The Borda Social Choice Movie Recommender

Johannes Kastner[1], Nemanja Ranitovic[2], Markus Endres[1]

Abstract: In this demo paper we present a recommender system, which exploits the *Borda social choice voting rule* for clustering recommendations in order to produce comprehensible results for a user. Considering existing clustering techniques like k-means, the overhead of normalizing and preparing the preferred user data is dropped. In our demo showcase we facilitate a comparison of our clustering approach to the well known k-means++ with traditional distance measures.

Keywords: Clustering, k-means, Borda, Social choice

1 Introduction

Recommendations are becoming more and more common, because the quantity of data, e.g., in online shopping platforms like Amazon, movie on-demand streaming services like Netflix and Amazon Prime Video, or music-streaming platforms, e.g., Spotify is growing continuously. In order to handle these large and confusing sets of objects easily, clustering is a very promising approach to encapsulate similar objects and to present only a few representatives of the sets to the user.

For example, consider the case where Bob wants to watch a movie. He favors *old-school movies of the late 70s to the early 90s*. He only wants to watch movies, which have a runtime *between 90 and 130 minutes*. Furthermore Bob prefers ambitious movies with a *user rating higher than 7* on a score from 0 to 10. The result of such a *preference query* (cp. [KEW11]) on a movie data set, e.g., the Internet Movie Database (IMDb) could produce large and confusing results (cp. Table 1). Now, Bob has to select one movie out of this quite confusing set of movies. This is in most cases a difficult decision, especially if the user preferences get more and more complex regarding constraints in several dimensions.

ID	movie	rating	running time	release year	genres
1	Star Wars	8.8	125	1977	Action, Sci-Fi
7	Reservoir Dogs	8.4	99	1992	Crime, Drama, Thriller
23	Indiana Jones II	7.6	118	1984	Action, Adventure, Fantasy
27	Die Hard 2	7.1	124	1990	Action, Thriller, Crime
...	...	...	...	...	...

Tab. 1: Sample result of Bob’s 4-dim preference query.

[1] Institute for Computer Science, University of Augsburg, 86135 Augsburg, Germany
<firstname>.<lastname>@informatik.uni-augsburg.de
[2] nemanja.ranitovic89@gmail.com

doi:10.18420/btw2019-31

Clustering approaches like k-means ensure that these large and confusing sets are encapsulated and presented in a clear manner to the user. However, if we consider the individual domains of each dimension, traditional distance measures, like the Euclidean distance, stretch to their limits. Since these dimensions have quite diverse domains, using traditional distance measures in k-means meet problems with this use case, because the domains are not set into an equal relation to each other. To apply k-means, one has to adjust the domains before the clustering process by normalization. However, this might be a very challenging task due to various and versatile domains.

In our work, we adapt the *Borda social choice voting rule for cluster allocation* in recommender systems. Each object is considered equally in each dimension and receives a "voting", which yields to a competitive result compared to a cluster allocation using traditional distance measures. In order to present our novel decision criterion we created an online movie recommender system to facilitate a visual comparison of using different distance measures for k-means clustering. Furthermore we include quality measures like Silhouette and Davies-Bouldin for choosing the possibly best number of desired clusters [DB79, Ro87].

2 Background

In general, social choice deals with the aggregation of individual preferences for managing social assessments and ruling. *Borda* is a voting rule, which is omnipresent in political or other elections, e.g., the Eurovision Song Contest. As mentioned in [De92], Borda is a very appealing approach to consider each dimension in a multi-dimensional scenario in an equal manner. We adapt Borda for the allocation of objects in k-means to one and only one cluster and therefore more influence of smaller domains are allowed, because every candidate receives equal weighed votes from each voter.

Given k candidates C_i, and d voters V_j, where each voter votes for each candidate. Each voter V_j has to allocate the voting $v_{jm} \in \{0, ..., k-1\}$, $m = 1, ..., k$, where all v_{jm} are pairwise distinct. Afterwards, the votes for each candidate are summed up as $bordaSum_{C_i} = \sum_{l=1}^{d} v_{li}$, while the Borda winner is determined as $bordaWinner = \max\{bordaSum_{C_i} \mid i = 1, .., k\}$.

If we apply this approach to our clustering framework, the *candidates* correspond to the available *clusters* and the *voters* to the *dimensions of the d-dimensional object which should be allocated to a cluster*. Then, for each dimension votes are assigned for the distances between the object and the centroids of the clusters. While the closest distance receives a maximum vote of k-1, the second closest a vote of k-2, the largest distance obtains a vote of 0. After the voting the sum of all votes for each cluster and subsequently the winner is determined. We integrated this novel approach into the basic k-means++ clustering algorithm as it is defined in [AV07]. Using Borda, dimensions, which would not be equally considered because of a smaller domain, get equal weighted votes like the other dimensions and have a higher influence on the clustering process. Finally we avoid the overhead of normalization.

Table 2 shows an example based on the dataset in Table 1 for our Borda cluster allocation. We want 3 clusters and present the allocation for movie (27). The centroids of the initial clusters C_1, C_2, C_3 are the movies with the IDs (1), (7), (23). For each dimension the distances between movie (27) and the centroids are calculated. The Borda votes are depicted in parentheses, e.g., the dimension *rating* is closest to C_3 and therefore gets a vote of $k - 1 = 2$. The second closest centroid is C_2 with vote 1, and C_1 gets the vote 0. Finally, C_2 with *movie (7)* as initial centroid is determined as the *bordaWinner* with a *bordaSum* of 5. Compared to the Euclidean distance we obtain a more concise result for the cluster allocation, due to ranking the values in each dimensions according to their closeness. Note that we used the Jaccard coefficient[3] for calculating the similarity of *genres* between the movie and the centroid and that a Jaccard coefficient of 1.0 is the best value.

	Movie (27): Die Hard		
Dimension	C_1	C_2	C_3
rating	1.70 (0)	1.30 (1)	**0.50 (2)**
running time	**1.00 (2)**	25.00 (0)	6.00 (1)
release year	13.00 (0)	**2.00 (2)**	6.00 (1)
genre	0.25 (1)	**0.50 (2)**	0.20 (0)
$\sum$	3	**5**	4

Tab. 2: Cluster allocation for movie (27).

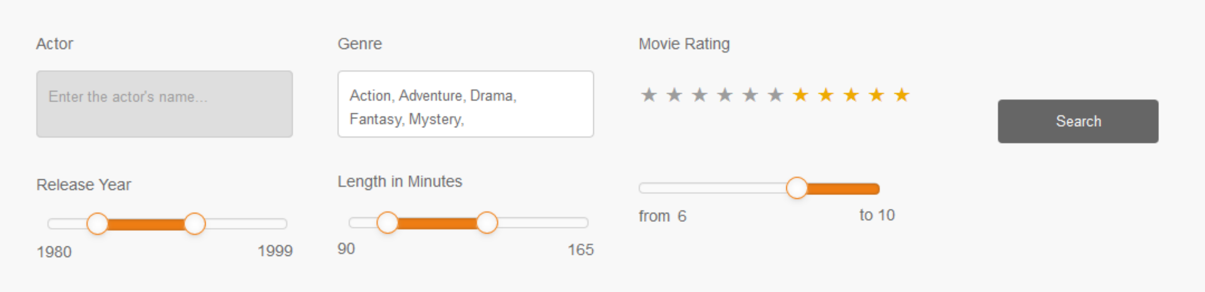
3 Showcase Application

Our demonstration scenario showcases a web application based recommender system on the IMDb. The application assists the users in finding movies which satisfy their preferences. In order to evaluate our novel Borda clustering technique in a user study, we added an evaluation mode as well. The user interface of our web-based demo-application consists of a search bar where users can determine their preferences for movie search (cp. Fig. 1). On the one hand, favorite actors can be searched and genres selected in the drop down list. On the other hand, further features of movies can be chosen, such as the release year and the length as an interval and the IMDb movie rating.

In our demo application always two clustering setups can be compared side by side. For each setup the distance measure, the number of desired clusters and the initialization with k-means++ can be chosen. Furthermore we allow the user to choose the number of desired clusters k, based on clustering quality measures. To find the most promising k, we used the quality measures Silhouette and Davies-Bouldin from which the user can choose a value.

In an extensive user study with 165 participants we evaluated our novel Borda social choice based k-means++ clustering technique against k-means++ using Euclidean and Canberra distance in order to investigate the quality of our approach on different scenarios. In each scenario sets between 50 and 60 movies where evaluated. We used this demo to show that our Borda clustering approach reaches adequate results considering the internal

[3] Jaccard: $J(A, B) = |A \cap B|/|A \cup B|$ for two sets A and B. $J_\delta(A, B) = 1 - J(A, B)$



Actor

Enter the actor's name...

Genre

Action, Adventure, Drama, Fantasy, Mystery,

Movie Rating

from 6 to 10

Release Year

1980 1999

Length in Minutes

90 165

Fig. 1: Showcase recommender search menu.

clustering quality. Furthermore we reached satisfying results in our study, so our Borda approach constitutes a worthwhile alternative to the basic k-means++ with traditional distance measures with the benefit of avoiding normalization before the clustering process.

We also investigated runtime and the number of iterations until a stable clustering is reached in [EKR19]. In summary, our *Borda* approach works nearly as good as the classic k-means algorithm w.r.t. the runtime, though the complexity of the Borda voting rule is $O(nd \cdot k^2 \cdot \log(k))$ where n is the number of d-dimensional objects, which get clustered in k clusters. However, our approach needs only a fractional part of iterations until termination and therefore is an alternative to k-means++ clustering using traditional distance measures without the overhead of normalization. While the convergence of our approach depends on the initial seeding, especially in higher dimensions, k-means++ provides an auxiliary benefit for our alternative approach.

References

[AV07] Arthur, D.; Vassilvitskii, S.: K-means++: The Advantages of Careful Seeding. In: 18th ACM-SIAM Symposium on Discrete Algorithms. SODA '07, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, pp. 1027–1035, 2007.

[DB79] Davies, D. L.; Bouldin, D. W.: A Cluster Separation Measure. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1(2):224–227, April 1979.

[De92] Debord, B.: An Axiomatic Characterization of Borda's k-choice Function. Social Choice and Welfare, 9(4):337–343, Oct 1992.

[EKR19] Endres, M.; Kastner, J.; Rudenko, L.: Analyzing and Clustering Pareto-Optimal Objects in Data Streams. In (Sayed-Mouchaweh, Moamar, ed.): Learning from Data Streams in Evolving Environments: Methods and Applications. Springer, pp. 63–91, 2019.

[KEW11] Kießling, W.; Endres, M.; Wenzel, F.: The Preference SQL System - An Overview. Bulletin of the Technical Commitee on Data Engineering, 34(2):11–18, 2011.

[Ro87] Rousseeuw, P.: Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. Journal of Comp. and Applied Math., 20:53 – 65, 1987.