Movie Recommendation System using K-means Algorithm

Maciej Janicki, Marcin Kapiszewski, Marcel Rojewski, Adam Tomys

Poznan University of Technology
Marii Skłodowskiej-Curie 5, 60-965 Poznan, Poland

Abstract. *Movie recommendation systems are commonly used by film enthusiasts. This paper examines viability and approach of using K-means clustering algorithm for creating movie recommendation system using MovieLense dataset.*

Keywords: *recommender systems, data mining, k-means, clustering, MovieLense*

1. Introduction

Modern day film enthusiasts have large catalog of movies to choose from. Due to limited time it is not possible to watch every movie that comes that out, and thus arises a need to use movie recommendation systems. Naive approaches consist of sorting movies according to some well established evaluation measure (e.g. review score on popular website) and then recommend those with the highest score. Since consumer demand often diverges from critical approval for a given movie, creation of a recommendation system predicting viewers rating was designed based on K-means clustering [1].

2. Related Work

Previous work on the topic includes research of the performance and practicality of recommendation system by creating clusters with virtual opinion leaders[2]. Study conducted by researches in India focused on using machine learning methods when creating a recommendation system [3]. This study differs from previous approaches by using simple and well know method of K-means clustering, originally introduced by MacQueen in 1967 [4], trying to create simple but reliable movie recommendation system. Also, in terms of recommendation systems, one can distinguish two types of them: user-based and product-based. Product-based systems create recommendations based on most similar products. User-based systems create recommendation by focusing on most similar users. The movie recommendation system in this study will use the latter type of system.

3. Dataset

The MovieLense Latest Datasets was chosen [5]. The dataset contained data generated by 330,975 users with around 33,000,000 ratings of 86,000 movies. During initial testing, smaller dataset of 10,000 users was selected as a subset of MoviLense dataset.

4. Algorithm

The algorithm, used as a basis for the recommendation system, applies K-means algorithm with distances calculated between user scores specific movie genres. It then predicts user's score for a movie, by calculating average score for the movie within the user's cluster.

```
function CREATE-CLUSTERS($M$: Set of all movie ratings, $G$: Set of all genres, $U$:
Set of all users, $k$: Number of clusters)
    $S$: Initially empty
    $R$: Initially empty
    for $g \in G$ do
        $M_g \leftarrow$ Subset of $M$ with genre $g$
        if $|M_g| > \frac{|M|}{1000}$ then
            Append $g$ to $S$
        end if
    end for
    for $g \in S$ do
        for $u \in U$ do
            $M_u \leftarrow$ Subset of $M$ with genre $g$ created by $u$
            $R[u][g] \leftarrow$ Mean($M_u$)
        end for
    end for
    return K-Means($U$, $R$, $k$)
end function
function Predict-Score($u$: User, $C$: Set of all user clusters, $m$: Movie, $M$ Set of all movie ratings)
    $C_u \leftarrow$ Cluster with $u$
    $M_U \leftarrow$ Subset of $M$ with ratings of movie $m$ created by $u \in C_u$
    return Mean($M_U$)
end function
```

5. Results

To evaluate the algorithm we assumed the baseline - model using an average score for a given movie. For initial model evaluation testing dataset of 10,000 was used. Baseline model's Mean Squared Error (MSE) performance for testing dataset was 0.97. The proposed K-means model achieved best performance for $k = 4$ clusters with MSE = 0.862. The results, shown on figure 1, indicate that increase in number of clusters does not increase the algorithm performance for $k > 4$ when using testing dataset. The model applied to whole dataset had MSE

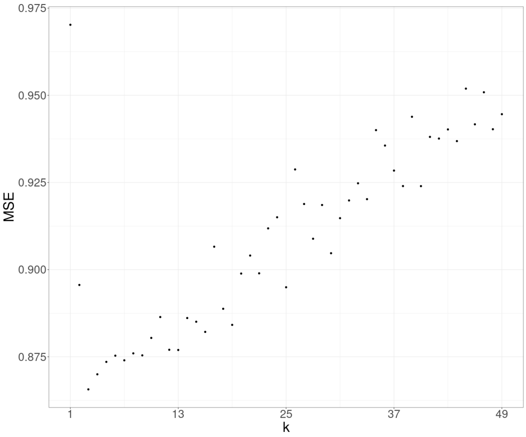

Figure 1. MSE compared to number of clusters using testing dataset of 10,000 users. $k = 1$ is the baseline.

= 0., compared to dataset baseline of MSE = 0.77. The best MSE was achieved for $k = 12$ with further increase in the number of clusters not improving the result (fig. 2).

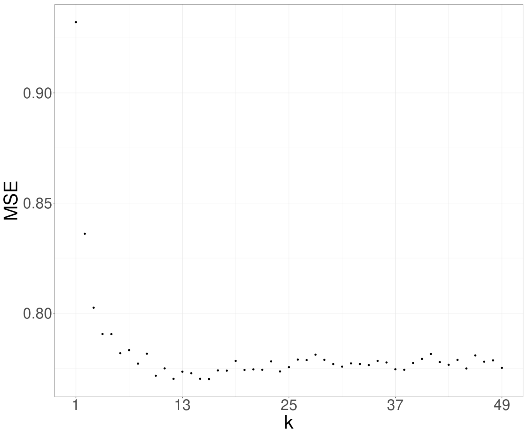

Figure 2. MSE compared to number of clusters using full MovieLense dataset. $k = 1$ is the baseline.

6. Conclusions

Movie recommendation system that uses K-means clustering to predict a review score for a given user is more effective than baseline approach of using average rating for the movie. Number of clusters required to achieve the best performance depends on the dataset and is best found by trial and error. Due to computing constraints, it is recommended to cache clustering results and only recreate them when significant changes are made to underlying dataset.

References

[1] Wallentin, E. Demand for cinema and diverging tastes of critics and audiences. *Journal of Retailing and Consumer Services*, 33:72–81, 2016. ISSN 0969-6989. doi:https://doi.org/10.1016/j.jretconser.2016.08.002. URL `https://www.sciencedirect.com/science/article/pii/S0969698916300510`.

[2] Zhang, J., Wang, Y., Yuan, Z., and Jin, Q. Personalized real-time movie recommendation system: Practical prototype and evaluation. *Tsinghua Science and Technology*, 25(2):180–191, 2020. doi:10.26599/TST.2018.9010118.

[3] Furtado, F. and Singh, A. Movie recommendation system using machine learning. *International journal of research in industrial engineering*, 9(1):84–98, 2020.

[4] MacQueen, J. et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[5] Harper, F. M. and Konstan, J. A. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4), 2015. ISSN 2160-6455. doi:10.1145/2827872. URL `https://doi.org/10.1145/2827872`.