Performance of Multi-Clustering Recommender System after Selection of Clusters based on V-Measures

Urszula Kużelewska
Faculty of Computer Science/Bialystok University of Technology/Wiejska 45a
Bialystok, Poland *u.kuzelewska@pb.edu.pl*

Abstract

Identification of neighbourhood based on multi-clusters has been successfully applied to recommender systems, increasing recommendation accuracy and eliminating divergence related to a difference in clustering schemes. The algorithm M-CCF was developed for this purpose that was described in author's previous papers. However, the solution do not equally take advantage on all the partitionings. Selection of clusters to forward to recommender system's input, without deterioration in recommendation accuracy, can simplify its structure. The article describes a solution of a cluster selection based on entropy measure between clustering schemes, eliminating ones, which are redundant. The results reported in this paper confirmed its positive impact on the M-CCF system's overall recommendation performance (measured by RMSE and Coverage).

Keywords: Recommender Systems, Multi-clustering, Cluster Selection, V-Measure

1. Introduction

Recommender systems (RSs) emerged as a response to the rapid development of the Internet, and as a consequence, a large expansion of distributed data. They are electronic applications to help users to reach the information or resource they are interested in, in a fast and convenient way. Their outcome is usually collected in a form of a list of recommender items, typically ranked, which is presented to users [1], [5].

Although many novel algorithms, which are complex and sophisticated, to generate recommendations were proposed by scientists, it is still an open research challenge to build a universal system which is accurate, scalable, and time efficient [14]. Clustering algorithms are attractive tools to address the vertical scalability problem [13]. They identify groups of similar objects (users or items) that can contribute to recommender systems for $a\ priori$ identification of neighbourhood objects related to a target one (e.g., a target user is a user to whom recommendations are generated). Clustering algorithms, on the other hand, have their weak points as well. First of all, most of them have input parameters, which different values highly influence final results [7]. Moreover, even the values remain the same, the outcomes can be different. It is related to the way how they work - their purpose is not to find a global optimal partition, but a local one, starting with different initial points [9]. Different clustering schemes affect the accuracy of recommendations generated by recommender systems due to changes in the neighbourhood range of target objects [4].

The disadvantages described above can be solved by techniques called alternate clustering, multi-view clustering, multi-clustering, or co-clustering [3]. They include a wide range of methods that are based on widely understood multiple runs of clustering algorithms [12] or multiple applications of a clustering process on different input data [17]. The algorithm M-CCF, which was described in [11], instead of one single-clustering scheme, works on a set of several ones, which come from several runs of a clustering method with different values of an input parameter. As a clustering method, k-means was used as the most common and comprehensive partitioning solution that was run with different values of a number of clusters. The experiments validated M-CCF against baseline predictors: an item-based recommender system, which identifies neighbourhood using k Nearest Neighbours algorithm [6] and a single-clustering recommender system, which utilises only one partitioning scheme for this purpose [13]. The advantage of M-CCF was gained in terms of recommendation quality. The selection of particularly effective clustering scheme, reduces the number of clusters to analyse by M-CCF and has a positive impact on recommendations accuracy. As a selection criterion, clusters' similarity was used, in terms of their compactness and homogeneity, expressed by V-measure [8]. Similar clusters, perceived as redundant, were deleted from the input set.

Our main contributions are as follows. Selection of clusters to forward to the M-CCF input is beneficial for its performance in terms of recommendation accuracy and coverage (measured respectively by RMSE and Coverage). Criteria based on V-measure is a suitable approach to identify redundant clusters, thereby supplying the selected clusters to the M-CCF input.

The article is organised as follows: the following section describes related work concerning clustering-based recommender systems with a cluster selection procedure. The next section presents the proposed algorithm, whereas the following one contains results of the performed experiments. The last section concludes the paper.

2. Related Work

Each clustering algorithm has its strengths and weaknesses. On a given data set, different algorithms or the same algorithms with different input parameters often have distinct clusterings. To address this issue, a concept of $cluster\ ensemble$ or $clustering\ aggregation$ is emerged to integrate several partitionings into a final outcome [15]. One of the approaches to this concept that generate a set of base clustering schemes is to run a single clustering algorithm with different initial sets of parameters several times [2]. Then, a cluster selection procedure can be applied to determine the relevant ones to a particular problem.

Evaluation of clustering algorithms performance is not a trivial task due to the lack of both group labels and precisely formulated objectives. In some cases if the labels can be delivered for evaluation, it is possible to use them in so-called external indices, e.g. Rand index, Fowlkes-Mallows score [10]. If they are not available, the only option is to use internal measures that exploit similarity among objects of data, e.g. Silhouette, Dunn, DB indices [10].

Cluster ensembles or cluster selection only is widely used in data mining tasks, including recommendation generation. In [2] it is proposed a k-means-based method, called KMCE, which selects a final result from many base clustering schemes. It evaluates the local credibility of each cluster label, building the relationship between clusters, and generating the final outcome. The authors used the Rand index as one of the evaluation criteria. Recommendation accuracy was raised in [16] by application a combination of PCA and k-means methods. The authors used Dunn index to evaluate clusterings. In [18] the authors used k-means and to avoid convergence in clustering results applied a procedure of initial centroid selection, which discovered underlying data correlation structures. They compared the proposed solution to base one, in which cluster centers are initialized randomly. As a result, recommendation accuracy and coverage have been improved.

3. Description of M-CCF Algorithm

The novel solution consists of multiple types of clustering schemes that are provided for the M-CCF method's input. It is implemented in the following way (for the original version, with one type of a clustering scheme, check in [11]).

Step I. Multiple clustering

The first step of M-CCF is to perform clustering on the input data. The process is conducted several times and all results are stored in order to deliver them to the algorithm. In this paper, k-means was selected as a clustering method, which was executed for $k = 5, 20, 50, 100$ to generate input schemes for one M-CCF RS system.

Step II. Clustering schemes' selection procedure

In M-CCF, one of the external evaluation measures, V-measure [8], was applied (1), which uses conditional entropy analysis.

$$v = \frac{(1+\beta) \cdot h \cdot c}{\beta \cdot h + c},\ h = 1 - \frac{H(C|K)}{H(C)},\ c = 1 - \frac{H(K|C)}{H(K)} \tag{1}$$

The homogeneity (h) measures the uniformity of original class labels distributed within every cluster. The higher values of homogeneity indicate a greater number of points from the same original class. Completeness (c) refers to the distribution of all members of a given class over the entire clustering scheme - if they are not spread over many clusters, the completeness is higher. The constant β assigns importance between the components. In this paper $\beta = 1$ that means that both components are significant equally. The component of both homogeneity and completeness formulas, $H(C|K)$ stands for a conditional entropy of the classes C and K given the cluster assignments. The component $H(K)$ is the entropy of the class K (2).

$$H(C|K) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|} \frac{n_{c,k}}{n} \log(\frac{n_{c,k}}{n_k}),\ H(K) = -\sum_{k=1}^{|K|} \frac{n_k}{n} \log(\frac{n_k}{n}) \tag{2}$$

The definition of $H(K|C)$ as well as $H(C)$ are analogous. In the formulas above, n is the total number of samples, n_c and n_k are the number of samples respectively belonging to class C and cluster K, and finally $n_{c,k}$ the number of samples from class C assigned to cluster K.

Step III. Building M-CCF RS system

It is a vital issue to have precise neighbourhood modelling for all input data. It consists in iterating every input object and selecting the best cluster from C set for it. The term $best$ refers to the cluster which center is the most similar to the particular input object. Then, when all input data have their connected clusters, traditional CF systems are built on these clusters. As a result, the M-CCF algorithm is created - a complex of recommender systems formed on their clusters.

Step IV. Recommendation generation

When generating recommendations for an active user, a relevant RS from M-CCF is selected. It is also based on the similarity between them and cluster centers. Then, the process of recommendation generation is performed, however, searching for similar objects is limited to the cluster connected to the particular recommender in M-CCF. When a neighbourhood is modelled by a single-clustering method, the border objects have fewer neighbours in their closest area than the objects located in the middle of a cluster. The multi-clustering prevents such situations, as it identifies clusters in which particular users are very close to its center.

4. Results and Discussion

The goal of the experiments was to verify whether the selection of clusters based on V-measure is efficient in the M-CCF recommender system. In other words, whether the performance of M-CCF is affected as a result of removing the clusters that are negatively evaluated by this index from the input set of clustering schemes.

The experiments were divided into 2 phases: clustering and clusters' evaluation and generation of recommendations and a measurement of its accuracy. In the first phase, k-means was taken as it is the most common clustering algorithm and has been successfully deployed in the previous version of the M-CCF approach.

Two subsets of a MovieLens dataset [19] were taken for this test. Originally, the data contained 25 million ratings, however in the experiments randomly selected samples were taken. The subsets, a small one consisting of 100 000 ratings ($100k$) and a big one consisting of 1 million ratings ($1M$) are presented in Table 1. Both datasets were split into training and testing parts in a proportion of about 100 to 1. Note, that the small set is more sparse than the big one - containing fewer ratings per item: 9.06 in comparison with 60.49. The training parts were clustered and forwarded to the M-CCF algorithm. The test part were used for recommendation evaluation.

Table 1. Description of the datasets (first row - training, second - test) used in the experiments.

Dataset	No. of ratings	No. of users	No. of items	ratings\user ratio	ratings\item ratio
small - $100k$	99 835	549	11 024	181.85	9.06
big - $1M$	991 116	5 430	16 384	182.52	60.49
small - $100k$	1 176	110	969	10.69	1.21
big - $1M$	8 962	1088	3840	8.24	2.33

4.1. Clustering and Evaluation of Clustering Schemes

The clustering was executed several times with the following value of k, which stands for a number of groups: 5, 20, 50 and 100. Furthermore, two types of distance measures were used: Cosine-based (CD) and Euclidean (ED). It was decided to cluster the items (movies). For this reason, in the following phase, item-item recommender systems were selected to use. The opposite version - users' clustering - was also examined, however, the problem arose in the groups. They were composed of one great cluster, which contained about 50% of data, and many very small ones with many users remained nonclustered.

Every run of k-means with the same value of k was repeated 6 times, thus obtaining 6 different competitive clustering schemes. Then V-measure was used to evaluate their mutual homogeneity and completeness. The implementation in Python's Scikit Learn library was utilized [20] for this purpose. Tables 2 and 3 report values of V-measure for pairs of the 6 clustering schemes obtained on the datasets: $100k$ and $1M$ respectively. The comparison of homogeneity and completeness in pairs provided information about clusters' coincidence, that is, if the clustering schemes were evaluated by high values of V-measure, they are more homogeneous and have comparable partitions. Such schemes were recognized as correlated or redundant and removed from the set of clustering schemes. Hence, only partitions that were dissimilar to each other, that is with low V-measure values, were forwarded to the input of the recommender.

The values in the tables tend to increase with a growth of k value, however, they differ in the case of particular distance measures. The lowest V-measure is observed for Euclidean one. Finally, the following schemes were selected to proceed: CD: 1,4 ($k = 5$), 1,3 ($k = 20$), 4,6 ($k = 50$), 4,6 ($k = 100$), ED: 1,6 ($k = 5$), 3,5 ($k = 20$), 3,5 ($k = 50$), 4,5 ($k = 100$).

In Table 3 the values of V-measure are slightly higher for $k = 5$, whereas for the remaining number of groups they are comparable. The schemes obtained by an approach using the CD measure were dissimilar - a range of the best values is [0.66;0.83]. The second solution was definitely worse in the terms of diversity of clustering schemes. Finally, the following schemes were selected to proceed: CD: 1,3,4,6 ($k = 5$), 3,5 ($k = 20$), 1,5 ($k = 50$), 5,6 ($k = 100$), ED: 4,6 ($k = 5$), 2,3 ($k = 20$), 1,4 ($k = 50$), 3,4 ($k = 100$).

Again, if 2 pairs of clustering schemes were equally dissimilar (V-measure was the same), the remaining combinations containing the clusters from the best pairs would be examined. If V-measure was high (above 0.8), the cluster from such combination was removed as too similar.

Table 2. Cluster quality evaluated by V-measure on $100k$ dataset. The best values are in bold.

Clustering Schemes	Cosine-Based Distance - CD				Euclidean Distance - ED			
	5 gr	20 gr	50 gr	100 gr	5 gr	20 gr	50 gr	100 gr
1,2	0.72	0.83	0.82	0.84	0.85	0.86	0.82	0.84
1,3	0.66	**0.77**	0.83	0.83	0.80	0.74	0.79	0.87
1,4	**0.61**	0.78	0.85	0.82	0.81	0.78	0.83	0.85
1,5	0.85	0.79	0.84	0.86	0.79	0.86	0.80	0.85
1,6	0.76	0.80	0.84	0.85	**0.54**	0.78	0.82	0.85
2,3	0.69	0.81	0.86	0.82	0.72	0.74	0.84	0.83
2,4	0.66	0.80	0.83	0.83	0.73	0.80	0.83	0.83
2,5	0.68	0.81	0.84	0.84	0.71	0.82	0.83	0.83
2,6	0.79	0.81	0.83	0.88	0.55	0.76	0.83	0.84
3,4	0.70	0.80	0.84	0.84	0.95	0.75	0.82	0.84
3,5	0.68	0.80	0.82	0.84	0.96	**0.73**	**0.78**	0.84
3,6	0.71	0.80	0.83	0.82	0.58	0.74	0.82	0.84
4,5	0.69	0.84	0.86	0.84	0.93	0.82	0.80	**0.82**
4,6	0.68	0.81	**0.81**	**0.81**	0.58	0.74	0.82	0.83
5,6	0.72	0.82	0.83	0.85	0.58	0.76	0.84	0.84

In Table 3, for $k = 5$ and CD the pairs: 1,4 and 3,6 were the most distinct, however V-measure for the pairs: 1,3 (0.72) and 1,6 (0.72) and 3,4 (0.75) and 4,6 (0.69), did not exceeded 0.8. For this reason, both pairs were forwarded to the recommender system. In contrast, for the same distance measure and $k = 20$, the pair 1,5 was removed, although its V-measure was equal to the best. However, the value for the pair 1,3 was greater than 0.8.

4.2. Evaluation of Recommendations

The best clustering schemes were forwarded to the M-CCF, which was evaluated in terms of accuracy against the same algorithm but with all clustering schemes on its input. Both recommenders were used to estimate missing ratings in the testing part of the datasets and then the calculated values were compared to the original ones in order to determine a difference in precision. Despite the accuracy, attention was paid to the completeness of recommendation lists generated by the systems. Evaluation criteria were the following standard main metrics:

Root Mean Squared Error (RMSE) (3) a baseline way to measure the error in model evaluation studies. The lower value of RMSE stands for a better prediction ability.

$$RMSE = \sqrt{\frac{1}{n \cdot k} \sum_{i=1}^{n} \sum_{j=1}^{k} (r_{real}(x_{ij}) - r_{est}(x_{ij}))^2}, r_{real}, r_{est} \in [2, 3, 4, 5] \tag{3}$$

Coverage (4) measures the system's responsiveness to the required length of a recommendation list. It is a portion of generated predictions to the needed length. During the evaluation process, there were cases in which estimation of ratings was not possible. It often occurs when the item for which the calculations are performed, is not present in the cluster that contains the other user's items, which were already rated by them. It was assumed that RMSE is significant if Coverage is greater than 90%.

$$Coverage = \frac{\forall_{i=1}^{N} \forall_{j=1}^{k} \parallel r_{est}(x_{ij}) > 0 \parallel}{N} \cdot 100\% \tag{4}$$

The symbols in the equations, as well as the method of calculation, are characterised in detail below. In all equations, n is a number of users taken for evaluation, k is a number of ratings to be estimated and the number of required recommendations is denoted as N.

Table 3. Cluster quality evaluated by V-measure on $1M$ dataset. The best values are in bold.

Clustering Schemes	Cosine-Based Distance - CD				Euclidean Distance - ED			
	5 gr	**20 gr**	**50 gr**	**100 gr**	**5 gr**	**20 gr**	**50 gr**	**100 gr**
1,2	0.69	0.84	0.83	0.84	0.85	0.92	0.93	0.83
1,3	0.72	0.84	0.85	0.84	0.91	0.83	0.92	0.82
1,4	**0.66**	0.86	0.83	0.84	0.94	0.85	**0.90**	0.82
1,5	0.87	0.78	**0.82**	0.86	0.81	0.84	0.92	0.81
1,6	0.72	0.82	0.83	0.84	0.81	0.83	0.93	0.80
2,3	0.67	0.81	0.84	0.84	0.90	**0.81**	0.92	0.80
2,4	0.77	0.83	0.84	0.84	0.87	0.85	0.91	0.79
2,5	0.69	0.79	0.86	0.84	0.88	0.82	0.92	0.82
2,6	0.74	0.83	0.83	0.85	0.87	0.82	0.95	0.82
3,4	0.75	0.82	0.83	0.84	0.90	0.92	0.94	**0.78**
3,5	0.71	**0.78**	0.83	0.85	0.83	0.94	0.95	0.80
3,6	**0.66**	0.81	0.83	0.84	0.83	0.94	0.92	0.82
4,5	0.68	0.79	0.84	0.84	0.81	0.93	0.93	0.80
4,6	0.69	0.82	0.84	0.86	**0.80**	0.91	0.91	0.81
5,6	0.72	0.80	0.84	**0.83**	0.96	0.93	0.92	0.85

In the evaluation process, the values of ratings from the testing part were removed and estimated by the systems. Although the test set itself remained constant during the experiments the values to remove and estimate were selected randomly every time. The difference between the original and the calculated value (represented, respectively, as $r_{real}(x_{ij})$ and $r_{est}(x_{ij})$ for user x_i and a particular item j) was taken for RMSE calculation.

This part of the experiments started from systems' evaluation on the small dataset. Table 4 contains the results - RMSE (the first value in the every case) and Coverage (the second value, in brackets) values. The following similarity measures were used to calculate affinity between items in both approaches: Cosine-based, LogLikelihood, Pearson correlation, both Euclidean and CityBlock distance-based and Tanimoto coefficients. In the table mentioned above, the column with Pearson correlation is missing due to the value of Coverage being below 90% in every case.

To have a compact view of the obtained results without reduction of the general concept to confirm in the experiments, only selected results are reported, which were generated by both versions of the M-CCF recommender system: with and without selection of clustering schemes. The results in the tables were selected to be significant in terms of Coverage. The examples which were omitted had a value of Coverage<90%.

Table 4 contains the following configurations the M-CCF systems for $100k$ set (labels in the brackets are used in the table): M-CCF-cos-5-20-50 (M-CCF-c-1) - a set of 18 clustering schemes with $k = 5, 20, 50$ using CD - 6 schemes per one k value, M-CCF-cos-5-20-50* (M-CCF-c-2) - a set of 6 clustering schemes with $k = 5, 20, 50$ using CD - selected using V-measure, M-CCF-eu-5-20 (M-CCF-e-1)- a set of 12 clustering schemes with $k = 5, 20$ using ED - 6 schemes per one k value, M-CCF-eu-5-20* (M-CCF-e-2)- a set of 4 clustering schemes with $k = 5, 20$ using ED - selected using V-measure,

The values in bold in Table 4 denote improvement in RMSE or Coverage indices. It can be observed that in most cases, the selection of clustering schemes improved either on lower RMSE or greater Coverage values. There are only 2 situations that the results were worse after the scheme selection. It does not refer to the inappropriate values of V-measure because the ones for Euclidean distance were low, in comparison to the other instances.

Next, the same evaluation procedure was applied to the big dataset. Table 4 presents the results, as well. It contains the following configurations the M-CCF systems for $1M$ set: M-CCF-cos-5-20-50-100 (M-CCF-c-3) - a set of 24 clustering schemes with $k = 5, 20, 50, 100$

Table 4. RMSE of the algorithms on both datasets. The numbers in bold denote improvement in indices' values. The best values in the table are underlined.

Algorithm	Similarity Measure					
	Cosine-Based	LogLike-lihood	Pearson	Euclidean	CityBlock	Tanimoto
M-CCF-c-1	0.96(91%)	-	-	0.94(90%)	-	-
M-CCF-c-2*	**0.94(92%)**	**0.99(91%)**	-	**0.92(92%)**	**0.99(92%)**	**0.98(91%)**
M-CCF-e-1	0.94(93%)	0.98(92%)	-	0.93(93%)	1.00(94%)	0.99(92%)
M-CCF-e-2*	0.94(93%)	1.00(93%)	-	**0.92(93%)**	1.01(94%)	**0.99(93%)**
M-CCF-c-3	0.87(92%)	0.90(91%)	0.91(91%)	0.86(92%)	0.90(91%)	0.89(91%)
M-CCF-c-4*	0.88(94%)	**0.90(93%)**	**0.90(92%)**	0.87(94%)	**0.90(93%)**	**0.89(93%)**
M-CCF-e-3	0.95(93%)	0.95(94%)	0.93(92%)	0.93(93%)	0.95(94%)	0.92(93%)
M-CCF-e-4*	**0.94(95%)**	**0.94(95%)**	**0.92(94%)**	**0.93(95%)**	**0.94(95%)**	**0.92(95%)**
M-CCF-e-5*	**0.94(97%)**	**0.94(97%)**	**0.91(96%)**	**0.93(97%)**	**0.94(97%)**	**0.91(97%)**

using CD - 6 schemes per one k value, M-CCF-cos-5-20-50-100* (M-CCF-c-4) - a set of 8 clustering schemes with $k = 5, 20, 50, 100$ using CD - selected using V-measure, M-CCF-eu-5-20-50 (M-CCF-e-3) - a set of 18 clustering schemes with $k = 5, 20, 50$ using ED - 6 schemes per one k value, M-CCF-eu-5-20-50* (M-CCF-e-4) - a set of 6 clustering schemes with $k = 5, 20, 50$ using ED - selected using V-measure, M-CCF-eu-5-20* (M-CCF-e-5) - a set of 4 clustering schemes with $k = 5, 20$ using ED - selected using V-measure,

In general, the accuracy of recommendations lists is greater for $1M$ dataset, which contains more ratings and is denser - the ratio of ratings per item is over 6 times greater. There is no analogy in RMSE values and configurations with the results obtained on $100k$ dataset. In this case, the lowest RMSE were for the schemes clustered with Cosine-based distance. However, in most cases, improvement in both accuracy and, particularly, Coverage is observed when M-CCF works with selected clustering schemes. To analyse the values of V-measures and RMSE no distinct relationship between them is observable, however, the schemes obtained with CD were evaluated the best. Moreover, for the case M-CCF-e-4*, removing the schemes for $k = 50$ due to its value V-measure>=0.9 benefited performance of M-CCF.

5. Conclusions

In this paper, a collaborative filtering recommender system based on multi-clustering neighbourhood modelling with clustering schemes selection is presented. The concept of the M-CCF algorithm is to store multiple clustering schemes on its input and dynamically match every item that takes part in the recommendation generation process with the most appropriate cluster. Similar partitionings are redundant and do not contribute to the recommendation phase. Clustering index V-measure compares clustering schemes in terms of compactness and homogeneity and identifies ones that are highly coincident or totally opposite, that allows making the selection of clustering schemes to forward for the M-CCF input. An exclusive set of partitions often benefits M-CCF performance measured by RMSE and Coverage. The experiments validated that the performance of M-CCF is often better when it works on a reduced set of input clusters.

Future experiments will be performed to validate the proposed approach on datasets of greater size. It is planned to check the impact of different types of a clustering method and a mixture of clustering schemes instead of one-algorithm output. Additionally, the research concerning an impact on time and memory consumption are also considered.

Acknowledgments

The work was supported by the grant from Bialystok University of Technology WZ/WI-IIT/2/

2020 and funded with resources for research by the Ministry of Education and Science in Poland.

References

1. Aggrawal, C.C.: Recommender Systems. The Textbook. Springer (2016)
2. Bai, L., Liang, Y., Cao, F.: A multiple k-means clustering ensemble algorithm to find nonlinearly separable clusters. Information Fusion **61**, 36-47 (2020)
3. Bailey, J. Alternative Clustering Analysis: a Review. Intelligent Decision Technologies: Data Clustering: Algorithms and Applications, Chapman and Hall/CRC, pp. 533–548 (2014)
4. Bilge A., Polat, H.: A Scalable Privacy-preserving Recommendation Scheme via Bisecting K-means Clustering. Information Process Management **49**(4), 912-927 (2013)
5. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender Systems Survey. Knowledge-Based Systems **46**, 109-132 (2013)
6. Desrosiers, C., Karypis, G.: A Comprehensive Survey Of Neighborhood-Based Recommendation Methods. Recommender Systems Handbook, Springer, pp. 107-–144 (2011)
7. Fränti, P., Sieranoja, S.: How much can k-means be improved by using better initialization and repeats? Pattern Recognition **93**, 95-112 (2019)
8. Hirschberg, J.B., Rosenberg, A.: V-Measure: A Conditional Entropy-based External Cluster Evaluation. Proceedings of EMNLP, (2007)
9. Jain, A.K., Murty, M.N., Flynn, P.J.: Data Clustering: A Review. ACM Comput. Surv. **31**, 264-323 (1999)
10. Kaufman, L.: Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley & Sons (2009)
11. Kużelewska, U.: Effect of Dataset Size on Efficiency of Collaborative Filtering Recommender Systems with Multi-clustering as a Neighbourhood Identification Strategy. Advances Intelligent Systems and Computing, In: Proceedings of Computational Science – ICCS 2021, Springer, pp. 342–354 (2020)
12. Li, T., Ogihara, M., Ma, S.: On combining multiple clusterings: An overview and a new perspective. Applied Intelligence **33(2)**, 207-219 (2010)
13. Sarwar, B.: Recommender Systems for Large-Scale E-Commerce: Scalable Neighborhood Formation Using Clustering. In: Proceedings of the 5th International Conference on Computer and Information Technology (2002)
14. Singh, M.: Scalability and Sparsity Issues in Recommender Datasets: a Survey. Knowledge and Information Systems, Springer, pp. 1–43 (2018)
15. Strehl, A., Ghosh, J.: Cluster ensembles: a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3**, 583-617 (2002)
16. Yadav, V., Shukla, R., Tripathi, A., Maurya, A.: A New Approach for Movie Recommender System using K-means Clustering and PCA. Journal of Scientific & Industrial Research **80**, 159-165 (2021)
17. Yaoy, S., Yuy, G., Wangy, X., Wangy, J., Domeniconiz, C., Guox, M.: Discovering Multiple Co-Clusterings in Subspaces. In: Proceedings of the 2019 SIAM International Conference on Data Mining, pp. 423–431 (2019)
18. Zahra, S., Ghazanfar, M.A., Khalid, A., Azam, M.A., Naeem, U., Prugel-Bennett, A.: Novel centroid selection approaches for KMeans-clustering based recommender systems. Information Sciences **320**, 156-189 (2015)
19. MovieLens Datasets. https://grouplens.org/datasets/movielens/25m/ (13 Feb 2022)
20. Scikit Learn Clustering Indices. https://scikit-learn.org/stable/modules/model_evaluation.html (10 Feb 2022)