Bilkent University

Department of Computer Science

CS 464 Introduction to Machine Learning

Movie Recommendation System

Final Report

Group 12

Alperen Alkan

Batu Arda Düzgün

Ece İzmir

Elif Özer

Kadir Can Kasan

Instructor: Ercüment Çiçek

Teaching Assistant: Sina Barazandeh

1. Introduction

Our project is a movie recommendation system, developed by using machine learning algorithms such as K-means, Self-Organizing Map (SOM) and frameworks like Multi Armed Bandit (MAB). The aim is to recommend the most suitable movies to the users by doing reinforcement learning. We used the Kaggle dataset [1] as our data which consists of 27278 movies, 138493 users and approximately 20 million ratings. We filtered out the movies watched by less than 500 users and the users who watched less than 400 movies to use high-contribution movies and users in the model we created.

In the clustering part, we applied different methods to compare results and adopt the most successful one. Firstly, we used the K-means algorithm and created randomly centralized clusters and updated them by minimizing the Euclidean distance of center and data points. This method worked successfully and gave good results which is discussed in the clustering part. Secondly, we applied the K-pod method which assigns NaN values to the mean value of features. This resulted in the majority of movies going into the same cluster. Thirdly, we used SOM to reduce dimensions and easily observe the similarities in the data. We obtained valid results discussed in the report.

Then, we dealt with the reinforcement learning problem MAB to develop our system. To solve MAB, we used several algorithms which are upper confidence bounds (UCB), Thompson Sampling (TS), and Multinomial Thompson Sampling (MTS). Meaningful results are achieved by dynamically updating user preferences and learning the best movies for a user.

When we got satisfied with our results, we wanted to create a website to display our project. We created a web application by using fast api python. The web application provides two choices: choosing a film with clustering or MAB. By using this interface it is possible to see the performance metrics of our model.

2. Problem Description

Today, movie streaming services are in high demand since they ease the process of consuming thousands of movies by creating consumer satisfaction. These services use machine learning algorithms to suggest their users the most optimal contents by processing millions of data and using appropriate methods to group it and filter the unnecessary elements. Therefore, recommendation systems play a key role in this competitive market as they steer the user behavior and catalyze consumption rate. Our project is designed as a movie recommendation system that meets the likings of each user and operates correspondingly. We used a dataset where there are 10 levels of rating from 0.5 to 5. We clustered the movies using unsupervised learning algorithms. The clusters were formed by only using the user given scores. This approach is named collaborative filtering. By processing this data, we aimed to give the most optimal movie option for a user.

3. Methods

3.1. Preprocessing the data

The initial data has a huge size, so we could not read the data properly. We implemented a filtering method, to reduce the huge data into smaller data that we can work on. Thus we filtered the data such that the movies which were watched by at least 500 users; and users who watched more than 400 movies were removed. Then, we achieved a data with 2826 data points in a 10638 dimensional space. Each data is a movie and each of the 10638 features are a score given by a user.

3.2. Clustering

3.2.1. K-means clustering algorithm

K-means algorithm assigns K randomly initialized cluster centers and iteratively moves the cluster centers according to the objective function. The objective function we chose was minimizing the Euclidean distance to the cluster center for all points. To solve the data sparsity problem we assigned the mean rating each user gave to the NaN values of that user. We did not give the mean rating of the movies to their NaN values because that would cause the clustering process to be dominated by the mean rating each movie got due to around 80% of the data being NaN. This created a good result as we will discuss in the following parts.

To tune the hyper-parameter, the number of clusters (K), we applied the elbow plot method. The distortion metric -which is the sum of squared error- is plotted with respect to K, which ranges from 2 to 24. Distortion is a metric that decreases with the increasing number of clusters, thus it is meaningful to choose the point where there is a significant drop in the decreasing speed of the distortion value. In our plot, we observed that after K=9, increasing K value leads to much more little gain in terms of a lower distortion value. Thus we chose 9 as the number of clusters we are going to separate the movies into.

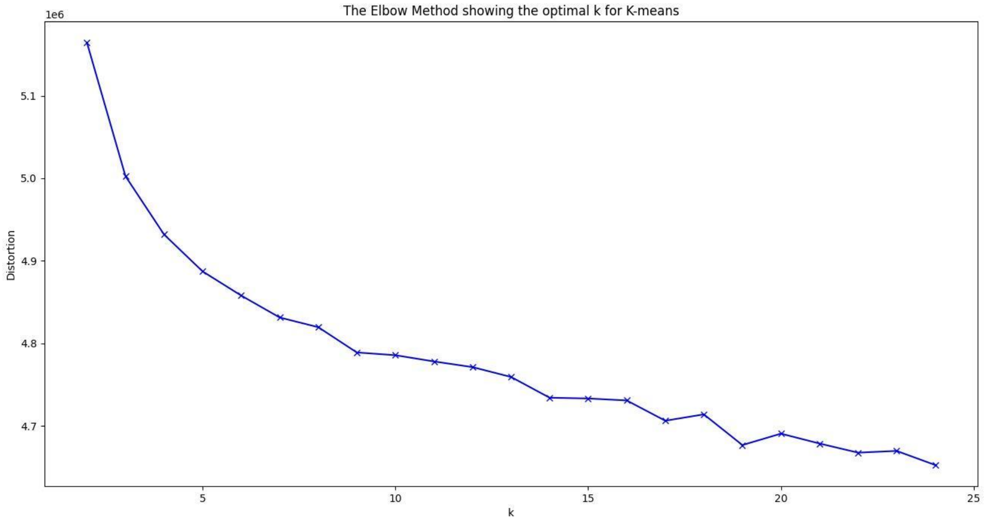


Figure 1: K-means Elbow Method Graph

3.2.2. Self-Organizing Map (SOM) method for clustering

We used SOM neural network for unsupervised learning in order to get low dimensional data. SOM can be thought as a method for dimensionality reduction. The benefit of SOM is to preserve the topological properties of input. Also, it does not use back propagation as we do not have labels for movies. After SOM, we cluster our data into 9 clusters. We represented tags of the movies in the clusters as a heat map to show the result of the SOM is meaningful. The result of the SOM will be explained in later sections.

3.2.3. K-Pod method for clustering:

K-Pod is a clustering algorithm which is proposed by Chi et. al and employs a majorization-minimization (MM) algorithm [2]. It assigns the mean values of features to NaN values and it iteratively updates these values based on the clusters they are assigned to, until the values converge.

3.3. Overview of MAB problem:

The MAB problem has an extensive history and is a classic reinforcement learning problem. The old mathematical problem says: A gambler goes to a casino with a lot of slot machines which give out their rewards according to an unknown reward distribution. What should the gambler do to maximize his profit? At the heart of this problem is the exploration and exploitation trade-off dilemma and it is extremely important as it can be fit into a lot of modern settings such as the one we will be dealing with in this project. There are many algorithms to solve this problem but we will be using upper confidence bounds (UCB), Thompson Sampling (TS), and Multinomial Thompson Sampling (MTS) as they are one of the most fundamental and flexible ones.

3.3.1. Upper Confidence Bound (UCB) Algorithm:

UCB uses the upper bounds of the arms sample means to make its decision and because of this, it is named upper confidence bound. The learner computes the upper confidence bounds of the arms according to *Figure 2* and always plays the arm with the highest one. In this step our recommendation system recommends a movie from the chosen arms cluster. It is called an optimistic algorithm because it always picks the arm with the highest potential mean, so it is optimistic. This algorithm achieves $O\left(\left(T\ ln(T)\right)^{0.5}\right)$ [3] regret. This will be used to choose which cluster of movies is the one with the highest sample mean for the new user.

$$\mu_i + \sqrt{\frac{2ln(n)}{n_i}}$$

Figure 2: The Representation of the Reward UCB of All Arms

3.3.2. Thompson Sampling (TS) Algorithm:

Thompson Sampling Algorithm is only for Bernoulli rewards, we are only showing it to build up to Multinomial Thompson Sampling. It assumes each arm's mean reward is a random variable, let call this p. It further assumes each arms p comes from a beta distribution and it initializes each arms beta distribution with $\beta = \alpha = 1$. Each round it samples the beta distribution of each arm and picks the arm with the highest value. It then samples the reward of that arm. Again in this step our recommendation system recommends a movie from the chosen arms cluster. Then calling the reward of this sample r it updates that arms β, α parameters according to *Figure 3.*

$$(\alpha_i, \beta_i) \leftarrow \begin{cases} (\alpha_i, \beta_i)\ if\ x_t \neq i \\ (\alpha_i + r, \beta_i + 1 - r)\ if\ x_t = i \end{cases}$$

Figure 3: β, α values update for Thompson Sampling

This method causes arms with a high success rate to be sampled more and more often, until the algorithm converges to the best arm. This algorithm achieves O(ln(T)) regret.

3.3.3. Multinomial Thompson Sampling(MTS) Algorithm

Thompson Sampling Algorithm is only for multinomial rewards, so it is ideal for a rating system where the users could give different 5 star ratings. Again it assumes each arm's mean reward is a random variable, let call this p. It further assumes each arms p comes from a Dirichlet distribution and it initializes each arms Dirichlet distribution with all α's = 1. Each round it samples the Dirichlet distribution of each arm. We multiply each ratings sampled probability with the rating scores and sum them up then pick the arm with the highest value. It then samples the reward of that arm. Again in this step our recommendation system recommends a movie from the chosen arms cluster and the user gives a star rating. Then calling the reward of this sample r it updates that arms α parameters according to *Figure 4.*

$$(\alpha_{i1}, \alpha_{i2} \dots\ \alpha_{iM}) \leftarrow \begin{cases} (\alpha_{i1}, \alpha_{i2} \dots\ \alpha_{iM})\ if\ x_t \neq i \\ (\alpha_{i1} + r_{1t}, \alpha_{i2} + r_{2t} \dots\ \alpha_{iM} + r_{Mt})\ if\ x_t = i \end{cases}$$

Figure 4: Update α values using Multinomial Thompson Sampling [4]

This method again causes arms with a high average star rating to be sampled more and more often, until the algorithm converges to the best arm. Dirichlet distribution is also named multivariate beta distribution, so this is very close to the TS algorithm.

4. Results

4.1. Clustering results

There are 3 methods of assessing the performance of the clustering algorithms. The heat maps of tag distribution between clusters, looking at the top rated movies in each cluster to see the cohesion between them, and a survey type application which asked people to compare the accuracy of different clustering approaches.

The heat maps are drawn by first dividing the number of movies with a specific tag in a given cluster by the number of movies in that cluster for all cluster and tag pairs. Then we sum these results for each tag and divide each of the results with its tag's sum. The second step is needed because almost all the movies have the romance tag. Because of this, if we just drew the heat map of the first steps results, it would be fully dominated by the values on the romance tag's column.

Even though clustering just based on tags would be not good, a successful cluster should still not have a homogeneous number of tags in each cluster. This Heat map aims to show that the tag distribution is not homogeneous.

The other method used as a performance metric was looking at the most popular movies in each cluster and seeing if they have some coherence. For this we sum up the total rating of each movie then pick the movies with the highest sum. This is used instead of just picking the movies with the highest mean rating because this metric finds movies which are popular and well received. Just looking at the mean rating could easily pick an obscure movie liked by a very small and specific group of people, as a good recommendation.

Lastly to have more numeric discrete performance metric a game created where users can pick a movie to see recommendations from different clusters was made. To compare the collaborative filtering methods with a more traditional approach we created other clusters using the user given tags with the k-means algorithm.

4.1.1. K-means Result

As can be seen from figure below, movie genres in a cluster are not homogeneous. Thus, each cluster is specified in a different way for recommendation.

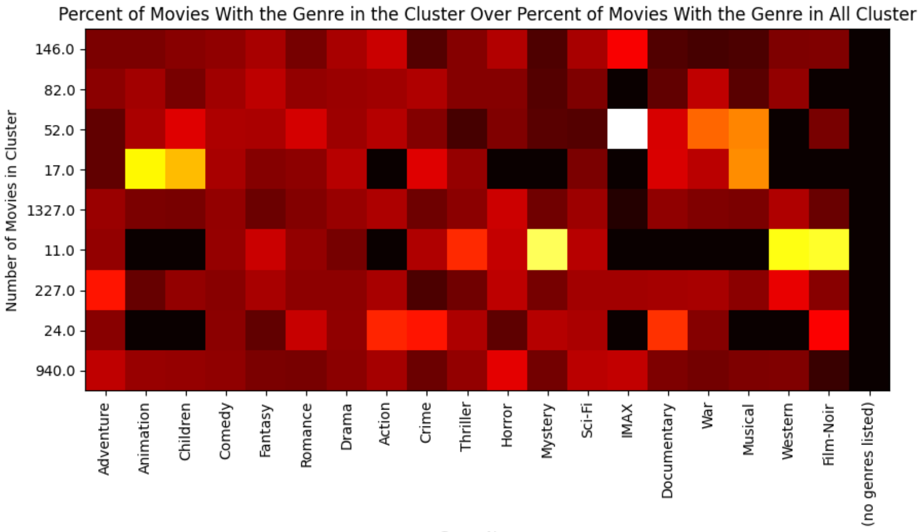


Figure 5: Heat Map of k-Means clustering

If movies from APPENDIX B are analyzed we would see: The blockbuster and critically acclaimed movies are not grouped into a single cluster. While Lord of the Rings movies are in Cluster 3 along with the likes of Schindler’s List and Forrest Gump other top movies such as Taxi Driver, Godfather: Part II, and Goodfellas are in Cluster 7.

The clusters are not restricted into genres. American Pie 2 -a comedy movie- is in the same cluster with Die Hard 2 -an action movie-; Psycho -a horror movie is in the same cluster with Taxi Driver -a drama movie.

The clusters are not restricted to sequels. Die Hard is in Cluster 3, while Die Hard 2 is in Cluster 8 and Die Hard: With a Vengeance is in cluster 4. This is very meaningful as these movies have different tones from each other.

4.1.2. Self-Organizing Map Result

As can be seen from figure below, movie genres in a cluster are not homogeneous. Thus, each cluster is specified in a different way for recommendation.

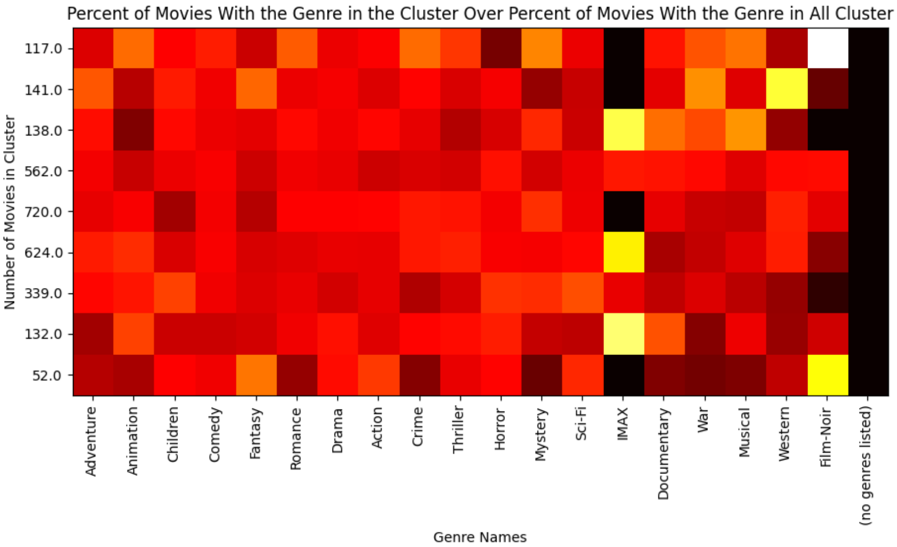


Figure 6: Heat Map of SOM Clustering

Also, the top 12 movies of clusters in *Figure 6* can be seen in APPENDIX C. In the APPENDIX C, it can be observed that we accessed meaningful clusters so that we can use them in the movie recommendation system.

For instance, Cluster 0 has well-known and popular movies. In Cluster 1, crime movies are dominant and in Cluster 2, adventure movies are dominant. Cluster 3 has Harry Potter movies and also DC and Walt Disney movies appealing to the similar segments and tastes. Cluster 5 and Cluster 6 has the movies that have been released in 90s.

4.1.3. K-Pod Result

The k-Pod algorithm led to “bad” results were approximately 97% of the movies were assigned to a single cluster, which can be observed in the *Figure 7*. This may be reasoned to our dataset being particularly sparse, in which the null values are not the minority. Because of this the algorithm might be simply iterating them closer and closer in its updates.

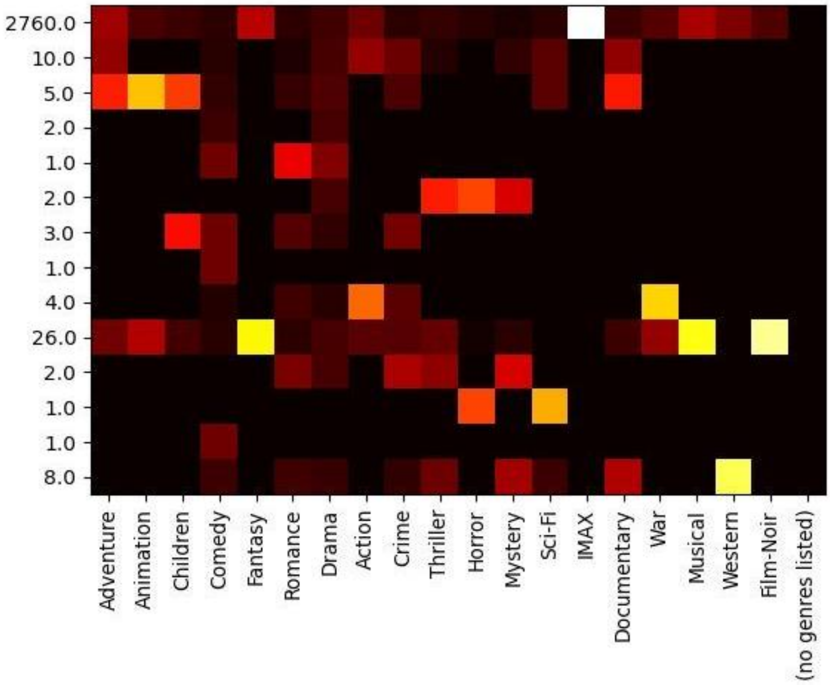


Figure 7: Heat Map of K-Pod Clustering

4.2. Multi-armed Banding Result

Regret is defined as the reward collected by an oracle algorithm minus the reward by the algorithm which is being tested. Oracle algorithm is an algorithm which knows the mean reward of all the arms and plays the optimal arm from the start.

The regret plots are created by generated data in a simulation. User preference to different clusters is governed by a multinomial distribution with randomly set values. The algorithms try to find the best sequence of recommendations for these values in a simulation. The average of multiple trials are done to minimize the effect of noise on the plots due to the stochastic nature of the algorithms.

These simulations are done to test and compare the effectiveness of different algorithms.

4.2.1. Upper Confidence Bound (UCB) Algorithm Result

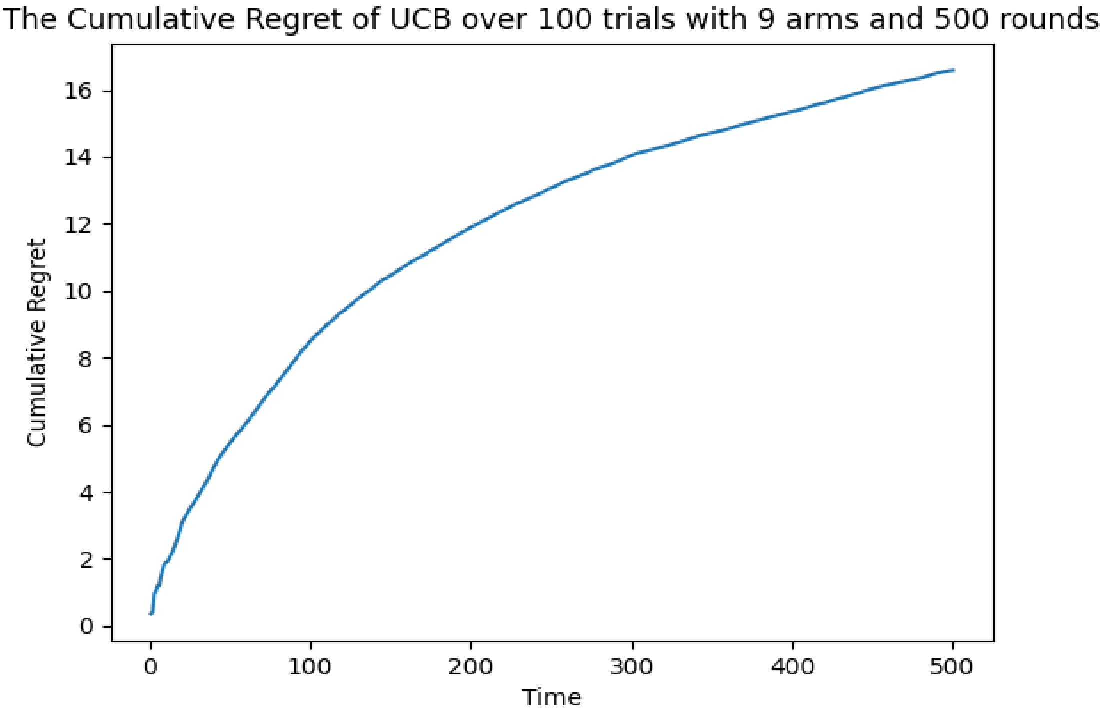


Figure 8: Cumulative Regret Graph for UCB

4.2.2. Thompson Sampling (TS) Algorithm Result

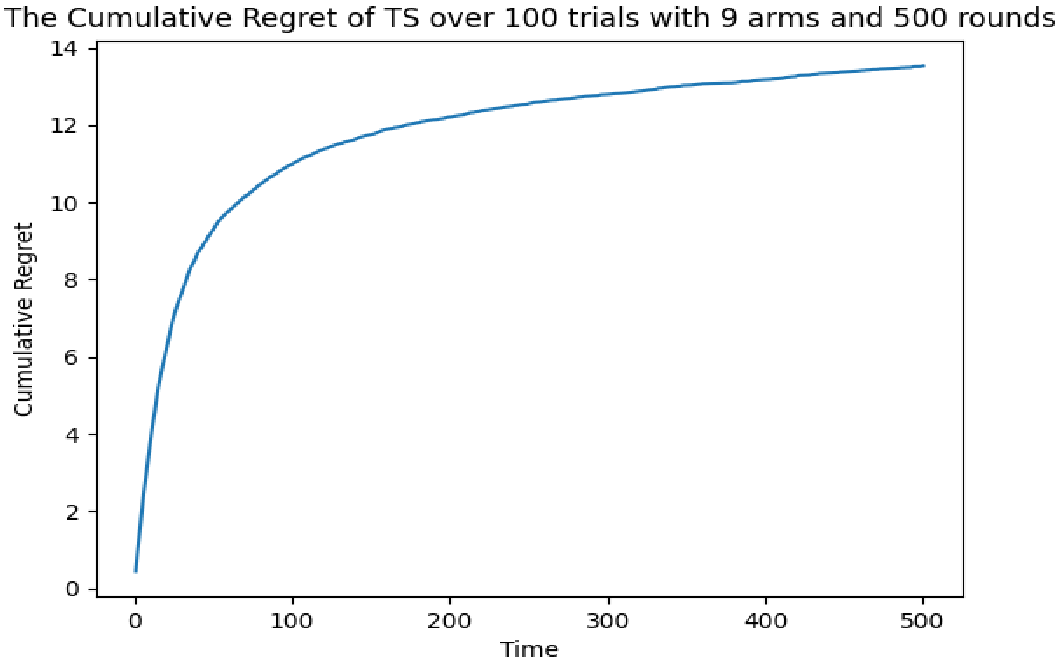


Figure 9: Cumulative Regret Graph for TS

4.2.3. Multinomial Thompson Sampling(MTS) Algorithm

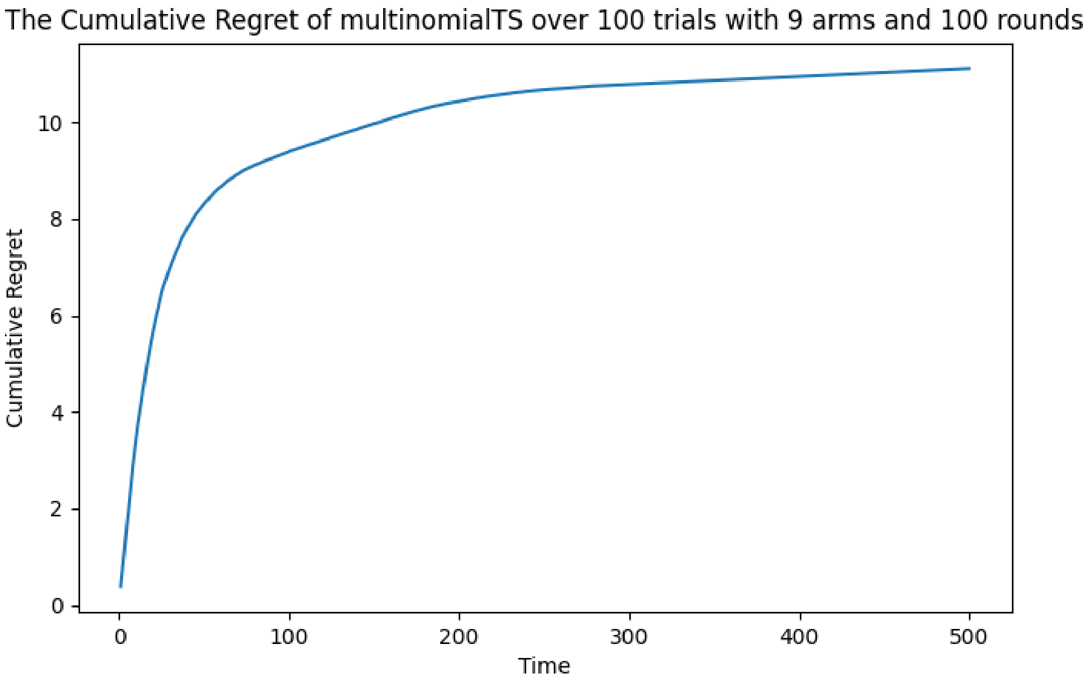


Figure 10: Cumulative Regret Graph for MTS

4.3. Movie Recommendation System

After implementing the algorithms and having the training data, we wanted it to show the performing metrics of our models. Therefore, we created a web application for this purpose. We used fast api python in backend to write the endpoints for our needs. There are six endpoints:

1. /getRandom20 endpoint: it gives the 20 random films which are selected from the first 100 rated films in our dataset. It is important that we used the same data that is used for trainings.

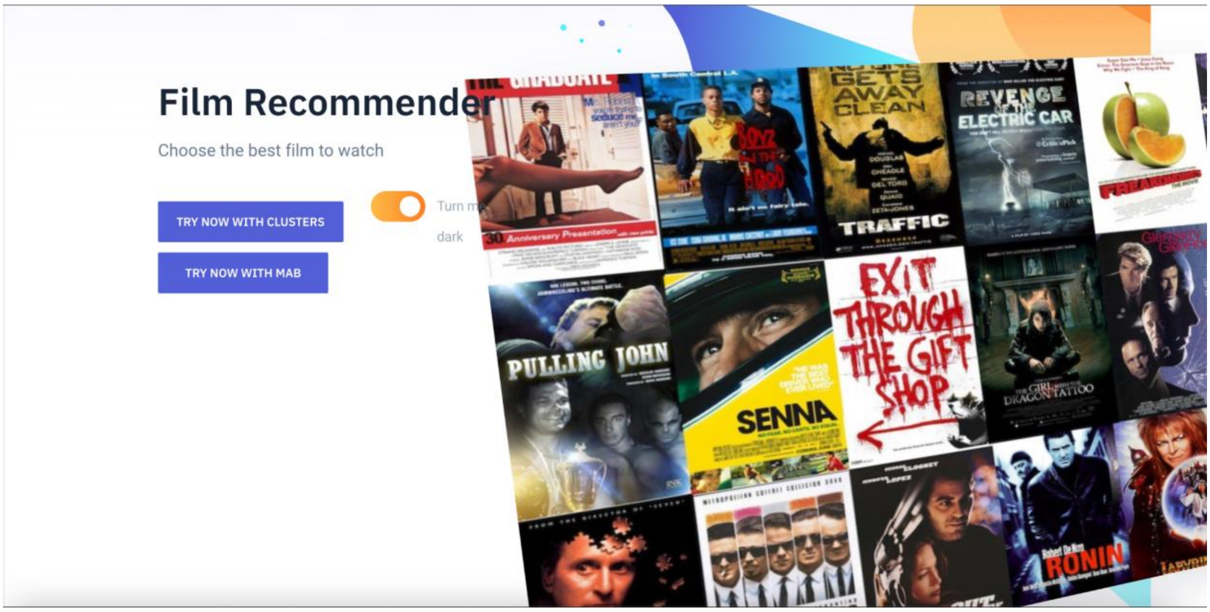
2. /getClusterKmeans(selected_movie): it gives the cluster of the selected movie among the random20films by k-mean clustering.

3. /getClusterTag(selected_movie): it gives the cluster of the selected movie among the random20films by tag clustering.

4. /getClusterSom(selected_movie): it gives the cluster of the selected movie among the random20films by Self Organizing Map(SOM) clustering.

5. /mabGet () gives ten film recommended by mab algorithms

6. /mabUpdate(movieOrder,rating) update mab film recommendation by rating and movie order



Film Recommender

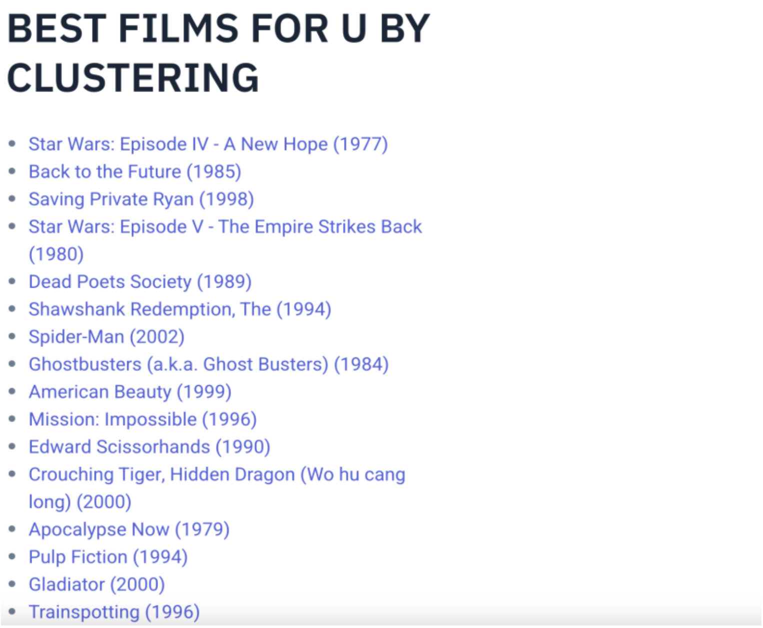
Choose the best film to watch

Figure 11: Main Page of the recommender

As you can see there are two option to choose:

1. Choosing a film with clusterings
2. Choosing a film with MAB

4.3.1. Clustering



BEST FILMS FOR U BY CLUSTERING

- Star Wars: Episode IV - A New Hope (1977)
- Back to the Future (1985)
- Saving Private Ryan (1998)
- Star Wars: Episode V - The Empire Strikes Back (1980)
- Dead Poets Society (1989)
- Shawshank Redemption, The (1994)
- Spider-Man (2002)
- Ghostbusters (a.k.a. Ghost Busters) (1984)
- American Beauty (1999)
- Mission: Impossible (1996)
- Edward Scissorhands (1990)
- Crouching Tiger, Hidden Dragon (Wo hu cang long) (2000)
- Apocalypse Now (1979)
- Pulp Fiction (1994)
- Gladiator (2000)
- Trainspotting (1996)

Figure 12: Recommendation Examples

When we choose film recommendation with clustering. At the beginning we get random 20 films among the most 100rated films. Then if we click the name and choose one of them, for example, I choose Pulp Fiction (1994). Then we get different pages showing its related cluster:

4.3.1.1. K-Means Clustering

- American History X (1998)
- Memento (2000)
- Seven (a.k.a. Se7en) (1995)
- Fight Club (1999)
- Usual Suspects, The (1995)
- Sixth Sense, The (1999)
- American Beauty (1999)
- Silence of the Lambs, The (1991)
- Shawshank Redemption, The (1994)
- Matrix, The (1999)
- Pulp Fiction (1994)

4.3.1.2. TAG Clustering

- Pulp Fiction (1994)
- Matrix, The (1999)
- Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)
- Shawshank Redemption, The (1994)
- Silence of the Lambs, The (1991)
- Star Wars: Episode V - The Empire Strikes Back (1980)
- American Beauty (1999)
- Sixth Sense, The (1999)
- Usual Suspects, The (1995)
- Godfather, The (1972)
- Fargo (1996)
- Fight Club (1999)
- Lord of the Rings: The Fellowship of the Ring, The (2001)
- Monty Python and the Holy Grail (1975)
- Saving Private Ryan (1998)
- Seven (a.k.a. Se7en) (1995)
- Schindler's List (1993)
- Braveheart (1995)
- Alien (1979)
- Die Hard (1988)
- Twelve Monkeys (a.k.a. 12 Monkeys) (1995)
- Blade Runner (1982)
- Gladiator (2000)
- Fugitive, The (1993)

4.3.1.3. Som Clustering

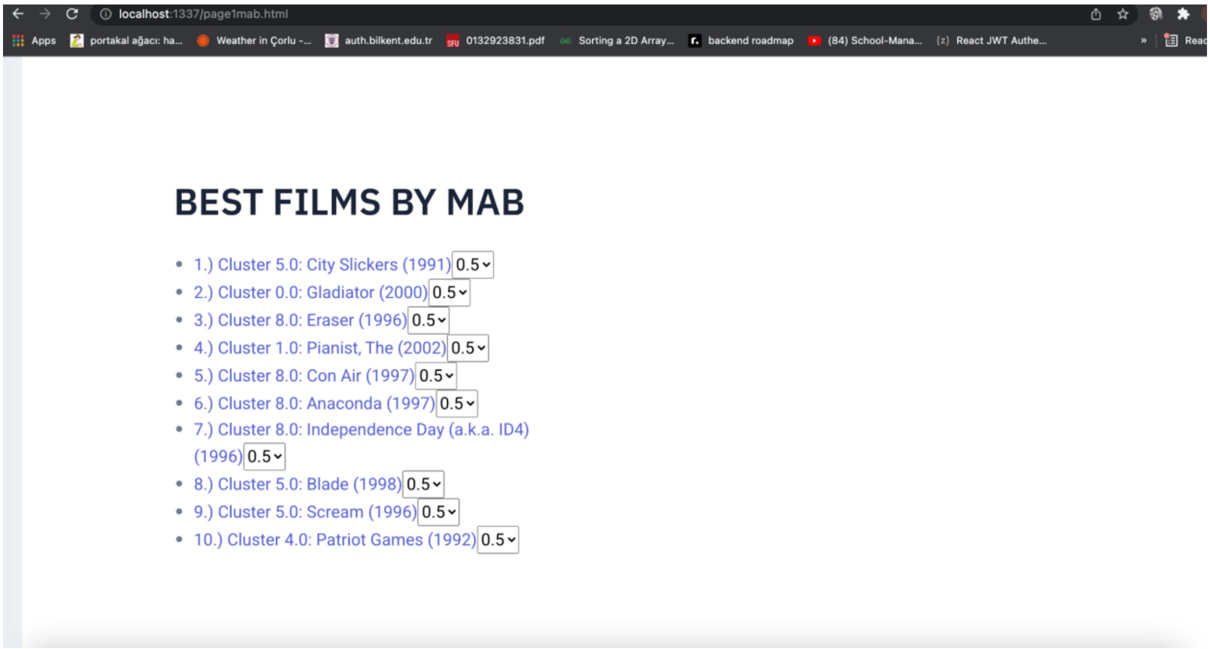
- Pulp Fiction (1994)
- Matrix, The (1999)
- Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)
- Star Wars: Episode IV - A New Hope (1977)
- Shawshank Redemption, The (1994)

- Silence of the Lambs, The (1991)
- Star Wars: Episode V - The Empire Strikes Back (1980)
- Back to the Future (1985)
- Forrest Gump (1994)
- American Beauty (1999)
- Sixth Sense, The (1999)
- Usual Suspects, The (1995)
- Star Wars: Episode VI - Return of the Jedi (1983)
- Godfather, The (1972)
- Terminator 2: Judgment Day (1991)
- Fargo (1996)
- Groundhog Day (1993)
- Fight Club (1999)
- Terminator, The (1984)
- Lord of the Rings: The Fellowship of the Ring, The (2001)
- Monty Python and the Holy Grail (1975)
- Saving Private Ryan (1998)
- Seven (a.k.a. Se7en) (1995)
- Indiana Jones and the Last Crusade (1989)

Thus, as you can see from the page, the films that are recommended by the clusterings are relevant. The survey on this results will be given later.

4.3.2. MAB

What if we want to rate the film and get a film recommendation for this rating? So to do that we simply use MAB algorithms. When we choose film recommendations with the MAB option. The images are form a special version of the recommendation system which shows the clusters of the movies for easier demonstration, normally the clusters wouldn't be visible to the user. At the beginning we get ten recommended film:

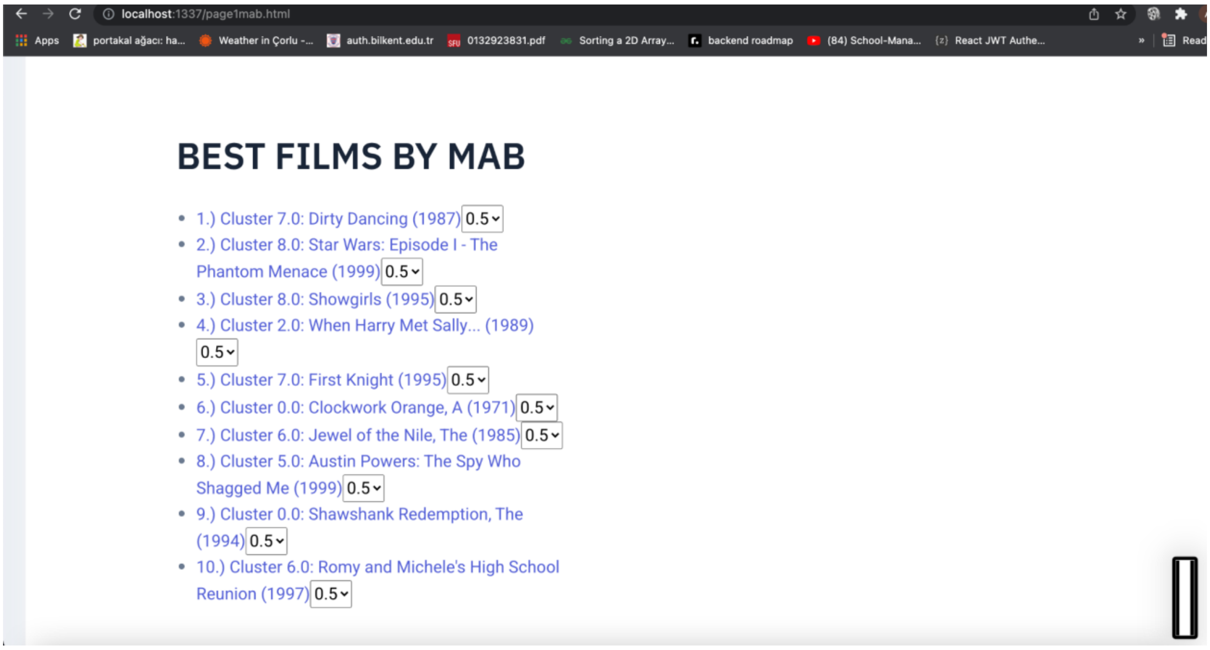


BEST FILMS BY MAB

- 1.) Cluster 5.0: City Slickers (1991) 0.5
- 2.) Cluster 0.0: Gladiator (2000) 0.5
- 3.) Cluster 8.0: Eraser (1996) 0.5
- 4.) Cluster 1.0: Pianist, The (2002) 0.5
- 5.) Cluster 8.0: Con Air (1997) 0.5
- 6.) Cluster 8.0: Anaconda (1997) 0.5
- 7.) Cluster 8.0: Independence Day (a.k.a. ID4) (1996) 0.5
- 8.) Cluster 5.0: Blade (1998) 0.5
- 9.) Cluster 5.0: Scream (1996) 0.5
- 10.) Cluster 4.0: Patriot Games (1992) 0.5

Figure 13: Best Films by MAB demonstration 1

Then by choosing a film and giving its rate, we can get this page again but with the new films that are affected from our choices. Thus if we click the name and choose one of them, for example, if we choose Gladiator (1994) from Cluster 0 and give rating as 5:



BEST FILMS BY MAB

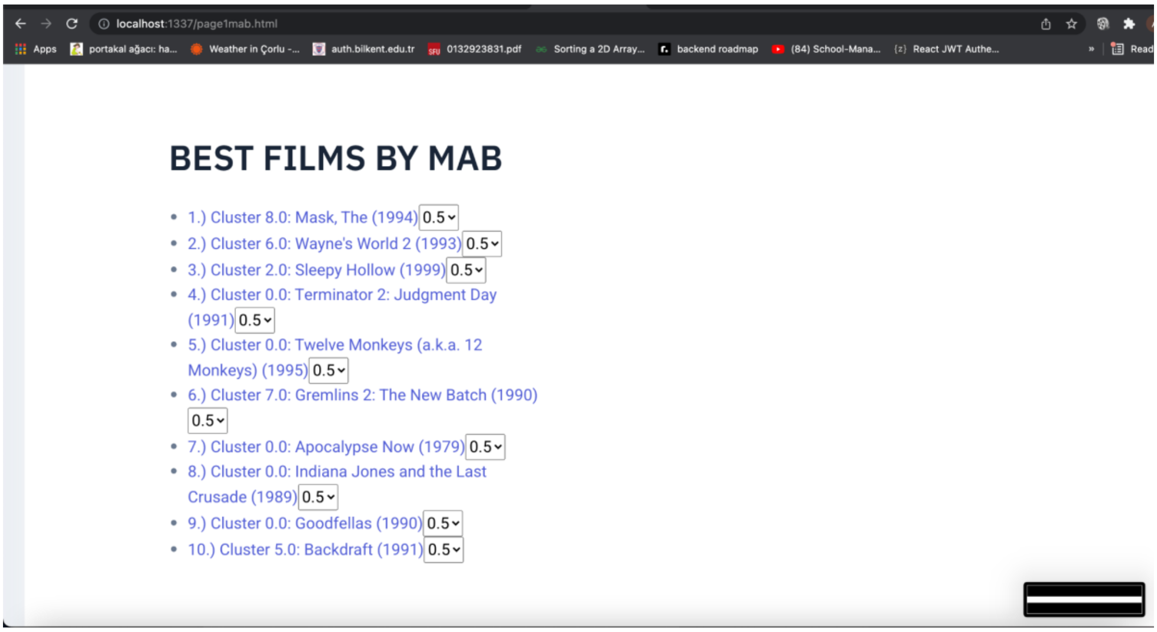
- 1.) Cluster 7.0: Dirty Dancing (1987) 0.5
- 2.) Cluster 8.0: Star Wars: Episode I - The Phantom Menace (1999) 0.5
- 3.) Cluster 8.0: Showgirls (1995) 0.5
- 4.) Cluster 2.0: When Harry Met Sally... (1989) 0.5
- 5.) Cluster 7.0: First Knight (1995) 0.5
- 6.) Cluster 0.0: Clockwork Orange, A (1971) 0.5
- 7.) Cluster 6.0: Jewel of the Nile, The (1985) 0.5
- 8.) Cluster 5.0: Austin Powers: The Spy Who Shagged Me (1999) 0.5
- 9.) Cluster 0.0: Shawshank Redemption, The (1994) 0.5
- 10.) Cluster 6.0: Romy and Michele's High School Reunion (1997) 0.5

Figure 14: Best Films by MAB demonstration 2

We get more related films and get more films according to the score that we give.

Simple test for our algorithm:

If we keep give rating 5 stars to the films which are from Cluster 0, we can get more film from this cluster:



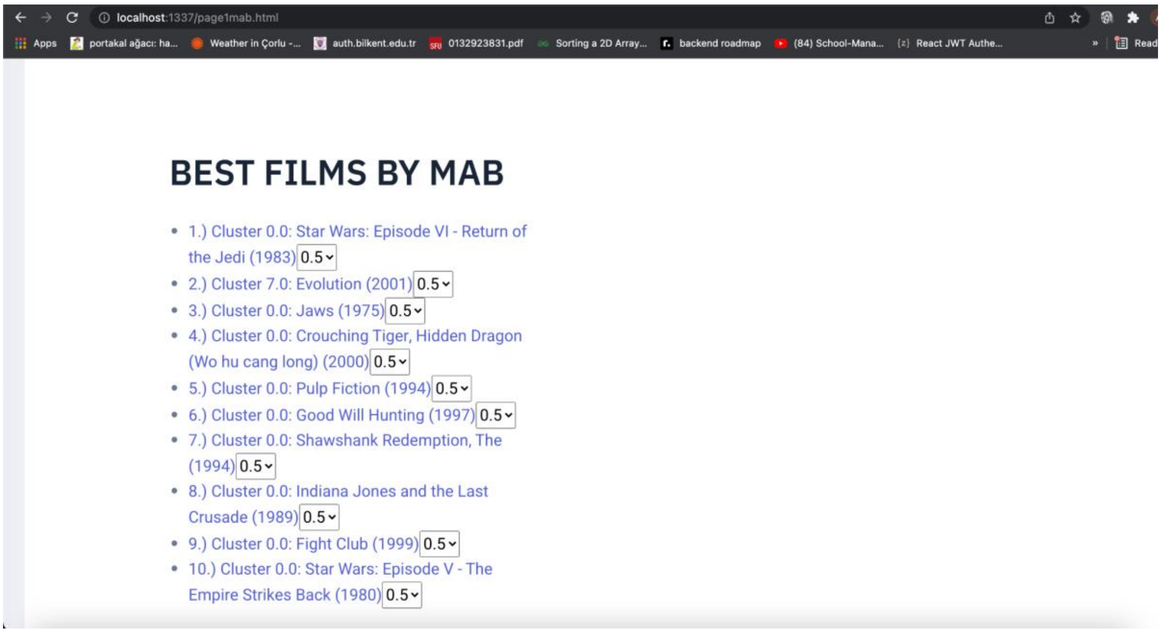
BEST FILMS BY MAB

- 1.) Cluster 8.0: Mask, The (1994) 0.5
- 2.) Cluster 6.0: Wayne's World 2 (1993) 0.5
- 3.) Cluster 2.0: Sleepy Hollow (1999) 0.5
- 4.) Cluster 0.0: Terminator 2: Judgment Day (1991) 0.5
- 5.) Cluster 0.0: Twelve Monkeys (a.k.a. 12 Monkeys) (1995) 0.5
- 6.) Cluster 7.0: Gremlins 2: The New Batch (1990) 0.5
- 7.) Cluster 0.0: Apocalypse Now (1979) 0.5
- 8.) Cluster 0.0: Indiana Jones and the Last Crusade (1989) 0.5
- 9.) Cluster 0.0: Goodfellas (1990) 0.5
- 10.) Cluster 5.0: Backdraft (1991) 0.5

[redacted]

Figure 15: Best Films by MAB demonstration 3

Check the cluster distribution. Let's keep giving 5 point to Cluster 0:



BEST FILMS BY MAB

- 1.) Cluster 0.0: Star Wars: Episode VI - Return of the Jedi (1983) 0.5
- 2.) Cluster 7.0: Evolution (2001) 0.5
- 3.) Cluster 0.0: Jaws (1975) 0.5
- 4.) Cluster 0.0: Crouching Tiger, Hidden Dragon (Wo hu cang long) (2000) 0.5
- 5.) Cluster 0.0: Pulp Fiction (1994) 0.5
- 6.) Cluster 0.0: Good Will Hunting (1997) 0.5
- 7.) Cluster 0.0: Shawshank Redemption, The (1994) 0.5
- 8.) Cluster 0.0: Indiana Jones and the Last Crusade (1989) 0.5
- 9.) Cluster 0.0: Fight Club (1999) 0.5
- 10.) Cluster 0.0: Star Wars: Episode V - The Empire Strikes Back (1980) 0.5

Figure 16: Best Films by MAB demonstration 4

It is almost all from Cluster 0. Thus we can see that our algorithm is working correctly.

5. Discussion

5.1. Clustering Algorithm comparison:

As mentioned the SOM and K-means heat maps show that the tags are not distributed homogeneously in the clusters. From the heat maps we could also see the distribution movies into the clusters. SOM's clusters have a similar number of elements compared to the k-means clustering. This is an advantage for SOM as the k-means cluster could easily run out of movies to recommend if one of the smaller clusters is liked by the user. K-pod algorithms clusters are distributed too unevenly (97% in one cluster). This may be reasoned to our dataset being particularly sparse, in which the null values are not the minority. Because of this the algorithm might be simply iterating them closer and closer in its updates.

To rigorously compare the success of the clustering methods, we should look at the movie recommendation system's results. In it participants claimed the SOM clustering results were slightly better than that of k-means. By looking at the movies directly we too agree on this decision. Because of this and the more commensurate number of elements in each of its clusters SOM was found as the better clustering algorithm for this setting.

5.2. MAB Algorithm comparison:

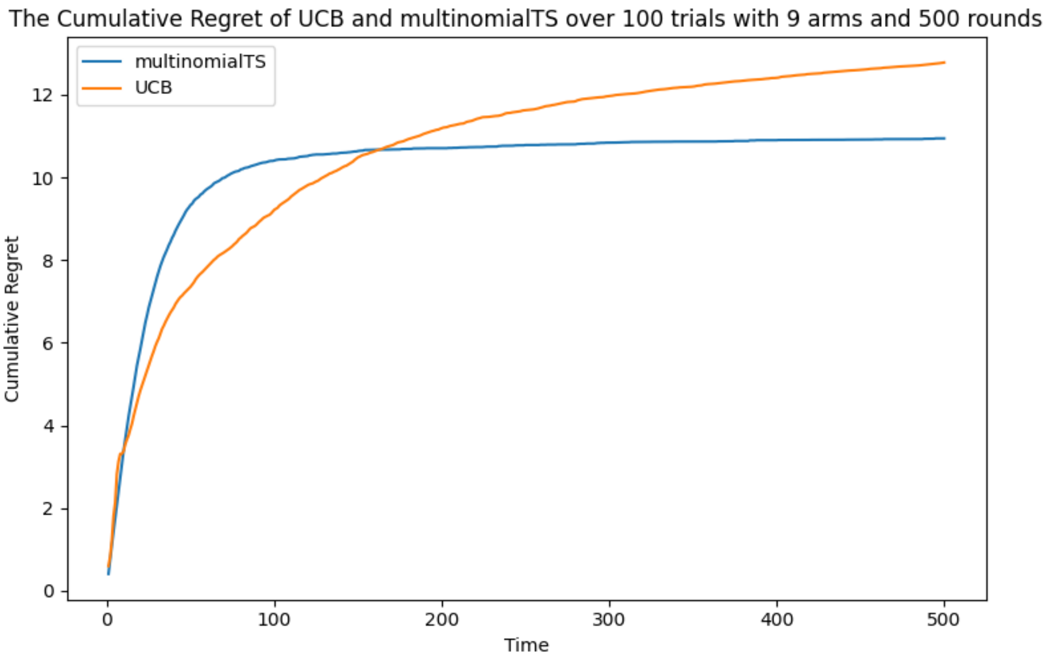


Figure 17: Comparison of Algorithms

Both MAB algorithms are working great and MTS seems to be working slightly better than UCB in terms of regret. However, the MTS algorithm has two bigger advantages over UCB in this setting.

UCB algorithm is deterministic in the arm it is recommending each round, so for a given set of previous actions it will always recommend the same arm. This means in our setting each round it will only recommend movies from the same cluster each round. On the other hand, MTS algorithm is probabilistic in the arm it is recommending each round, so even if the previous actions and rewards are the same it can recommend a different arm on the same round. This means in our setting it can recommend movies from different clusters in the same round. The latter gives a lot more diverse options to the user so it is preferable.

The other important issue is use of preliminary information. Sites like Netflix ask their new users to tell them some movies or show they like. This information could be used to head start the reinforcement learning algorithms and recommend the users movies they would like as fast as possible. Implementing this information could be done by giving the chosen movies as actions with high rewards to the MAB algorithm. However doing this for the UCB algorithm would cause it to recommend movies which are not from the clusters that are initially given as liked. Because its algorithm would try to equalize the upper confidence bounds of clusters which can only be done by sampling the not sampled clusters whereas MTS would simply recommend movies from the clusters that are initially given as liked. A recommendation system which initially doesn’t recommend things that you specifically liked would be perceived as a bad system by most users so MTS is better in this regard as well.

6. Conclusion

It is hard to calculate a numeric accuracy of the clusters because there is not a definitive “true” cluster for the movies. However even without a numeric performance metric, we can see collaborative filtering is intrinsically better than simply using user given tags or film genres. Additionally the survey results indicate SOM is slightly better than k-means in this setting. MTS is great for recommendation system applications. Its regret plots are better and its algorithm has superior qualities which makes it better for recommendation systems. We assumed a recommendation system pipeline using a combination of these approaches is the best pipeline. The whole recommendation system with the clusters cannot be tested because for it to be tested we would need a user which has given a score to all the movies, yet we can still use the system to assess if it is successful.

The date had interesting problems. Even when people who did not watch a lot of movies and movies which were not watched a lot were eliminated the data was still very sparse (80% NaN). This was one of the primary problems with the movie recommendation system. Also, because the data has more than 10000 dimensions, distance metrics lose most of their meaning and most clustering algorithms do not work. This was definitely the main problem we faced in this project as there were no easy ways to fix it or circumvent it without significantly changing our approach. We did not include them in our report however we tested all the clustering algorithms available in the sklearn package and all of them except hierarchical clustering gave results quite similar to k-pod algorithm. From this the simpler methods are better in difficult data sets such as this as there are less parts of the algorithm that could break.

References

[1] [Online]. Available: https://www.kaggle.com/grouplens/movielens-20m-dataset.

[2] J. T. Chi, E. C. Chi and R. G. Baraniuk, "K-pod: A method for k-means clustering of missing data," *The American Statistician,* vol. 70, no. 1, pp. 91-99, 2016.

[3] R. Agrawal, «Sample Mean Based Index Policies with O(log n) Regret for the Multi-Armed Bandit Problem,» *Advances in Applied Probability,* cilt 27, no. 4, pp. 1054-1078, 1995.

[4] C. Riou and J. Honda, "Bandit Algorithms Based on Thompson Sampling for Bounded," in *31st International Conference on Algorithmic Learning Theory*, 2020.

[5] T. Kohonen, «Essentials of the self-organizing map,» *Neural Networks,* cilt 37, pp. 52-65, 2013.

[6] O. Mbaabu, "Clustering in unsupervised machine learning," Section, 18 November 2020. [Online]. Available: https://www.section.io/engineering-education/clustering-in-unsupervised-ml/. [Accessed 25 November 2021].

APPENDIX

APPENDIX A - Work Allocation

- Alperen Alkan:

Implemented the frontend and the backend of the web application. Integrate ML algorithms such as MAB algorithms to the backend. Worked on the pre-process of the dataset. Worked on the clusters to choose best rating films.

- Batu Arda Düzgün:

Worked on K-means clustering and its elbow plot. Worked on the heat map of tags for the clustering algorithms. Implemented UCB, TS, MTS algorithms and their simulations with generated data.

- Ece İzmir:

Worked on the heat map of tags for the clustering algorithms. Did background research to specify the problem and requirements. Literature review for UCB, TS and MTS algorithms.

- Elif Özer:

Reprocessed the dataset (filtering users and movies for the model). Implemented clustering with neural network (SOM algorithm).

- Kadir Can Kasan:

Worked on k-means clustering and implemented k-pod clustering. Constructed the additional k-means clustering which was constructed for comparison purposes using the tag relevance scores present in the dataset.

APPENDIX B - Clusters of k-Means

Cluster 0 Prominent Movies:

1.) Independence Day (a.k.a. ID4) (1996)

2.) Star Wars: Episode I - The Phantom Menace (1999)

3.) Mask, The (1994)

4.) Austin Powers: The Spy Who Shagged Me (1999)

5.) Mrs. Doubtfire (1993)

6.) Home Alone (1990)

7.) Ace Ventura: Pet Detective (1994)

8.) Mummy, The (1999)

9.) Twister (1996)

10.) Dumb & Dumber (Dumb and Dumber) (1994)

11.) Armageddon (1998)

12.) Mars Attacks! (1996)

13.) Liar Liar (1997)

14.) Star Wars: Episode II - Attack of the Clones (2002)

15.) Con Air (1997)

16.) Jumanji (1995)

17.) Air Force One (1997)

18.) Blair Witch Project, The (1999)

19.) Honey, I Shrunk the Kids (1989)

20.) Lost World: Jurassic Park, The (1997)

21.) Batman Forever (1995)

22.) Mission: Impossible II (2000)

23.) Charlie's Angels (2000)

24.) Waterworld (1995)

Cluster 1 Prominent Movies:

1.) Groundhog Day (1993)

2.) Twelve Monkeys (a.k.a. 12 Monkeys) (1995)

3.) Truman Show, The (1998)

4.) Being John Malkovich (1999)

5.) Fifth Element, The (1997)

6.) Crouching Tiger, Hidden Dragon (Wo hu cang long) (2000)

7.) Edward Scissorhands (1990)

8.) Big Lebowski, The (1998)

9.) Léon: The Professional (a.k.a. The Professional) (Léon) (1994)

10.) Beetlejuice (1988)

11.) O Brother, Where Art Thou? (2000)

12.) Trainspotting (1996)

13.) Monty Python's Life of Brian (1979)

14.) Kill Bill: Vol. 1 (2003)

15.) Office Space (1999)

16.) Blues Brothers, The (1980)

17.) Amelie (Fabuleux destin d'Amélie Poulain, Le) (2001)

18.) Gattaca (1997)

19.) Eternal Sunshine of the Spotless Mind (2004)

20.) Clerks (1994)

21.) High Fidelity (2000)

22.) Platoon (1986)

23.) Almost Famous (2000)

24.) Donnie Darko (2001)

Cluster 2 Prominent Movies:

1.) Jurassic Park (1993)

2.) Toy Story (1995)

3.) Men in Black (a.k.a. MIB) (1997)

4.) Ghostbusters (a.k.a. Ghost Busters) (1984)

5.) E.T. the Extra-Terrestrial (1982)

6.) Gladiator (2000)

7.) Fugitive, The (1993)

8.) Good Will Hunting (1997)

9.) Princess Bride, The (1987)

10.) Shrek (2001)

11.) Batman (1989)

12.) Ferris Bueller's Day Off (1986)

13.) Rain Man (1988)

14.) Apollo 13 (1995)

15.) Indiana Jones and the Temple of Doom (1984)

16.) Lion King, The (1994)

17.) X-Men (2000)

18.) Spider-Man (2002)

19.) Pirates of the Caribbean: The Curse of the Black Pearl (2003)

20.) Ocean's Eleven (2001)

21.) Breakfast Club, The (1985)

22.) Stand by Me (1986)

23.) Dances with Wolves (1990)

24.) Minority Report (2002)

Cluster 3 Prominent Movies:

1.) Lord of the Rings: The Return of the King, The (2003)

2.) Aliens (1986)

3.) Lord of the Rings: The Two Towers, The (2002)

4.) Die Hard (1988)

5.) Braveheart (1995)

6.) Schindler's List (1993)

7.) Indiana Jones and the Last Crusade (1989)

8.) Saving Private Ryan (1998)

9.) Lord of the Rings: The Fellowship of the Ring, The (2001)

10.) Terminator, The (1984)

11.) Terminator 2: Judgment Day (1991)

12.) Star Wars: Episode VI - Return of the Jedi (1983)

13.) Forrest Gump (1994)

14.) Back to the Future (1985)

15.) Star Wars: Episode V - The Empire Strikes Back (1980)

16.) Star Wars: Episode IV - A New Hope (1977)

17.) Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)

Cluster 4 Prominent Movies:

1.) There's Something About Mary (1998)

2.) Total Recall (1990)

3.) Austin Powers: International Man of Mystery (1997)

4.) Rock, The (1996)

5.) Contact (1997)

6.) Cast Away (2000)

7.) Die Hard: With a Vengeance (1995)

8.) Predator (1987)

9.) Four Weddings and a Funeral (1994)

10.) Unbreakable (2000)

11.) RoboCop (1987)

12.) Wayne's World (1992)

13.) Interview with the Vampire: The Vampire Chronicles (1994)

14.) Superman (1978)

15.) Meet the Parents (2000)

16.) Clueless (1995)

17.) Erin Brockovich (2000)

18.) Sleepy Hollow (1999)

19.) Pleasantville (1998)

20.) Good Morning, Vietnam (1987)

21.) Scream (1996)

22.) Get Shorty (1995)

23.) Galaxy Quest (1999)

24.) Harry Potter and the Chamber of Secrets (2002)

Cluster 5 Prominent Movies:

1.) American History X (1998)

2.) Memento (2000)

3.) Seven (a.k.a. Se7en) (1995)

4.) Fight Club (1999)

5.) Usual Suspects, The (1995)

6.) Sixth Sense, The (1999)

7.) American Beauty (1999)

8.) Silence of the Lambs, The (1991)

9.) Shawshank Redemption, The (1994)

10.) Matrix, The (1999)

11.) Pulp Fiction (1994)

Cluster 6 Prominent Movies:

1.) Wizard of Oz, The (1939)

2.) Who Framed Roger Rabbit? (1988)

3.) Willy Wonka & the Chocolate Factory (1971)

4.) Fish Called Wanda, A (1988)

5.) Airplane! (1980)

6.) Amadeus (1984)

7.) Shakespeare in Love (1998)

8.) Babe (1995)

9.) Graduate, The (1967)

10.) Close Encounters of the Third Kind (1977)

11.) Raising Arizona (1987)

12.) Unforgiven (1992)

13.) This Is Spinal Tap (1984)

14.) Full Monty, The (1997)

15.) Chicken Run (2000)

16.) It's a Wonderful Life (1946)

17.) Young Frankenstein (1974)

18.) North by Northwest (1959)

19.) Vertigo (1958)

20.) Three Kings (1999)

21.) Sting, The (1973)

22.) Butch Cassidy and the Sundance Kid (1969)

23.) To Kill a Mockingbird (1962)

24.) Planet of the Apes (1968)

Cluster 7 Prominent Movies:

1.) Godfather, The (1972)
2.) Fargo (1996)
3.) Monty Python and the Holy Grail (1975)
4.) Alien (1979)
5.) Blade Runner (1982)
6.) Reservoir Dogs (1992)
7.) One Flew Over the Cuckoo's Nest (1975)
8.) L.A. Confidential (1997)
9.) Shining, The (1980)
10.) Jaws (1975)
11.) Goodfellas (1990)
12.) Clockwork Orange, A (1971)
13.) Godfather: Part II, The (1974)
14.) 2001: A Space Odyssey (1968)
15.) Apocalypse Now (1979)
16.) Full Metal Jacket (1987)
17.) Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1964)
18.) Casablanca (1942)
19.) Psycho (1960)
20.) Taxi Driver (1976)
21.) Citizen Kane (1941)
22.) Rear Window (1954)
23.) Chinatown (1974)
24.) Good, the Bad and the Ugly, The (Buono, il brutto, il cattivo, Il) (1966)

Cluster 8 Prominent Movies:

1.) Speed (1994)
2.) Titanic (1997)
3.) Mission: Impossible (1996)
4.) True Lies (1994)

5.) Back to the Future Part II (1989)

6.) Pretty Woman (1990)

7.) Top Gun (1986)

8.) Face/Off (1997)

9.) American Pie (1999)

10.) Back to the Future Part III (1990)

11.) Ghost (1990)

12.) Die Hard 2 (1990)

13.) Sleepless in Seattle (1993)

14.) GoldenEye (1995)

15.) Starship Troopers (1997)

16.) Matrix Reloaded, The (2003)

17.) Stargate (1994)

18.) Batman Returns (1992)

19.) Grease (1978)

20.) Spaceballs (1987)

21.) Crocodile Dundee (1986)

22.) Lethal Weapon 2 (1989)

23.) Devil's Advocate, The (1997)

24.) Happy Gilmore (1996)

APPENDIX C - Clusters of SOM

Cluster 0 Prominent Movies:

1.) Pulp Fiction (1994)

2.) Matrix, The (1999)

3.) Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)

4.) Star Wars: Episode IV - A New Hope (1977)

5.) Shawshank Redemption, The (1994)

6.) Silence of the Lambs, The (1991)

7.) Star Wars: Episode V - The Empire Strikes Back (1980)

8.) Back to the Future (1985)

9.) Forrest Gump (1994)

10.) American Beauty (1999)

11.) Sixth Sense, The (1999)

12.) Usual Suspects, The (1995)

Cluster 1 Prominent Movies:

1.) Edward Scissorhands (1990)

2.) Willy Wonka & the Chocolate Factory (1971)

3.) Airplane! (1980)

4.) Blues Brothers, The (1980)

5.) Clerks (1994)

6.) High Fidelity (2000)

7.) Traffic (2000)

8.) Kill Bill: Vol. 2 (2004)

9.) Close Encounters of the Third Kind (1977)

10.) Nightmare Before Christmas, The (1993)

11.) Lost in Translation (2003)

12.) Raising Arizona (1987)

Cluster 2 Prominent Movies:

1.) Jurassic Park (1993)
2.) Men in Black (a.k.a. MIB) (1997)
3.) Shrek (2001)
4.) Batman (1989)
5.) Fifth Element, The (1997)
6.) Indiana Jones and the Temple of Doom (1984)
7.) Lion King, The (1994)
8.) X-Men (2000)
9.) There's Something About Mary (1998)"]
10.) Who Framed Roger Rabbit? (1988)
11.) Spider-Man (2002)
12.) Pirates of the Caribbean: The Curse of the Black Pearl (2003)

Cluster 3 Prominent Movies:

1.) Beauty and the Beast (1991)
2.) Babe (1995)
3.) Bug's Life, A (1998)
4.) Harry Potter and the Sorcerer's Stone (a.k.a. Harry Potter and the Philosopher's Stone) (2001)
5.) Four Weddings and a Funeral (1994)
6.) Superman (1978)
7.) Chicken Run (2000)
8.) Erin Brockovich (2000)
9.) Get Shorty (1995)
10.) Galaxy Quest (1999)
11.) Harry Potter and the Chamber of Secrets (2002)
12.) Mary Poppins (1964)

Cluster 4 Prominent Movies:

1.) RoboCop (1987)

2.) Wayne's World (1992)

3.) Clueless (1995)

4.) Enemy of the State (1998)

5.) Gremlins (1984)

6.) Natural Born Killers (1994)

7.) Clear and Present Danger (1994)

8.) Eyes Wide Shut (1999)

9.) Rocky Horror Picture Show, The (1975)

10.) Bridget Jones's Diary (2001)

11.) Ice Age (2002)

12.) NeverEnding Story, The (1984)

Cluster 5 Prominent Movies:

1.) Speed (1994)

2.) Titanic (1997)

3.) Mission: Impossible (1996)

4.) True Lies (1994)

5.) Back to the Future Part II (1989)

6.) Rock, The (1996)

7.) Austin Powers: The Spy Who Shagged Me (1999)

8.) Top Gun (1986)

9.) Face/Off (1997)

10.) American Pie (1999)

11.) Die Hard: With a Vengeance (1995)

12.) Die Hard 2 (1990)

Cluster 6 Prominent Movies:

1.) Pretty Woman (1990)
2.) Back to the Future Part III (1990)
3.) Ghost (1990)
4.) Batman Returns (1992)
5.) Outbreak (1995)
6.) Alien³ (a.k.a. Alien 3) (1992)
7.) Tomorrow Never Dies (1997)
8.) Bruce Almighty (2003)
9.) Star Trek: Generations (1994)
10.) Hook (1991)
11.) Legally Blonde (2001)
12.) Perfect Storm, The (2000)

Cluster 7 Prominent Movies:

1.) Mummy, The (1999)
2.) Dumb & Dumber (Dumb and Dumber) (1994)
3.) Matrix Reloaded, The (2003)
4.) Mars Attacks! (1996)
5.) Liar Liar (1997)
6.) Star Wars: Episode II - Attack of the Clones (2002)
7.) Air Force One (1997)
8.) Blair Witch Project, The (1999)
9.) Crocodile Dundee (1986)
10.) Matrix Revolutions, The (2003)
11.) You've Got Mail (1998)
12.) Dirty Dancing (1987)

Cluster 8 Prominent Movies:

1.) Independence Day (a.k.a. ID4) (1996)

2.) Star Wars: Episode I - The Phantom Menace (1999)

3.) Mask, The (1994)

4.) Mrs. Doubtfire (1993)

5.) Home Alone (1990)

6.) Ace Ventura: Pet Detective (1994)

7.) Twister (1996)

8.) Armageddon (1998)

9.) Con Air (1997)

10.) Jumanji (1995)

11.) Honey, I Shrunk the Kids (1989)

12.) Lost World: Jurassic Park, The (1997)