Report

On

Content Based Recommendation System

Prepared by

Darshit Patel (18012011057)

Guided By

Prof. Ketan J. Sarvakar

B.Tech Semester VII

Nov-Dec -2021

Subject: Deep Learning

Subject code: 2CEIT78PE1

Department of Computer Engineering
U. V. Patel College of Engineering
Ganpat University, Ganpat Vidyanagar – 384012

Abstract:

In this paper we study Content-Based Recommendation Systems. This definition refers to systems used in the Web in order to recommend an item to a user based upon a description of the item and a profile of the user's interests. To start with, we will give a definition of a Recommendation system in generally.Then, we will discuss why recommendation systems are necessary for Web users nowadays and pinpoint the problem that are trying to solve. Furthermore, we will focus on techniques used in content-based recommendation systems in order to create a model of the user's interests and analyze an item collection, using the representation of the items. Additionally, we will emphasize on the advantages and drawbacks of recommendation systems, both in the context of making recommendations and in contrast with other types of recommendation systems. Finally, we will discuss the LIBRA content-based recommendation system and we will emphasize on the CBMRS, PRES and COBRA systems, which are implemented using the Java Platform.

Notebook used:

Google Colab - Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.

Tools and Libraries

Tools and Libraries	Usage
Keras	Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.
seaborn	is an open-source Python library built on top of matplotlib. It is used for data visualization and exploratory data analysis. Seaborn works easily with dataframes and the Pandas library. The graphs created can also be customized easily
Numpy	We are using it for the Image matrix handling
Logistic Regression	Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes.

Logistic Regression is a supervised learning that computes the probabilities for classification problems with two outcomes. It can also be extended to predict several classes. In Logistic Regression model, we apply the sigmoid function, which is

$$\sigma(\mathrm{z}) = \frac{\&}{\& \, ^{-} \, e|)^{*}}.$$

This function successfully maps any number into the value between 0 and 1 and we can regard this value as the probability of predicting classes. For example, we have two classes and they are presence of heart disease and absence of disease. If we set the threshold as 0.5, applying the sigmoid function gives us a value of 0.7, which means the man has the 70% probability of having heart disease so we will predict that he has heart disease.

I used the L2 penalty, the square of the magnitude ofcoefficients, supported by Logistic Regression to avoidoverfitting. The train accuracy is 83.88% and test accuracy is 85.25%. It performs well but not the best for us. The advantageof the Logistic Regression is that it does not need too much computational resources and it is highly interpretable. So it is easy and sufficient to apply Logistic Regression. However, thelimitation of Logistic Regression is that it assumes linearity between the features of the dataset. In the real world, the data israrely separable, neither as our dataset. That is why we cannot reach a very high accuracy of 90 %.

Work Flow:

Step 1: Import Requires Libraries

Step 2: Get the data

Step 3: Split the data into train and test

Step 4: Split features from labels Train and test

Step 5: Normalize the data

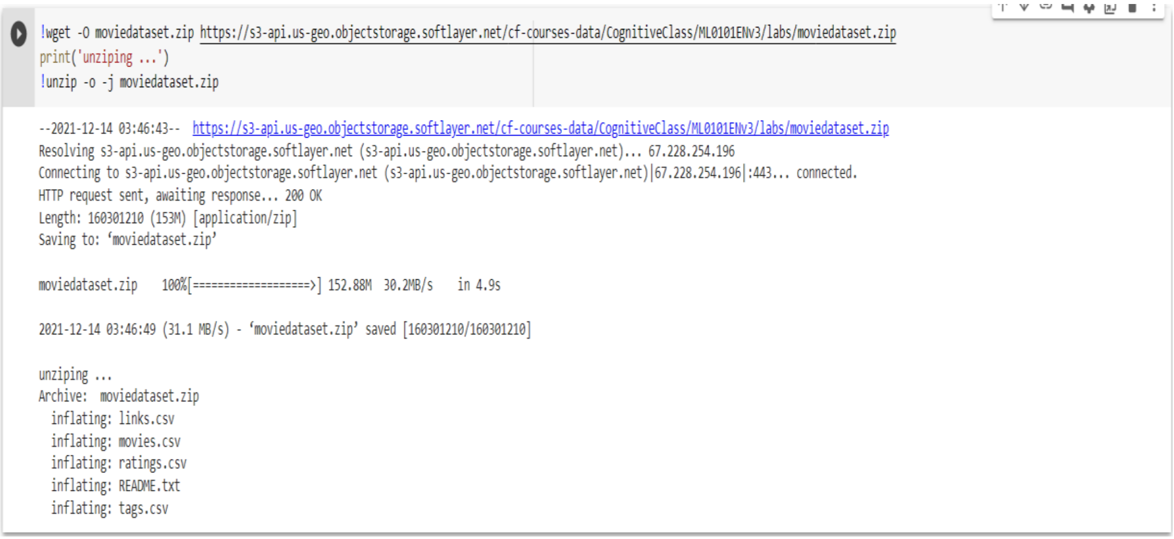
Step 6: Build the model

Step 7: Inspect the model

Step 8: Train the model for 300 epochs

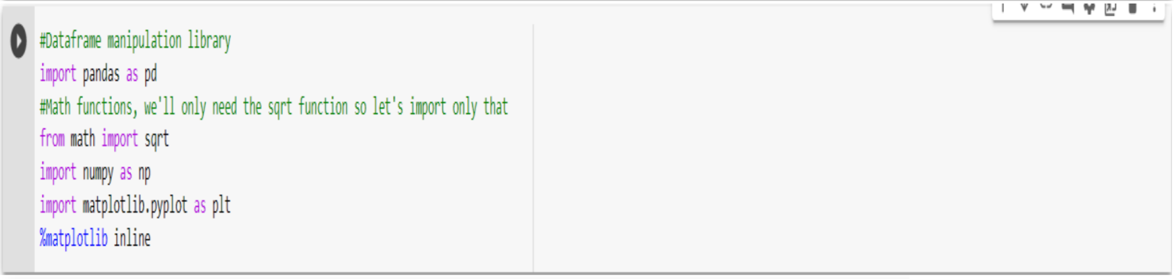
Step 9: Result Prediction

Output:

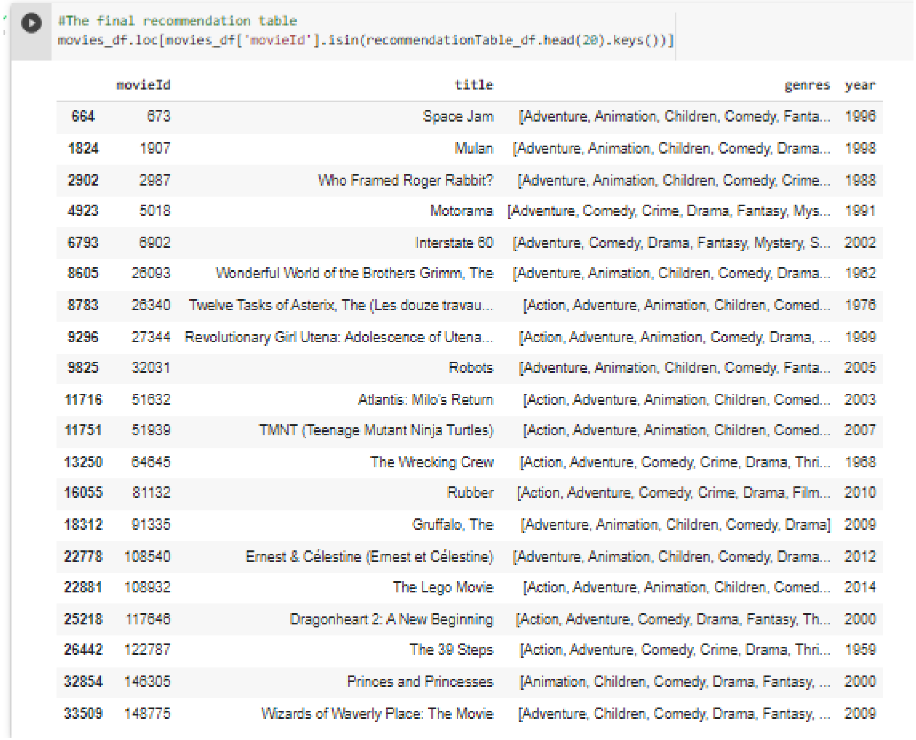


```
!wget -O moviedataset.zip https://s3-api.us-geo.objectstorage.softlayer.net/cf-courses-data/CognitiveClass/ML0101ENv3/labs/moviedataset.zip
print('unziping ...')
!unzip -o -j moviedataset.zip
```

```
--2021-12-14 03:46:43--  https://s3-api.us-geo.objectstorage.softlayer.net/cf-courses-data/CognitiveClass/ML0101ENv3/labs/moviedataset.zip
Resolving s3-api.us-geo.objectstorage.softlayer.net (s3-api.us-geo.objectstorage.softlayer.net)... 67.228.254.196
Connecting to s3-api.us-geo.objectstorage.softlayer.net (s3-api.us-geo.objectstorage.softlayer.net)|67.228.254.196|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 160301210 (153M) [application/zip]
Saving to: ‘moviedataset.zip’

moviedataset.zip    100%[===================>] 152.88M  30.2MB/s    in 4.9s

2021-12-14 03:46:49 (31.1 MB/s) - ‘moviedataset.zip’ saved [160301210/160301210]

unziping ...
Archive:  moviedataset.zip
  inflating: links.csv
  inflating: movies.csv
  inflating: ratings.csv
  inflating: README.txt
  inflating: tags.csv
```



```
#Dataframe manipulation library
import pandas as pd
#Math functions, we'll only need the sqrt function so let's import only that
from math import sqrt
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```



```
#The final recommendation table
movies_df.loc[movies_df['movieId'].isin(recommendationTable_df.head(20).keys())]
```

	movieId	title	genres	year
664	673	Space Jam	[Adventure, Animation, Children, Comedy, Fanta...	1996
1824	1907	Mulan	[Adventure, Animation, Children, Comedy, Drama...	1998
2902	2987	Who Framed Roger Rabbit?	[Adventure, Animation, Children, Comedy, Crime...	1988
4923	5018	Motorama	[Adventure, Comedy, Crime, Drama, Fantasy, Mys...	1991
6793	6902	Interstate 60	[Adventure, Comedy, Drama, Fantasy, Mystery, S...	2002
8605	26093	Wonderful World of the Brothers Grimm, The	[Adventure, Animation, Children, Comedy, Drama...	1962
8783	26340	Twelve Tasks of Asterix, The (Les douze travau...	[Action, Adventure, Animation, Children, Comed...	1976
9296	27344	Revolutionary Girl Utena: Adolescence of Utena...	[Action, Adventure, Animation, Comedy, Drama, ...	1999
9825	32031	Robots	[Adventure, Animation, Children, Comedy, Fanta...	2005
11716	51632	Atlantis: Milo's Return	[Action, Adventure, Animation, Children, Comed...	2003
11751	51939	TMNT (Teenage Mutant Ninja Turtles)	[Action, Adventure, Animation, Children, Comed...	2007
13250	64645	The Wrecking Crew	[Action, Adventure, Comedy, Crime, Drama, Thri...	1968
16055	81132	Rubber	[Action, Adventure, Comedy, Crime, Drama, Film...	2010
18312	91335	Gruffalo, The	[Adventure, Animation, Children, Comedy, Drama]	2009
22778	108540	Ernest & Célestine (Ernest et Célestine)	[Adventure, Animation, Children, Comedy, Drama...	2012
22881	108932	The Lego Movie	[Action, Adventure, Animation, Children, Comed...	2014
25218	117646	Dragonheart 2: A New Beginning	[Action, Adventure, Comedy, Drama, Fantasy, Th...	2000
26442	122787	The 39 Steps	[Action, Adventure, Comedy, Crime, Drama, Thri...	1959
32854	146305	Princes and Princesses	[Animation, Children, Comedy, Drama, Fantasy, ...	2000
33509	148775	Wizards of Waverly Place: The Movie	[Adventure, Children, Comedy, Drama, Fantasy, ...	2009

Dataset Name:

movies.csv

Dataset location:

https://drive.google.com/file/d/1vyyXiYDjGKz9e3dQRMtD_r5ZIsxhVcKx/view?usp=sharing

GitHub Code link:

https://github.com/dp3400/Content-based-recommendation-system