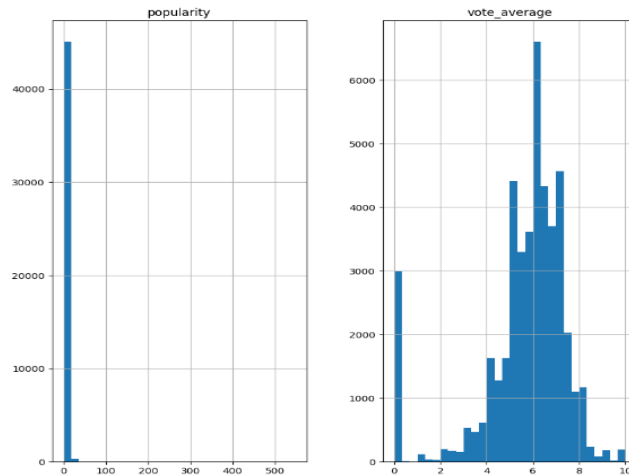


Figure 12

Now that all our features are strings, we embed them all and separate the value of each feature with a (.).

We must consider the fact that there is no way to say which recommendation is the best because it varies from person to person. We can just improve our models to fit better to the users' expectations.

UI

Movie Recommender System

Select 3 movies from dropdown

Choose an option

Recommend Me

We used tensor and pickle to load and open our needed data. Streamlit is used for designing the user interface. And we made a function called make_recommendation that gets the index of the row of movies, the model and the movies that the user selected from the select box and the number of movies to recommend.

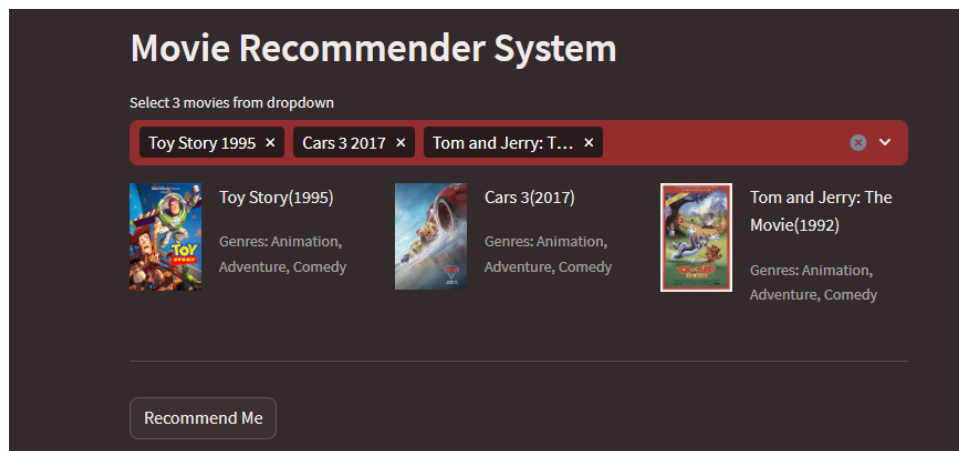
We used OMDB API to load the posters and all the information shown in the user interface.

--	--	--

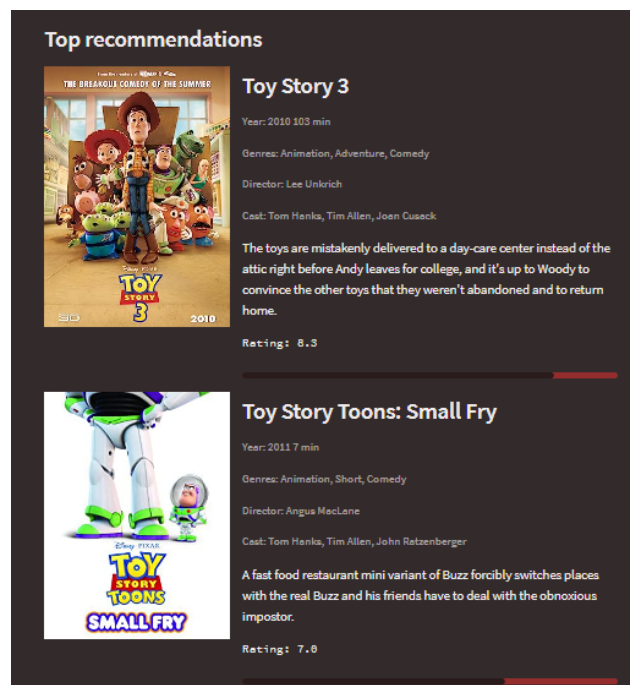
Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project

Users can select up to 3 movies and get 15 recommendations.

Then we deployed our model in Hugging face.




User selected 3 items (their title and release year and genres are shown below the select box)



--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project



Aloha, Scooby-Doo!

Year: 2005 74 min


Genres: Animation, Adventure, Comedy, Family, Mystery

Director: Tim Maltby

Cast: Frank Welker, Casey Kasem, Mindy Cohn, Grey Griffin

The Mystery Gang goes to Hawaii for the Big Kahuna of Hanahuna Surfing Contest. However, the gang and the locals find the island invaded by the vengeful Wiki Tiki spirit and his demons.

Rating: 6.4



Toy Story 2

Year: 1999 92 min


Genres: Animation, Adventure, Comedy

Director: John Lasseter, Ash Brannon, Lee Unkrich

Cast: Tom Hanks, Tim Allen, Joan Cusack

When Woody is stolen by a toy collector, Buzz and his friends set out on a rescue mission to save Woody before he becomes a museum toy property with his roundup gang Jessie, Prospector, and Bullseye.

Rating: 7.9



Petunia

Year: 2012 112 min


Genres: Comedy, Drama, Romance

Director: Ash Christian

Cast: Tobias Segal, Thore Birch, Christine Lehti

The story of a family whose growth is stunted... a family that learns how to love themselves while loving each other (a little too much).

Rating: 5.1



En Büyük Saban

Year: 1984 93 min

Genres: Comedy, Drama, Romance

Director: Kartal Tibet


Cast: Kemal Sunal, Nilgün Bubikoğlu, Kamran Usluer

A good man tries to help a blind girl to see again but he doesn't have enough money for the operation.

Rating: 6.7

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project



Lava

Year: 2014 7 min

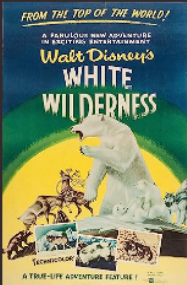
Genres: Animation, Short, Family

Director: James Ford Murphy

Cast: Napua Greig, Kuana Torres Kahele, Muneer Lyeti

A story that takes place over millions of years and is inspired by the beauty of tropical islands and the allure of ocean volcanoes.

Rating: 7.2



White Wilderness

Year: 1958 72 min


Genres: Documentary, Family

Director: James Alger

Cast: Winston Hibler, Volmer Sorensen

The wildlife of the arctic is explored in this true-life adventure.

Rating: 5.3



Secret of the Wings

Year: 2012 75 min


Genres: Animation, Family, Fantasy

Director: Boba Gennaway, Peggy Holmes

Cast: Mae Whitman, Lucy Hale, Timothy Dalton

Tinkerbell wanders into the forbidden Winter woods and meets Periwinkle. Together they learn the secret of their wings and try to unite the warm fairies and the winter fairies to help Pixie Hollow.

Rating: 7.0



Mars Needs Moms

Year: 2011 88 min

Genres: Animation, Action, Adventure

Director: Simon Wells


Cast: Seth Green, Joan Cusack, Dan Fogler

A young boy named Milo gains a deeper appreciation for his mom after Martians come to Earth to take her away.

Rating: 5.4

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project



Cars 2

Year: 2011 106 min

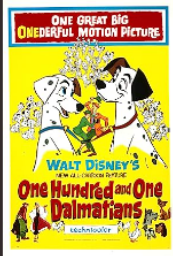
Genres: Animation, Adventure, Comedy

Director: John Lasseter, Bradford Lewis

Cast: Owen Wilson, Larry the Cable Guy, Michael Caine

Star race car Lightning McQueen and his pal Mater head overseas to compete in the World Grand Prix race. But the road to the championship becomes rocky as Mater gets caught up in an intriguing adventure of his own: international espionage.

Rating: 6.2



One Hundred and One Dalmatians

Year: 1961 79 min

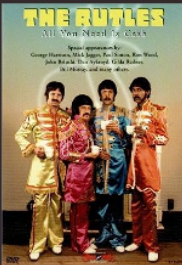
Genres: Animation, Adventure, Comedy

Director: Clyde Geronimi, Hamilton Luske, Wolfgang Reitherman

Cast: Rod Taylor, Betty Lou Gerson, J. Pat O'Malley

When a litter of Dalmatian puppies are abducted by the minions of Cruella De Vil, the owners must find them before she uses them for a diabolical fashion statement.

Rating: 7.3



The Rutles: All You Need Is Cash

Year: 1978 76 min


Genres: Comedy, Music

Director: Eric Idle, Gary Weiss

Cast: Eric Idle, John Halsey, Ricky Fatso

Charts the adventures of the prefab four, possibly the most famous band of all time.

Rating: 7.3



Ciao Maschio

Year: 1978 113 min

Genres: Comedy, Drama, Fantasy

Director: Marco Ferreri

Cast: Gérard Depardieu, Marcello Mastroianni, James Coco

A man walking on the beach near New York City finds the corpse of King Kong. He also finds Kong's orphaned son, and takes it to a friend who lives in the city, and they decide to raise it.

Rating: 6.4

--	--	--

Names: zahra shariati and parsa esmaeili		Student Numbers : 98222056, 98222004
Course: Machine Learning		Final Project



THE NAKED MAN
Michael Rapaport
From Ethan Coen, Co-creator of Fargo
Chiropractor by day. Wrestler by night.

The Naked Man

Year: 1999 98 min

Genres: Comedy, Drama, Sport

Director: J. Todd Anderson

Cast: Michael Rapaport, Michael Jeter, John Carroll Lynch

A man takes matters into his own hands when a pharmaceutical kingpin moves into his town to cause some real trouble.

Rating: 5.1

Top 15 recommendation

--	--	--