Journal of Ambient Intelligence and Humanized Computing (2021) 12:10629–10636
https://doi.org/10.1007/s12652-020-02876-1

ORIGINAL RESEARCH

An optimized item-based collaborative filtering algorithm

Chigozirim Ajaegbu[1]

Received: 25 May 2020 / Accepted: 23 December 2020 / Published online: 23 January 2021

Abstract
Collaborative filtering over the years have emerged as an alternative recommender system to address some of the setbacks of content based filtering. Although, Collaborative filtering has offered some benefits to the majority of the online stores in recommending products to users using users' ratings of similarity measure, its usage has also raised some doubt in the minds of researchers, regarding its effectiveness in handling ratings with limited number of users or no rating record from users. Thus, this has resulted in efforts by researchers in determining further ways to combat the issues attributed with the existing collaborative filtering techniques in terms of data sparsity or cold-start situations. This study focused on improving the traditional similarity measurements that currently exist on the item-based collaborative filtering, in order to accommodate and mitigate further the issue of cold-start situations. Thus, this study proposed an algorithm which is meant to balance the three current traditional measurement metrics such as: Cosine-based similarity, Pearson correlation similarity and Adjusted cosine similarity, in the direction of cold-start situations. The improved algorithm of traditional measurement metrics were further compared with the existing algorithm of the traditional metrics. Results showed that the proposed algorithm offered a better item-based collaborative filtering algorithm to recommendation systems than the existing, using data set from Movielens recommender system. Hence, the proposed algorithm did not only mitigate the drawbacks experienced with the three traditional algorithms in terms of data sparsity or cold-start situations but also retained the good features of the existing item-based collaborative filtering algorithm. Thus, the proposed algorithm complemented the strength of the three traditional measurement metrics with evidence shown when measured with Mean Absolute Error.

Keywords Collaborative filter · Recommender system · Cosine-based similarity · Pearson correlation similarity · Adjusted cosine similarity

1 Introduction

One of the main aims of businesses is to increase sales and revenue generated yearly. For online platforms, the added objective is to ensure personalized experience each time a user visits a platform. Implementations of personalized experiences for users in online solutions in the past year has led to a boost in revenue by 6%—10% (Abraham et al. 2017). This personalized experience comes from having some sort of data regarding user habits such as users' likes and ratings on items on these platforms. It is for this reason that platforms like Netflix, Facebook, Amazon and many more, have implemented recommendation algorithms which use the scores of identical users on a certain item to provide suggestions/recommendations of other items to similar users when next they visit that platform. According to Eckhardt (2018), the combined effect of personalization and recommendation, have saved Netflix over $1 billion each year and over 75% of content users were influenced through content recommendation.

Over the years, several types of recommender's systems have emerged in a bid to predict accurate user interests and give them the best possible experience. Recommendation system consists of the following types: Content-based filtering and collaborative filtering. In content-based filtering, the system recommends similar items to a user based on past likes or purchases of that particular user while collaborative filtering which aims at addressing some of the drawbacks of content-based filtering, uses the concept of similar liked items by similar users to recommend items. The next recommender system that has been proposed recently is the hybrid

✉ Chigozirim Ajaegbu
ajaegbuc@babcock.edu.ng

[1] School of Computing and Engineering Sciences, Babcock University, Ilishan-Remo 121103, Ogun State, Nigeria

recommender system and this is the combination of the two aforementioned recommendation systems (Houtao 2019).

Collaborative filtering engines consist of: user-based and item-based filtering as its types. User-based collaborative filtering engine follows the "people like you" logic which recommends to a user an item that similar users liked before. The problem that arises most times is due to the severity of inadequate data, which could be as a result of an item not being rated by two users. This would leave little records or no records to measure similarity between users. In order to overcome this issue, came the idea of Item-based collaborative filtering algorithm. This functions in such that if a greater number of users score most items alike, the intended user will also follow suit in grading the items. Item-based follows the logic "if you like this, you might also like that also". Thus, it recommends items that are similar to the ones you previously liked.

Studies have shown several efforts by researchers in the past to develop algorithms that should ensure accuracy in recommendation, and such algorithms include the following but not limited to these: collaborative filtering based on clustering, the probability of collaborative filtering algorithms, collaborative filtering based on neural networks, collaborative filtering, matrix decomposition based on a variety of models such as the probability model, Bayes model abstraction, maximum entropy model, Gibbs abstract, linear regression and also shine. Despite above proposed optimised algorithms in collaborative filtering engines for most recommender systems, there still exists problems of sparsity, early rater and cold start just to mention a few.

Cold-start problem, derived from cars when their engines are cold and are having problems starting up until it reaches a particular temperature. In recommendation engines, cold start refers to the condition when the recommendation system is not yet optimal to generate the best results because of data sparsity i.e. problems in finding an ample number of similar users since in general the active users only scored a small fraction of items.

Although, studies like Ahn (2007) presented an algorithm to improve on the similarity measure for collaborative filtering in terms of cold-start situation, its algorithm had a drawback in the sense that its effectiveness when compared with the traditional measure metrics alone lies more in the direction of cold-start problems hence there was no clear cut improvement of Proximity-Impact-Popularity beyond the normal general performance of existing traditional measurement metrics.

However, the author's proposed method is slightly a different approach in the sense that, the author's proposed algorithm is meant to complement the strength of the three traditional measurement metrics in the direction of mitigating the cold-start issue while also retaining other benefits of the three traditional measurement metrics as mentioned above.

To further enhance the accuracy of the recommendations, this study proposes an optimized item based collaborative recommendation algorithm. To obtain high quality recommendation, the study has narrowed it down to the accuracy of inferring rating scores for prospective users. Items similar to those that users rated are sought for, along with the anticipated score. To calculate the similarity between items, the ratio of users that two items i and j to the ratio of users that rated i or j or gotten. The results gotten from our experimental analysis shows that the quality of the proffered algorithm improves the quality of the recommendation engine than the traditional algorithms.

2 Literature review

As mentioned earlier, there are a lot of proposed algorithms in the direction recommendation systems for online stores. However, researchers have found a way of grouping them into content-based and collaborative filtering. (Nikolakopoulos and Karypi 2019), proposed a novel random walk-based method known as Rec Walk in order to boost item-based collaborative filtering. Their method used the spectral properties of nearly uncoupled markov chains to address the challenge of restricting the range of K-step distributions which can be exploited for personalized recommendations.

Thakkar et al. (2019), proposed a method of combining predictions of two recommendation methods: combining user-based collaborative filtering (UbCF) and Item-based collaborative filtering (IbcF) through the use of multiple linear regression (MLR) with the aim of reducing prediction error. Their study reported superiority of approach when compared with the result of other fusion approaches.

Xue et al. (2019), argued that researcher's interest in Item-based Collaborative Filtering (ICF), have centred more in linear and shallow relationships between items, and so, are not sufficient to capture decision-making process of users. Their study therefore, proposed a more expressive ICF solution, by accounting for the nonlinear and higher-order relationships among items. The researchers concluded by their findings that the performance of the proposed method could be further improved by factoring a more fine-grain second-order interaction modeling with attention network.

Schelter et al. (2019), proposed an efficient incremental algorithm for item-based collaborative filtering based on co occurrence analysis. Their study showed an order of magnitude faster than existing open source recommender libraries on many datasets and also scales to high dimensional datasets which they claimed existing recommenders failed to process.

Lu and Xia (2019), applies the Item-based collaborative filtering Algorithm to MOOC recommendation system with intentions of preventing possible defects of the algorithm in terms of low performance and offline processing while paying attention to its good real-time processing and high recommendation efficiency.

Ninan and Rajan (2019), developed a framework for efficient key-word-Aware representative travel route that uses extraction of keywords from historical records and social interactions of users. Furthermore, their study was able to recommend places based on the likeness between places, calculated using ratings of places given by users with the application of Item-based collaborative filtering algorithm.

Barman et al. (2019), proposed a new item-item based similarity metric and a little improvement in prediction method that could compute ratings effectively in order to provide more accurate recommendation. Their technique addressed the cold start problem by employing a technique that calculates the similarity based on item genre.

Latitha (2019), proposed an item-based collaborative filtering algorithm for social networking sites. Their study looked at different ideas pertaining to item-based collaborative filtering as at the time of the research. It was concluded that their proposed item-based collaborative filtering algorithm is utilized to discover the similarity along with forecasting of models.

Jiang et al. (2019), proposed a slope one algorithm which is based on fusion of trusted data and user similarity with the ability of being deployed in various recommender systems. Their result showed a better recommender algorithm than the traditional slope one algorithm.

Hasan and Roy (2019), considered the inability of existing item-based collaborative techniques to predict accurately recommendations for cold-start items which has affected the performance of the recommender system. Hence, their study proposed a two-item-based similarity measure in order to overcome the identified problem by incorporating items' genre data. Their result showed better performance when compared with the traditional techniques.

Basiri et al. (2010) opined that we can improve performance of a new hybrid recommender system which has no rating by applying the high growing type ordered weighted average operator to integrate the output of the recommender system design.

Huang and Yin (2010), proposed a novel efficient association clusters filtering algorithm in order to solve the cold start problem associated with traditional research of recommendations systems, where the algorithm establishes clusters models based on the ratings matrix. Their method was reported to enlarge the prediction scope and improve the accuracy of recommendations.

Gao and Huang (2015) proposed an algorithm with time modification built on a reference to the center of gravity model which uses already rated items table to compute each items reference weight and used that reference center of gravity to compute the initial similarity then making adjustments to the synthesized similarity of the initial similarity and common similarity to get the final similarity.

Wei et al. (2012) opined that to solve problems of data sparsity and poor prediction quality, collaborative filtering algorithms could be optimised using item clustering and global similarity. To their study applied K-MEANS to cluster items based on user rating on items and the local user similarity. Their study showed that in order to improve the selection of target users, neighbours and to achieve better prediction, a new global similarity between users was obtained.

Tyagi and Bharadwaj (2013) proposed the use of multi-objective particle swarm optimal algorithms (MOPSO) for collaborative filtering engines. The MOPSO algorithm has an edge since it extracts appropriate and high-ranking association order in the wilder sense such that no order is higher to them when both aims are concurrently considered. Computational efficiency is improved as a result of mining rules which are only for the giver user and over the associated transactional database.

Qing (2014) opined that to improve collaborative filtering algorithms, the similarity of users is first calculated by introducing similar vectors which shows that two users are vectors and fall under the same interest vector. By using a user rating matrix, the product with the top rating is shown to the prospective users.

Quan (2013) introduced user personality as a way to improve user model thereby leading to identical based collaborative filtering recommendations to compute user similarity from the user personality perspective and then select the nearest neighbour which leads to the generation of a recommendation. The other is centered on the identity item rating matrix which makes recommendations to the intended user.

In summary of the revealed literature, it is evident that this study's proposed algorithm is the very first idea that has focused on using the already existing traditional similarity measurement metrics, to demonstrate clearly how they could be improved further by including a balancing factor in order to enable them deal with extreme data sparsity and cold-start issues in measuring rating similarities among users.

3 Proposed solution

The innovation strategy being followed is innovation by numeric and the proposed solution towards optimizing the collaborative filtering engine, is to improve on the current similarity measurements that currently exist. This will aim at improving the similarity algorithm which should result in improvement in the entire recommendation algorithm as a whole. The three traditional measurement metrics that are considered in this study are:

Cosine-based similarity
Pearson correlation similarity, and
Adjusted cosine similarity.

Cosine-based similarity: Ratings for items are represented in vector forms. Items which users have not rated is set as 0. To obtain the similarity between two items, we find the cosine of the angles between both vectors. In m dimensional space, the rating of the *ith* and *jth* item can be written as: $\bar{i}, \bar{j}$ (Xia, E-commerce Product Recommendation Method based on Collaborative Filtering Technology, 2016). To compute the similarity of both items, it is given as:

$$sim(u,v) = \cos(u,v) = \frac{\sum_{i\in I_{uv}} R_{u,i}R_{v,i}}{\sqrt{\sum_{i\in I_{uv}} R_{u,i}^2}\sqrt{\sum_{i\in I_{uv}} R_{v,i}^2}} \tag{1}$$

Pearson Correlation-based similarity: The approach here is to find situations where users scored both items i or j and then obtain the users that scored both items i and j. This is used in cases when the items are linear (Xia 2016). The correlation can then be represented as:

$$sim(u,v) = \frac{\sum_{i\in I_{uv}} \left(R_{u,i} - \bar{R}_u\right)\left(R_{v,i} - \bar{R}_v\right)}{\sqrt{\sum_{i\in I_{uv}} \left(R_{u,i} - \bar{R}_u\right)^2}\sqrt{\sum_{i\in I_{uv}} \left(R_{v,i} - \bar{R}_v\right)^2}} \tag{2}$$

Adjusted Cosine Similarity: Users rating on some items might not be uniform. Some users might rate some items higher than the others thereby leading to a variation between the score given of the same item (Xia 2016). The Adjusted Cosine Similarity considers the difference of user rating grading and computes it as follows:

$$sim(u,v) = \frac{\sum_{i\in I_{uv}} \left(R_{u,i} - \bar{R}_u\right)\left(R_{v,i} - \bar{R}_v\right)}{\sqrt{\sum_{i\in I_u} \left(R_{u,i} - \bar{R}_u\right)^2}\sqrt{\sum_{i\in I_v} \left(R_{v,i} - \bar{R}_v\right)^2}} \tag{3}$$

4 Similarity analysis

Due to the expansive growth of systems that deploy recommendation engines, the number of users and items are increasing and because of the size of these items, very few get rated. In most cases, regular users usually score less than 1 percent of the entire items. Table 1, contains a rating matrix example to show how loosely rated items are in a matrix:

In Table 1, three items are presented with the various ratings from the five users accordingly and the author has used the Pearson Correlation based formula to demonstrate the behaviour of the traditional measure metrics in computing the similarity between items1 & items2 and item1 & item3 and the result is as follows:

Let U represent Item1 and V represent Item2.

Using the Pearson Correlation-based similarity

$$sim(u,v) = \frac{\sum_{i\in I_{uv}} \left(R_{u,i} - \bar{R}_u\right)\left(R_{v,i} - \bar{R}_v\right)}{\sqrt{\sum_{i\in I_{uv}} \left(R_{u,i} - \bar{R}_u\right)^2}\sqrt{\sum_{i\in I_{uv}} \left(R_{v,i} - \bar{R}_v\right)^2}}$$

for sim(Item1, Item2) result details & calculation

U Values
$\sum = 3$
Mean = 0.6
$\sum(U - M_u)^2 = SS_u = 7.2$

V Values
$\sum = 3$
Mean = 0.6
$\sum(V - M_v)^2 = SS_v = 7.2$

U and V Combined
$N = 5$
$\sum(U - M_u)(V - M_v) = 7.2$

Similarity Calculation
$r = \sum((U - M_v)(V - M_u)) / \sqrt{((SS_u)(SS_v))}$

$r = 7.2 / \sqrt{((7.2)(7.2))} = 1$

this implies that **sim(Item1 and Item2) = 1**

And shows a strong correlation meaning that high U variable scores go with high V variable scores and vice versa.

for sim(Item1, Item3) result details & calculation

U Values
$\sum = 10$
Mean = 2
$\sum(U - M_u)^2 = SS_u = 18$

V Values
$\sum = 11$
Mean = 2.2
$\sum(V - M_v)^2 = SS_v = 18.8$

U and V Combined
$N = 5$
$\sum(U - M_u)(V - M_v) = 18$

Similarity Calculation
$r = \sum((U - M_v)(V - M_u)) / \sqrt{((SS_u)(SS_v))}$

$r = 18 / \sqrt{((18)(18.8))} = 0.9785$

This implies that **sim(Item1 and Item3) = 0.9785** and also shows a strong correlation

From the above results, it could be seen that the similarity between Item1 and Item2 is larger than that of Item1 and Item 3 and this is as a result of smaller number of users that rated the items as compared to the ratings on Item2 and Item2. However, this method of conclusion also showed that the similarity of an item is 1 when users' ratings among two items, gave equal scores to respective items. Hence, this would result in a conclusion which is unacceptable because such scenarios would end up with inaccurate recommendations in cases of sparse rating. In order to optimize the similarity computation, this study proposed a balancing factor that should be added to the similarity formulas earlier presented.

Table 1 Users rating matrix

	Item1	Item2	Item3
User1	2	NA	3
User2	5	NA	5
User3	3	3	3
User4	NA	5	NA
User5	NA	2	3

5 The balancing factor is been established using the following methods:

Assume "i" and "j" to represent items
Then take a count for the number of users that have scored the same items and store them in a variable N
Take a count of the number of users that have rated either items i or j and store in a variable M.
The ratio of co-rated items N to the total items M is taken and apply a weight similarity function for each case f (N/M).
The similarity then becomes sim(i,j) = f(N/M) sim(i,j)
Write f(N/P) as 1-α(1-N/M).
Assume α to be values between [0,1]
Then optimise the similarity computation by applying the weighted factor. When α is 0, it gives the same similarity values with the initially calculated similarity values above.

Hence, the following equations presents the author's balancing factors as incorporated in the Pearson Correlation based similarity measurement metrics:

$$sim'(i,j) = \left(1 - \alpha + \alpha\frac{N}{M}\right)\frac{\vec{ij}}{\|\vec{i}\| * \|\vec{j}\|} \tag{4}$$

$$sim'(i,j) = \left[1 - \alpha(1 - \frac{N}{M}\right]\frac{\sum_{u\in U_{ij}}\left(R_{u,i} - \bar{R}_i\right)\left(R_{u,j} - \bar{R}_j\right)}{\sqrt{\sum_{u\in U_{ij}}\left(R_{u,i} - \bar{R}_i\right)^2}\sqrt{\sum_{u\in U_{ij}}\left(R_{u,j} - \bar{R}_j\right)^2}} \tag{5}$$

$$sim'(i,j) = \left[1 - \alpha(1 - \frac{N}{M}\right]\frac{\sum_{u\in U_{ij}}\left(R_{u,i} - \bar{R}_i\right)\left(R_{u,j} - \bar{R}_u\right)}{\sqrt{\sum_{u\in U_{ij}}\left(R_{u,i} - \bar{R}_i\right)^2}\sqrt{\sum_{u\in U_{ij}}\left(R_{u,j} - \bar{R}_u\right)^2}} \tag{6}$$

When α is 1, the similarity would be different, and thus would result to:

For the balancing factor, applying values accordingly in the equation for sim(Item1,Item2):

$[1 - \alpha(1 - N/M)] = 0.2$

Hence, multiplying the author's balancing factor with the Pearson Correlation based similarity measurement results to:

sim(item1, item2) = 0.2

For the balancing factor, applying values accordingly in the equation for sim(Item1,Item3):

$[1 - \alpha(1 - N/M)] = 0.75$.

Hence, multiplying the author's balancing factor with the Pearson Correlation based similarity measurement results to:

sim(item1, item3) = 0.73.

6 Experimental results

To test the effectiveness of the proposed algorithm, the author used a data set from the Movielens recommender system (http://movielens.umn.edu/) developed by GroupLens research team at the Minnesota University. In order to evaluate the efficiency of the author's balancing equation, the m1 dataset was used, which has 100,000 ratings ranging from 1 to 5.

This dataset has 1682 movies, and 943 users where each user has rated over 15 movies. The dataset was divided into groups, one for training and the other for testing. 30,000 of the ratings were used for testing while 70,000 were used for training.

The sparse degree of the dataset was calculated by finding the percentage of no ratings in the entire rating matrix. For this data set the sparse level is:

1—10,000/943 × 1682 = 0.937.

In order to further measure the quality of the proposed solution, the author adopted a statistical metric which is the Mean Absolute Error (MAE) which is used to calculate predicted rating against the actual users' ratings.

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \quad (7)$$

where {p1, p2,p3,p4,……pN) represents the anticipated ratings and {q1,q2,q3,q4,…..qN} represent the actual usual ratings.

The three similarity measures mentioned above were tested with different values of α on the same data set and It showed that at $\alpha = 0.1$, the optimized algorithm gave the same output as the original.

The outcomes while using MAE to determine the quality of the proposed system, are shown in Figs. 1–3.

Figure 1 presents the performance of various values of α as where to be used in the proposed improved cosine-based similarity. It shows that at $\alpha = 0$, the output still behaved same as were without the balancing factor while at $\alpha = 1$, offered a better similarity measurement performance.

In Fig. 2, although better result was obtained also with Adjusted Cosine similarity, when $\alpha = 1$. The Cosine-based similarity performed much better than the Adjusted Cosine

Fig. 1 Comparison of Cosine similarity of different α

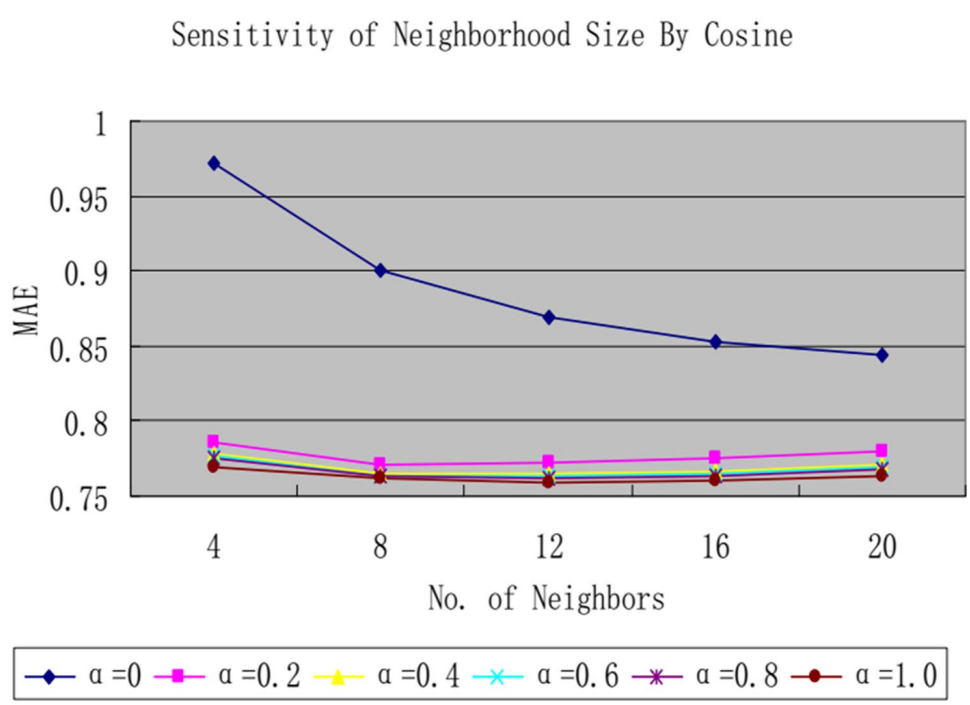
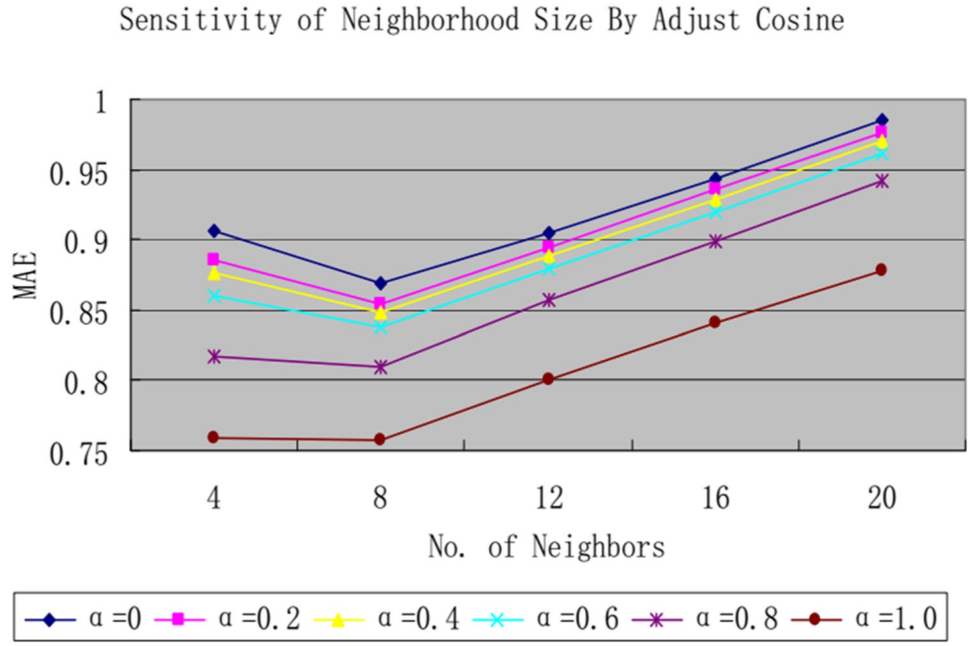


Fig. 2 Comparison of Adjust Cosine similarity of different α

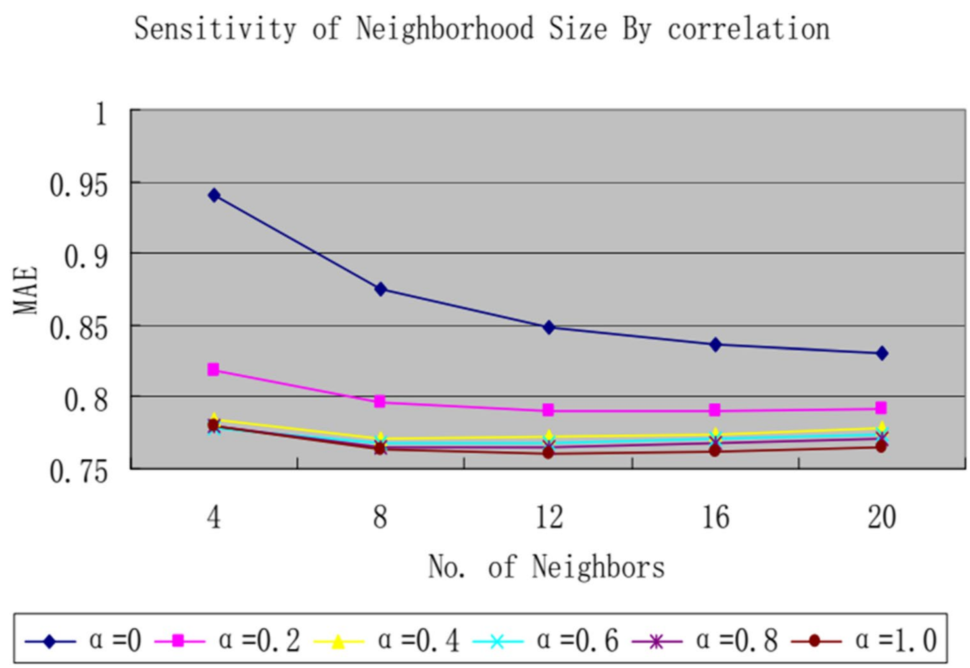


similarity with the proposed balancing algorithm combined to their traditional measurement metric.

In Fig. 3, it shows that the Pearson-based correlation similarity measure performed best of the earlier two traditional similarity measurement metrics.

Hence, the figures above, showed that the MAE and α are inversely proportional irrespective of which similarity metrics we apply. The graph shows that the value of the MAE is least when we use α as 1 which is evidence that the author's proposed algorithm performed better than the traditional algorithm, in the sense that it added more value in the direction of tackling data sparsity or cold-start issues in the existing algorithm.

Fig. 3 Comparison of Correlation Similarity of different α



7 Conclusion

In this paper, the author looked at item-based collaborative filtering and the cold-start problem associated with obtaining the similarity evaluation with the traditional methods (Cosine-based Similarity, Pearson Correlation-based Similarity and Adjusted Cosine Similarity). The study proposed an optimized algorithm to tackle this problem by adding a balancing factor to the similarity calculation methods and varying the values of alpha. From the experiments carried out and the experimental results obtained, it shows that the improved item-based collaborative filtering algorithm added more value to the traditional algorithm by improving its abilities in tackling the extreme data sparsity and cold start issues leading to a better recommendation system.

References

Abraham M, Mitchelmore S, Collin S, Maness J, Kistulinec M, Khodabandeh S,Visser J (2017) Profiting from Personalization. https://www.bcg.com/publications/2017/retail-marketing-sales-profiting-personalization.aspx

Ahn H.J (2007). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences. ELSEVIER

Basiri J, Shakery A, Moshiri B, Hayat MZ (2010) Alleviating the cold-start problem of recommender system using a new hybrid approach. In: 5th international symposium on telecommunication.

Eckhardt S (2018) Promise and Peril for Machine Learning at Netflix. https://digital.hbs.edu/platform-rctom/submission/promise-and-peril-for-machine-learning-at-netflix/

Gao L, Huang M (2015) A Collaborative filtering recommendation algorithm with time adjusting based on atrribute center of gravity model. Web information system and application conference.

Hasan M, Roy F (2019) An Item-item collaborative filtering recommender system using trust and genre to address the cold-start problem. Big Data CognitComput. https://doi.org/10.3390/bdcc3030039

Houtao D (2019) How companies make product recommendations. Recommender System in Practice. https://towardsdatascience.com/recommender-systems-in-practice-cef9033bb23a

Huang C, Yin J (2010) Effective association clustering filtering to cold start recommendations. In: International conference on fuzzy systems and knowledge discovery.

Jiang L, Cheng Y, Yang L, Li J, Yan H, Wang X (2019) A Trust-based collaborative filtering algorithm for E-commerce recommendation system. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-018-0928-7

Latitha K (2019) Recommend suitable jobs in career-oriented social networking sites using item-based collaborative filtering algorithm. Int J Res Sci Eng Technol 6(10). http://www.ijrset.in/index.php/ijrset/article/view/409

Liu G, Wu X (2011) Collaborative filtering algorithm combined with DOC2VEC.

Lu Q, Xia J (2019). Research on the application of item-based collaborative filtering algorithm in MOOC. J Phys 1302. https://doi.org/10.1088/1742-6596/1302/3/032020.

Nikolakopoulos AN, Karypis G (2019) Boosting Item-based collaborative filtering via nearly uncoupled random walks. ACM Trans Knowl Discov. Data. https://arxiv.org/pdf/1909.03579.pdf

Ninan AM, Rajan JE (2019) An item based collaborative filtering on recommendation of travel route. Int Res J Eng Technol 6(5). www.irjet.net

Qing YX (2014) An intelligent E-commerce recommendation algorithm based on Collaborative filtering technology. In: International conference on intelligent computation technology and automation.

Quan Z (2013) Collaborative filtering recommendation based on user personality. In: International conference on information management, innovation management and industrial engineering.

Schelter S, Celebi U and Dunning T (2019). Efficient incremental cooccurrence analysis for item-based collaborative filtering. In: SSDBM '19: international conference on scientific and statistical database management, July 23–25, 2019, Santa Cruz, CA. ACM, New York, NY, USA, 12 pages. https://dl.acm.org/doi/pdf/https://doi.org/10.1145/3335783.3335784

Thakkar P, Varma K, Ukani V, Mankad S and Tanwar S (2019). Combining user-based and item-based collaborative filtering using machine learning. Inform Commun Technol Intell Syst pp 173–180 https://link.springer.com/chapter/https://doi.org/10.1007/978-981-13-1747-7_17

Tyagi S, Bharadwaj K (2013) Enhancing collaborative filtering recommendations by utilizing multi-objective particle swarm optimization embedded association rule mining. Swarm and Evolutionary Computation.

Wei S, Ye N, Zhang S, Huang X, Zhu J (2012) Collaborative filtering recommendation algorithm based on item clustering and global similarity. In: International conference on business intelligence and financial engineering.

Xia J (2016) E-commerce product recommendation method based on collaborative filtering technology. In: International conference on smart grid and electrical automation.

Xue F, He X, Wang X, Xu J, Liu K and Hong R (2019). Deep item-based collaborative filtering for top-N recommendation. ACM Trans Inform Syst 37(3), Article 33. https://doi.org/10.1145/3314578