xIlkogretim Online - Elementary Education Online,2021; Vol 20 (Issue 5): pp.2867-2874
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.05.313

Expert System for Recommendations of Healthy Food Recipes using machine learning

Dr. Naveed Jhamat, Assistant Professor, Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala, 52250, Pakistan, Naveed.jhamat@pugc.edu.pk
Dr. Ghulam Mustafa*, Assistant Professor, Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala, 52250, Pakistan, gmustafa@pugc.edu.pk
Zeeshan Arshad, Lecturer, Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala, 52250, Pakistan, Zeeshan.arshad@pugc.edu.pk
Rizwan Abbas, Lecturer, Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala, 52250, Pakistan, Rizwan.abbas@pugc.edu.pk

ABSTRACT- Making customized recommendations has been a key feature of many websites, and it's just getting bigger as more people have access to vast volumes of data on the internet. When done correctly, giving recommendations based on the individual's preferences rather than trendy products improves consumer satisfaction and will eventually draw more buyers. One more challenge is to deliver healthy recommendations that represent the needs of consumers and preferences and information on the wellbeing of the consumers. This paper proposes a method that categorizes recipes by cuisine and calories that will help to improve rating predictions. We built the proposed Hybrid Hierarchical Clusters Recommender System (HHCRS) with cuisine and calories information and compared its performance with baseline and other minimal recommender systems. The findings showed that our approach can substantially boost prediction and can help to minimize the sparsity of the rating matrix.

Keywords: Recipe Recommender System, Cuisines, Calories, Clustering, Artificial Intelligence

I. INTRODUCTION

Recommendation schemes are a natural line of defence against customer over-choice. They can be seen on a wide range of blogs, as well as e-commerce and media websites. Recommendation models are classified into three groups basedon the types of input data: collaborative filtering (CF), content-based recommender systems (CB), and hybrid recommender systems. (Burke,2002)(Zhang et al., 2019)(Toledo, 2019). Machine learning algorithms are being used by sites like Netflix and Amazon to determine what things a person would want to watch. To some point, genetics also plays a part in this. People, for example, prefer the taste of sugar over bitterness, but in the era of social media, bitter tastes can overshadow initial preferences. This helps to understand that taste is such a complicated phenomenon (Sean, 2016).

In different implementations, recommendation mechanisms have attempted to provide consumers with reliable recommendations to suit their needs. In recommendation systems, collaborative filtering is a well-known and successful technology. Collaborative filtering technology has been used by many websites to personalize the surfing experience for each customer. Through using collaborative filtering mechanisms to boost revenue and rental prices, Amazon increased sales by 29 percent (Mangalindan, 20120), Netflix increased video rentals by 60 percent (Koren, 2009), and Google News increased click-through rates by 30.9 percent (Liu et al.,2010). The author believes that collaborative filtering can be used to boost company profit margins and service in several ways. He believes that using this technology will improve the digital world's revenue, rentals, and customer service costs. He concludes that this advice approach is more efficient than conventional customer service approaches and is a means of improved customer service (Su et al., 2009).

Related users and similar items are searched for collaborative filtering (CF). Traditional CF examines local consequences and is a strong and common approach for items and consumers. CF continues to have a bias in popularity and rare items may be overrun with familiar ones. Hybrid methods can be used to address CF flaws like sparsity and cold start. CB approaches examine item attributes in a catalog to identify specific products to be assessed by a recipient. This can be a blend of CF and CB approaches to locate the things that a person is probably interested in (Jiang et al.,2019)(Melville et al.,2002)(De Campos)

The cutting edge for Recommending programs generally uses mostly some hybrid approaches between CF and CB methods such as factoring a rating matrix, which is very effective and is used by many sites, including Netflix. Different hybrid recommendation systems have been used to recommend food and recipes, provided various requirements. For example, (Syensson et al.,2005) have created a social navigation system where users select based on their recipes to anticipate what users want.

A Content-based (CB) approach considers the cuisine and calories in a formula. Rather than describing a recipe as a vector of attributes, this paper explores whether high-level representations of recipes (see Fig. 1) can offer stronger recommendations than standard CF and Matrix Factorization (MF) approaches. The research issue is whether grouping recipes and classifying them according to their calories and cuisine would improve prediction accuracy when CF and MF are also used. The paper continues by examining how a mixture of the CB and MF approaches would increase prediction accuracy. This article discusses the various approaches to recommender schemes, as well as how they've been combined and applied.

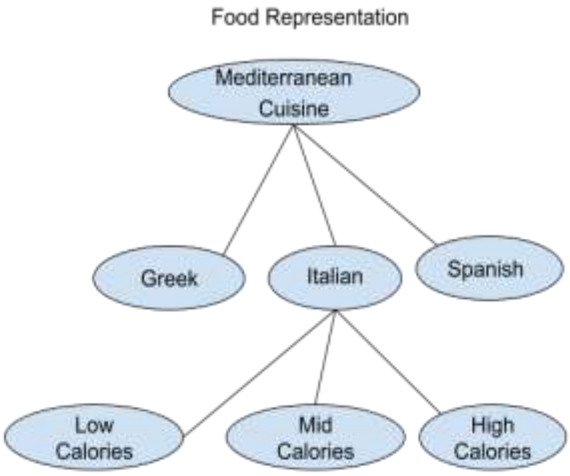


Figure 1: A glimpse of the suggested hierarchical tree structure for high-level recipe representation. The recipes are first grouped in the Cuisine, and then the Calories.

The remainder of the paper is structured as follows: Section II introduces related work. We define our Hybrid Hierarchical Clusters Recommender System (HHCRS)in Section III. Experiments that demonstrate the feasibility of our strategy are described in Section IV.The observations are documented in Section V and the findings are explored in Section VI. Finally, in Section VII, a conclusion is drawn, which involves a brief discussion of potential future work and improvements.

II. RELATED WORK

A neighbourhood algorithm is minimal CF solution, which forecasts ratings on the basis of what related items are to one another or if other users have ratings. The Pearson correlation coefficient is a common way to measure similarities. This strategy alone doesn't, though, conceive of some consumers being tougher than others and others tending to offer less scores (Schafer et al.,2007) (Bbadilla et al.,2020)

Collaborative filtering (CF) differs from content-based filtering (CB) in many ways (Wang et al., 2002). CF recommends items that have been favourably rated by consumers with similar preferences in the past, rather than making recommendations based on customer and item similarity. The primary benefit of CF is that it does not require any prior knowledge of the items in order to make recommendations. Recommendations are solely dependent on user comparisons, allowing for the recommendation of extremely complicated objects.The lack of detail, on the other hand, may be a drawback. About the fact that Expedia and TripAdvisor are very similar items, the device will not group those items together.

Both the item and the user profile are represented by a collection of attributes in content-based filtering (CBF). Items are ranked according to how closely they match the user's attribute profile. Because of its simplicity, this form of RS is very common. It can be used to recommend blogs, music, movies, books, restaurants, and hotels, among other things. The following are the steps to take when developing a content-based framework. Recommendations that are solely dependent on the user's profile risk being overspecialized.For example, if a consumer enjoys multiple films from the same genre, the algorithm would tend toward recommending only the remaining films in that genre. Extracting an item's attributes and obtaining all of its various facets can be a difficult process (Van et al.,2011)(Tran et al.,2018).

Singular Value Decomposition (SVD)method decomposes a matrix into three matrices by dividing it into sub-matrices. The best low-rank approximation of the ranking matrix R is SVD, but it has two flaws: overfitting and the fact that all entries in matrix arerequired because the objective function needs sufficient scores for all entries. To avoid overfitting, reduce the number of features and this can also be overcome by using a regularisation in the objective function (Zhou et al., 2008). Gradient descent (GD) can be used to find the matrices P and Q with respect to objective function, and it does not require that all entries in the ranking matrix be available.Due to sparsity, this can be more feasible for a more practical solution in the form of recommender systems, and various versions of the sparsity process have proved to be effective (Kabbur et al., 2013). The approach can be used to strike a balance between the possibility of system accuracy and complexity, as well as overfitting due to model overfitting, and selecting the right number of feature options for the best performance.

In order to provide diverse and reliable recommendations, several recommender programmes merge CF and CB. Melville et al. (2002) suggested a content-boosted collaborative filtering (CBCF) approach, in which a CF method is used to minimise sparsity in the ranking matrix, followed by a CB method to fill in the gaps. Using Pearson Correlation Coefficient Coefficient(1) and Freyne et al. (2010) produced recommendations based on a similarity measure between different items. The process performed 9.2 percent better than a pure CB method and 4% better than a pure CF method. In the form of recipe recommendation systems (Li et al.,2016).

Minimal CF is capable of resolving all CB-related issues. Combining collaboration and content-based approaches, for example, is very common because they complement each other well. A framework like this will make recommendations based on the profile of a single user as well as groups of users with identical ranking histories. The framework will also switch between collaborative and content-based recommending in specific situations. Where the RS is unable to remove distinguishing features from items, collaborative filtering is used. Content interpretation may also be used by the system to address CF-specific issues. It may also switch between different approaches to create some synergy. Weighted, Mixed, Meta-level, Cascade, Feature augmentation, and Feature mixture are the seven forms of hybridization methods defined by Burke (2002).

Different hybrid recommending systems are used for recommendation systems in the form of food and recipes provided different requirements. For example, in order to determine what people choose to do, Svensson et al. (2005) has created a social browsing method that uses the recipes used, while van Pinxteren et al. (2011) has derived and assessed a correlation measure to suggest healthier recipes. Additional recommendation mechanisms examined use feature-based CB methods in which recipes are shown as vectors with ingredients, for instance, as entries (Freyne et al., 2010) (Jagithyala, 2014). Instead, Hanai et al. (2015) have successfully clustered recipes based on names and ingredientsRecipes as vectors with ingredients are described as inputs. Many programmes consider using approaches dependent on ingredients. The processing and collection of any ingredient and function of an item will prove computationally costly. This paper explores whether the high-quality recurrence proposed in the recipes can offer stronger recommendations than standard CF and MF might provide, rather than presenting a recipe as a vector of such features as ingredients. The research issue is therefore whether it would help to improve predictions by grouping recipes and classifying them according to their key calories and cuisines, if CF and MF are both included.

III. METHOD

The Baselinemodel has three components; the average ranking in the catalogue, the average consumer rating and the average item rating. By linearly integrating these elements, where b_i is the bias of the item i and b_u the bias of the user u respectively, Baseline model provides calculations based on how the average ranking of users and items deviates from the total average rating.

$b_{\mu,i} = \mu + b_i + b_u = \mu + (\mu_i - \mu) + (\mu_\mu - \mu)$ ………………………….. (1)

The suggested solution, as seen in Figure 1, first classifies which cuisine section a dish belongs to. The theory is that this knowledge is sufficient for clustering recipes so users may express a simple preference for a certain set of recipes. If the rating range is narrow, a meaningful average of the rating scan can be calculated as a prediction for other recipes in the same cluster.If there is no consistent concentration of ratings (trend) in that cluster (high rating variance), other clusters from the same cuisine population with the same class ofcalories (neighbourhood clusters) are investigated. The classification of various cuisines may be based on a qualitative or subjective calculation of resemblance.

Algorithm 1 formalises the proposed Hybrid Hierarchical Clusters Recommender System (HHCRS) in this article, having two phases. The SVD gradient is used in phase II. Since the CB approach can forecast more missing entries, it can be used to estimate missing entries in phase II.The pseudo-code for the proposed functional implementation of the HHCRS high-level hierarchical solution is seen in Algorithm 1. The vector *predicted_rating* stores the predicted rating of the item of interest for the consumer of interest. This is determined by determining whether the statistical value for all items scored by the user of interest in the cluster $(C_{i=I,j=J,k=K})$ is less than a certain threshold and whether the number of rated items in a cluster is greater than a certain threshold.

Thresholds can be set at arbitrarily. If the rating range is narrow and enough items in the neighbourhood cluster have been scored, an average of all the ratings within that cluster is taken and stored in predicted rating. Line 5-10 collects scores from all neighbourhood clusters if this is the case. If at least one of the restrictions in line 11 is not met, it indicates either a dispute between the neighbourhood clusters (high rating variance) or a lack of sufficient knowledge.Phase II of HHCRS illustrates how to use CB,CF, Baseline and SVD to estimate missed entries in the ranking matrix.

In Algorithm 1, line 2, 8, and 11, how to pick thresholds is a trade-off between tighter constraints and more personalizedpredictions. Stricter criteria would almost certainly result in more reliable and fair predictions, but they will limit the number of things that can be predicted. Line 3 in both Phase II can be skipped to reject the proposed HHCRSsolution.

Step	**Algorithm 1:**Hybrid Hierarchical Clusters Recommender System
	Phase I
1	predicted_rating ={}
2	**If** var{ratings($C_{i=I,j=J,k=K}$)}<$T_{ov}$and size ($C_{i=I,j=J,k=K}$) >T_{os} then
3	predicted_rating =mean{C_{IJK}}
4	**else**
5	RatingClusters=[]
6	N=0
7	**foreach** ($C_{i=I,j=J,k=K}$) ∉ {}**do**
8	**If** var{$C_{i=I,j=J,k=K}$}<T_{ov} and size ($C_{i=I,j=J,k=K}$) >T_{os}**then**
9	Ratingclusters.add(mean($C_{i=I,j=J,k=K}$))
10	N=N+1
11	**If** var{ratingclusters}<T_v and N>T_s**then**
12	predicted_rating=mean(RatingClusters)
13	**return** predicted_rating
	Phase II
1	$R_{normalized}$ =*normalize*(R)
2	R=CollaborativeFiltering ($R_{normalized}$)
3	R=ContentBased (R) % from phase 1
4	**foreach** (i,j) ∈to R**do**
5	**If** R_{ij} ∈{} **then**
6	R_{ij} = Baseline(R_{ij})
7	**else**
8	R_{ij} = (Baseline(R_{ij})+R_{ij})/2

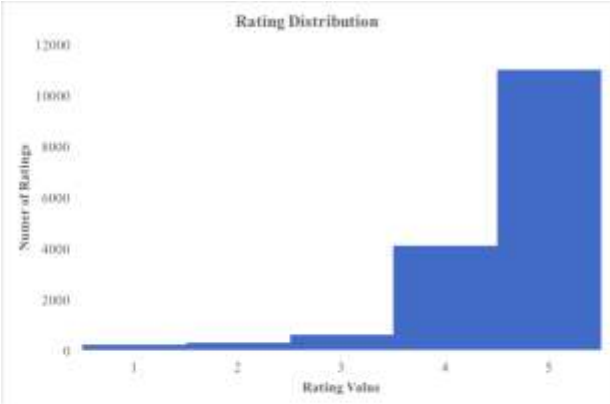
9	R=SVD(R)
10	**return** R

IV. EXPERIMENTAL EVALUATION

DATASET

User ratings for various recipes get from Allrecipes[14], where each rating is a numerical value between one and five. This sparse data set represents the ground reality for the rating matrix, and only 2,000 of the 16,000 entries in the dataset are available. A one-star rating is the least popular granted rating in the dataset. The most common test set (2,000 entries) is a five-starrating, and the remaining 90% of the available data in the dataset is used as a test set. The dataset has a median rating of 5 and an average rating of around 4.5.

Fig: This chart demonstrates the distribution of the dataset scores. With greater scores, the quality of ratings seems to rise exponentially. One-star rankings are the least common and five-star rates are the most common.



EVALUATION METHODOLOGY

The following measures have been used to assess the efficiency of our proposed method and other approaches.The root-mean-square error (RMSE) is calculated to measure the algorithms efficiency.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i,j\in S}(R_{ij} - \widehat{R_{ij}})^2} \quad \text{...} (2)$$

$$Accuracy = \frac{1}{N}\sum_{i,j\in S} I(round(R_{ij}) = round(\widehat{R_{ij}})) \text{..............................} (3)$$

I (x = y) is a predictor vector that is one when x equals y and zero otherwise.For quantifying the perturbations around the obtained mean RMSE and Accuracy values, the *standard error* is used. The t-statistic is used to decide whether one sample differs significantly from another. The p-value is taken as a whole to determine if one sample differs significantly from another. The significance value has been set at 5%.

V. RESULTS

Using just the proposedPhase I, an average ranking of 150.2 out of 2,000 test entries can be predicted.This results in an average RMSE of around 1.10 and an Accuracy of 0.55. Increasing the thresholds even further yields apredicted entry average of 13.2 with 0.90 RMSE and 0.52 Accuracy value. Since the Phase I failed to estimate any scores in certain test sets, the RMSE and Accuracy values are NA. With respect to the selected thresholds, the prediction rate is sensitive (cluster sizes).As can be seen from Table I, the findings cannot be affected by stricter threshold conditions for the rating variance. Using tighter thresholds results in fewer rating predictions and, as a result, untrustworthyRMSE and Accuracy values, since the prediction rate is sensitive to the selected thresholds (cluster sizes).

Table I: The proposed HHCRS(Phase I) approach was tested with a 10-fold cross-validation. The size of the clusters and variance limits in Phase I are the ToS/TS and ToV/TV. Normal errors in the various thresholds are shown on the average predictions, RMSE and Accuracy.

Clusters Size	Variance Limits	Entries predicted	RMSE	Accuracy
0	1.7	150.2±2.8	1.10±0.03	0.55±0.01
1	1.7	13.2±1.2	0.90±0.07	0.52±0.04
2	1.7	2.9±0.70	NA	NA
0	1.0	150±2.8	1.10±0.03	0.55±0.01
1	1.0	13.2±1.2	0.90±0.07	0.52±0.04
2	1.0	2.95±0.70	NA	NA

When CF is used before CB, the sparsity is reduced and the prediction rate is increased. Ratings for 499.1 test entries can be expected on average with CF, with an average RMSE of around 1.02 and an Accuracy of 0.54.

TABLE II: Using 10-fold cross validation, the proposed HHCRSapproach (Phase I) was tested alongside CF. Standard errors are seen for the average number of predictions, RMSE, and Accuracy.

Model	Entries predicted	RMSE	Accuracy
CB	150.2±2.8	1.10±0.03	0.55±0.01
CF	499.1±4.8	1.02±0.02	0.54±0.01
CBCF	545.8±6.2	1.03±0.02	0.54±0.01
CF &Baseline	all	0.91±0.02	0.55±0.00
HHCRS Phase I	all	0.90±0.02	0.55±0.00

When using SVD on a rating matrix, one must have access to all of the matrix's entries. SVD can provide an average RMSE of 0.89 and an Accuracy of 0.54 at most. There is noticeable difference between HHCRS Phase IIand HHCRS Phase I.The confidential interval for SVD with 350 latent factors is also narrower than the CBCF, as seen in Table III.

TABLE III: For various numbers of latent variables, k, the average RMSE and Accuracy values, as well as their normal errors.

Model	K (SVD)	RMSE	Accuracy
HHCRS Phase I	/	0.90±0.02	0.55±0.00
HHCRSPhase II	50	2.719±0.01	0.34±0.01
HHCRSPhase II	150	1.67±0.02	0.45±0.00
HHCRSPhase II	250	1.17±0.02	0.54±0.00
HHCRSPhase II	350	0.89±0.01	0.57±0.00

How to choose Phase I thresholds in line 2, 8, and 11 is a trade-off between tighter limitations and additional predictions. Phase II is the most effective solution to RMSE, while Baseline is the optimal approach to Accuracy. Baseline is the easiest way to forecast the Item of interest ranking.

TABLE IV: The results for the best methods in terms of RMSE and Accuracy are presented in this table.

Model	RMSE	Accuracy
Baseline	0.89 ± 0.01	0.56 ± 0.01
CF & Baseline	0.91±0.02	0.55±0.00
HHCRS Phase I	0.90±0.02	0.55±0.00
HHCRS Phase II	0.90 ± 0.02	0.55 ± 0.01

VI. DISCUSSION

This article is aimed at examining whether the clustering of food recipes according to their cuisine and calories can help increase predictions compared to the use of methods such as CF, MF and Baseline, which are comparatively affordable in a hierarchical system based on a few major descriptors. In terms of Accuracy, Baseline offers the right solution (0.55).When combined with state-of-the-art approaches, the proposed HHCRS solution is able to boost integrated predictions.

The CB approach alone can only predict150.2 out of 2000 entries on average. The number of predictions degrades easily as stricter thresholds on the number of rated items in a cluster are used. The number of predicted ratings that the CB approach will provide seems to be highly sensitive to threshold values. By looking at neighbourhood clusters, the proposed HHCRS solution aims to minimise this weak point to some degree. However, it is obviously insufficient in this case, and sparsity is unavoidable in fact.

It may be proposed that rating objects and then looking at whether a machine learning algorithm will generate recommendations that consumers would actually choose is a more relevant form of training algorithms. Instead of attempting to predict all scores, the best outcomes can be achieved by using a selection of things that a person would be most interested in and training the algorithms to identify those.

It can be argued that the clusters are not well organised or the data is too small. Building or monitoring a social navigation device is a hazardous jump into a future where a person selects a lot of privately-owned data and knowledge about others.In order to know where to draw the line, open conversations between businesses that use machine learning algorithms and customers are needed. It's also interesting to talk about what kind of recommendations to offer for a safe, long-term consumption among users, particularly in the sense of food recipes. As stated in (Van et al., 2011), more research on recipe recommender systems should consider the health aspect. There is a trade-off between collecting more data to improve recommendations and protecting users' privacy. The proposed CB approach is less invasive since it collects less data about user behaviour and only collects data that users directly have (ratings). It is a less invasive tool since it only extracts information that the customer expressly provides.

VII. CONCLUSION

A method that considers cuisines and calories in a recipe is proposed. Instead of defining a recipe as an attractiveness variable, this paper examines the possibility of offering better recommendations than traditional CF and Matrix Factorisation (MF) approaches at a high level representing recipes. The research question was whether it would increase prediction accuracy if CB and CF are also used to group recipes and label them according to their calories and cuisines. Our approach performed better as compared to other recommendations techniques discussed in the paper.

However, since rating matrices are typically sparse in use, the proposed HHCRS approach should be combined with other feature-based algorithms, even though this paper aims to investigate a method that is considered computationally simple. A less sparse dataset may be used to further explore how well the CB method performs. More complex versions of hierarchical clustering can also be tested. Another layer of grouping, such as dish, may be applied in addition to cuisine and calories. If a dish is eaten as a dessert, with pasta, or as a soup, for example.

REFERENCES

1. Bobadilla, J., Ortega, F., Gutiérrez, A., & Alonso, S. (2020). Classification-based Deep Neural Network Architecture for Collaborative Filtering Recommender Systems. *International Journal of Interactive Multimedia & Artificial Intelligence*, *6*(1).
2. Burke, R. (2002). Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction, 12(4), 331-370.
3. De Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. International journal of approximate reasoning, 51(7), 785-799.
4. Freyne, J., & Berkovsky, S. (2010, June). Recommending food: Reasoning on recipes and ingredients. In *International Conference on User Modeling, Adaptation, and Personalization* (pp. 381-386). Springer, Berlin, Heidelberg.

5. Hanai, S., Nanba, H., & Nadamoto, A. (2015, December). Clustering for closely similar recipes to extract spam recipes in user-generated recipe sites. In *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services* (pp. 1-5).
6. Jagithyala, A. (2014). *Recommending recipes based on ingredients and user reviews* (Doctoral dissertation, Kansas State University).
7. Jiang, S., Fang, S. C., An, Q., & Lavery, J. E. (2019). A sub-one quasi-norm-based similarity measure for collaborative filtering in recommender systems. Information Sciences, 487, 142-155.
8. Kabbur, S., Ning, X., & Karypis, G. (2013, August). Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 659-667).
9. Koren, Y. (2009, June). Collaborative filtering with temporal dynamics. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 447-456).
10. Li, Z., Hu, J., Shen, J., & Xu, Y. (2016, July). A scalable recipe recommendation system for mobile application. In *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)* (pp. 91-94). IEEE.
11. Liu, J., Dolan, P., & Pedersen, E. R. (2010, February). Personalized news recommendation based on click behavior. In Proceedings of the 15th international conference on Intelligent user interfaces (pp. 31-40).
12. Mangalindan, JP. 2012. “Amazon’S Recommendation Secret.” Fortune.
13. http://fortune.com/2012/07/30/amazons-recommendation-secret/.
14. Melville, P., Mooney, R. J., & Nagarajan, R. (2002). Content-boosted collaborative filtering for improved recommendations. Aaai/iaai, 23, 187-192.
15. Tran, T. N. T., Atas, M., Felfernig, A., & Stettinger, M. (2018). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems*, *50*(3), 501-526.
16. Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In *The adaptive web* (pp. 291-324). Springer, Berlin, Heidelberg.
17. Sean Coughlan: Why do we love and hate different tastes? BBC Business 10 November 2016. http://www.bbc.com/news/business-37800097
18. Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. Advances in artificial intelligence, 2009.
19. Svensson, M., Höök, K., & Cöster, R. (2005). Designing and evaluating kalas: A social navigation system for food recipes. ACM Transactions on Computer-Human Interaction (TOCHI), 12(3), 374-400.
20. Toledo, R. Y., Alzahrani, A. A., & Martínez, L. (2019). A food recommender system considering nutritional information and user preferences. IEEE Access, 7, 96695-96711.
21. Van Pinxteren, Y., Geleijnse, G., & Kamsteeg, P. (2011, February). Deriving a recipe similarity measure for recommending healthful meals. In Proceedings of the 16th international conference on Intelligent user interfaces (pp. 105-114).
22. Wang, D., Yih, Y., & Ventresca, M. (2020). Improving neighbor-based collaborative filtering by using a hybrid similarity measurement. *Expert Systems with Applications*, *160*, 113651.
23. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 52(1), 1-38.
24. Zhou, Y., Wilkinson, D., Schreiber, R., & Pan, R. (2008, June). Large-scale parallel collaborative filtering for the netflix prize. In *International conference on algorithmic applications in management* (pp. 337-348). Springer, Berlin, Heidelberg.