Recommendation Algorithm Based on Probabilistic Matrix Factorization with Adaboost

Hongtao Bai[1, 2], Xuan Li[1, 2], Lili He[1, 2], Longhai Jin[1, 2], Chong Wang[1, 2, 3] and Yu Jiang[1, 2, *]

Abstract: A current problem in diet recommendation systems is the matching of food preferences with nutritional requirements, taking into account individual characteristics, such as body weight with individual health conditions, such as diabetes. Current dietary recommendations employ association rules, content-based collaborative filtering, and constraint-based methods, which have several limitations. These limitations are due to the existence of a special user group and an imbalance of non-simple attributes. Making use of traditional dietary recommendation algorithm researches, we combine the Adaboost classifier with probabilistic matrix factorization. We present a personalized diet recommendation algorithm by taking advantage of probabilistic matrix factorization via Adaboost. A probabilistic matrix factorization method extracts the implicit factors between individual food preferences and nutritional characteristics. From this, we can make use of those features with strong influence while discarding those with little influence. After incorporating these changes into our approach, we evaluated our algorithm's performance. Our results show that our method performed better than others at matching preferred foods with dietary requirements, benefiting user health as a result. The algorithm fully considers the constraint relationship between users' attributes and nutritional characteristics of foods. Considering many complex factors in our algorithm, the recommended food result set meets both health standards and users' dietary preferences. A comparison of our algorithm with others demonstrated that our method offers high accuracy and interpretability.

Keywords: Recommendation, probabilistic matrix factorization, Adaboost, characteristics correlation.

[1] College of Computer Science and Technology, Jilin University, Changchun, 130012, China.

[2] Symbol Computation and Knowledge Engineer of Ministry of Education, Jilin University, Changchun, 130012, China.

[3] Department of Engineering Mechanics, State Marine Technical University of St. Petersburg, St. Petersburg, 190008, Russia.

[*] Corresponding Author: Yu Jiang. Email: jiangyu2011@jlu.edu.cn.

Received: 02 February 2020; Accepted: 28 June 2020.

CMC. doi:10.32604/cmc.2020.09981 www.techscience.com/journal/cmc

1 Introduction

The gradual advance of networks accompanied by a surge of information has led to the expansion of recommendation services to users in various fields. These services help users navigate volumes of information in a more precise way [Liu, Meng, Ding et al. (2019); Yin, Ding and Wang (2019)].

Diet recommendations are of particular interest, considering nutritional needs for patients in view of their dietary preferences. Current methods employ association rules, content-based collaborative filtering, and constraint-based methods. However, these traditional methods have several limitations: (1) it is easy to miss some valuable food information; (2) many particular groups of users are overlooked, such as patients with diabetes, heart disease, obesity, and hypertension; (3) traditional recommendation methods do not balance health requirements with users' acceptance; and (4) the sparsity of food data for those users with special needs leads to low-quality recommendations. The main contribution of our work is researching and addressing the current limitations. In view of the existing research, we propose a combination of probabilistic matrix factorization with Adaboost combining dietary guidelines and users' individualized criteria. Recommendation quality improves based on the user's own attributes and food characteristics. In addition to considering the recommendation results, we classify some implicit factors that contribute to the correlation between dietary preferences and food nutritional content to discover those factors making greater contributions. The association between the clients and foods leads to personal diet recommendations accounting for both health needs and client preferences. Our experimental results show that the proposed algorithm effectively balances nutritional requirements and client preferences compared with the traditional methods, especially for diabetic diets.

We divide our study as follows: Section 2 presents some problems relating to the diabetic diet and the inadequacies of traditional diet recommendation methods. Section 3 presents our proposed algorithm. Section 4 presents our experimental results and an analysis of them. Section 5 presents our conclusions.

2 Basic theory

2.1 Content-based recommendation algorithm

Content-based recommendation models consider users' preferences and food attributes separately. These models then compare historical details of a user's diet with food models [Oh and Kim (2017); Bobadilla, Ortega, Hernando et al. (2013)].

Although the content-based recommendation algorithm, without human involvement (artificial participation), is an unsupervised learning process, it is highly automated and capable of considering complex objects. However, the learning algorithm is unable to assist with newly added users. Hence, content-based recommendation systems have scalability issues.

2.2 Association-based recommendation algorithm

The most common representative of association-based recommendation algorithms is the Apriori algorithm [Liu, Yu, Wei et al. (2018); Djenouri and Comuzzi (2017)]. This algorithm generates strong association rules in a frequency set, with rules satisfying the minimum confidence support. Association analysis results in a network that describes data relationships.

Association analysis is carried out when the minimum support and confidence threshold parameters are set. As long as the minimum confidence and support do not satisfy the preset threshold, the algorithm performs a pruning process. An artificially set threshold inevitably results in missing some food contributing factors, such as users' dietary preferences during the pruning process. Thus, the extraction rules needed for association-based recommendation systems are more complicated. This method also does not handle synonyms well, such as product names or recommended content attributes, and its results are not well-individualized.

2.3 Collaborative filtering recommendation algorithm

The collaborative filtering recommendation algorithm [Zhang, Yang, Xu et al. (2017); Wen and Shu (2014); Qi, Li and Zhou (2017)] requires the user's ratings or behaviors. After calculating the similarity between users or items according to score records, a nearest neighbor dataset is obtained according to the similarity score. Then, recommendations are made relying on the selection of the target users or items in the nearest neighbor dataset.

In the case of complex project attributes, the similarity can no longer be calculated by a user's rating or behavior. In this event, the algorithm does not reassign the weights of attributes, and the attributes of the items also differ in terms of user preferences. Therefore, it is difficult for the collaborative filtering recommendation approach to deal with complex object attributes.

2.4 Constraint-based recommendation algorithm

The constraint-based recommendation algorithm [Porcel, Ching-López, Bernabé-Moreno et al. (2017); Brown (2015); Felfernig, Schippel, Leitner et al. (2013)] relies on the project properties, recommending items from the project set that match user's preferences and requirements. To meet both diet criteria and food preferences, the users' dietary structures are constrained rigidly.

In fact, if there are conflicting demands or no solution, it may be difficult to address problems using constraint solving. Although the constraint-based recommendation algorithm has improved compared with the traditional methods, the solution is obtained by gradually relaxing constraints surrounding conflicting requirements. While the demand conflicts and null solutions can be avoided, inappropriate foods matched to user's preferences rather than the nutritional need may enter the recommended dataset due to the relaxation of constraints.

3 Our algorithm

3.1 Problem definition

We define the dietary preferences set for users as $U=\{u_1, u_2, \ldots, u_n\}$ and the food attributes set as $V=\{v_1, v_2, \ldots, v_m\}$, as shown in Tab. 1.

Table 1: Characteristic correlation matrix

	v_1	…	v_b	…	v_m
u_1	$r_{1,1}$	…	$r_{1,b}$	…	$r_{1,m}$
…					
u_a	$r_{a,1}$	…	$r_{a,b}$	…	$r_{a,m}$
…					
u_n	$r_{n,1}$	…	$r_{n,b}$	…	$r_{n,m}$

In the characteristic correlation matrix (CCM), the *n* rows represent the *n* dietary preference characteristics of a diabetic patient, and the *m* columns represent the nutritional characteristics of foods [Felfernig, Schippel, Leitner et al. (2013); Xu and Xie (2017)]. If a diabetic patient has a preference on a food, the value of *r* will be distributed in the matrix. By calculating *r*, we can find combinations of users and foods satisfying both preferences and requirements.

3.2 Probabilistic matrix factorization

The basic assumption of probabilistic matrix factorization [Shi, Zheng and Yang (2017); Wang, Liu, Xia et al. (2017); Ren, Song, E et al. (2017)] is that only a small number of implicit factors contribute to the correlation between a user's preferences and an item's attributes. In the case of the personalized diet recommendation algorithm, there is a small number of implicit factors contributing to the correlation between user's dietary preferences and food characteristics.

We extract the users' diet preferences to form a low-dimensional matrix $U \in R^{K\times M}$ and the food attributes to form a low-dimensional matrix $V \in R^{K\times N}$, where K $\leq$ min {M, N}. We use the inner product of *U* and *V* to constitute a "Users-Food Characteristics Correlation" matrix.

Given the users' diet preferences characteristic vector U_i and food's feature vector V_i, the correlation between users' preferences and food attributes is quantified as r_{ij}, which has the distribution:

$$N\left(r_{ij} \mid U_i^T V_j, \sigma_R^2\right) = \frac{1}{\sqrt{2\pi}\sigma_R} e^{\frac{\left(r_{ij}-U_i^T v_j\right)^2}{2\sigma_R^2}}, \tag{1}$$

where $\mathrm{N}(x \mid \mu, \sigma^2)$ is a density function of the Gaussian distribution with an average value μ and squared difference σ^2. If each r_{ij} is independent, the conditional probability of observable correlation strength is as follows:

$$P\left(R \mid U, V, \sigma_R^2\right) = \prod_{i=1}^{M} \prod_{j=1}^{N} \left[N\left(r_{ij} \mid U_i^T V_j, \sigma_R^2\right)\right]^{I_{ij}}, \tag{2}$$

where I_{ij} is an indicator function. If a user *i* has an action on food *j*, the value of I_{ij} is 1, otherwise it is 0.

3.3 Adaboost classifier factor

Adaboost is an iterative algorithm that trains different weak classifiers using the same training set and combines them to form a stronger one [Dinakaran and Thangaiah (2017); Zhang and Yang (2016)]. If the distribution correlation between a user's dietary preference and a food attribute is considered as a weak classifier, the Adaboost algorithm will filter it into a strong classifier.

In the CCM, all r_{ij} ($r_{ij}\neq0$), which, respectively, correspond to $x_1, x_2, \ldots, x_N$, are extracted to form a set T, where $T=\{(x_1,y_1), (x_2,y_2), \ldots, (x_N,y_N)\}$ with y_i denoting the mark collection $\{+1, -1\}$. Each r_{ij} is initially given the same weight 1/, with $D_1=(w_{11}, w_{12}, \ldots, w_{1i}, \ldots, w_{1N})$, $w_{1i}=1/N$, and i=1, 2, …, N. The number of iterations is denoted by m=1, 2, …, M. Using a training dataset of D_m with a weight distribution to learn, the basic classifier is obtained: $G_m(x):\chi\rightarrow\{+1,-1\}$.

We then calculate the classification error rate of $G_m(x)$ on the training data set as follows:

$$e_m = P(G_m(x_i)\neq y_i)=\sum\nolimits_{i=1}^{N} w_{mi} I(G_m(x_i)\neq y_i). \tag{3}$$

The error rate e_m on the training dataset is the sum of sample weights misclassified by $G_m(x)$. We calculate the coefficient of $G_m(x)$, with a_m representing the importance degree of $G_m(x)$ in the set of final relevance degree r_{ij}. Because the algorithm proppposed in this paper will combine $G_m(x)$ and r_{ij} into $G_m(r_{ij})$ in the following description, and a_m shows the importance of $G_m(r_{ij})$ in the final classifier. The purpose is to calculate the weight of each relevance degree in the final relevance degree set to determine whether both nutritional and individual preferences are met. $a_m = 1/2\log(1-e_m)/e_m$, where when $e_m \leq 1/2$, $a_m \geq 0$ means the degree of relevance classification error rate and a_m increases as the e_m decreases. A smaller set of basic relevance levels affects the diabetic patient more in the final set. a_m is artificially set by consulting attending physicians and nutritionists.

As the previous degree of association has been classified according to a certain feature set, there is an indirect effect on users. Namely, it affects users when the feature set of the association degree approaches the threshold. Using the new training set of weight distributions, the calculations during the next iteration become:

$$D_{m+1}=\left(w_{m+1,1}, w_{m+1,2}, \ldots, w_{m+1,i}, \ldots, w_{m+1,N}\right) \tag{4}$$

$$w_{m+1,i}=\frac{w_{mi}}{z_m}\exp\left(-a_m y_i G_m\left(x_i\right)\right) \quad i=1,2,\ldots,N. \tag{5}$$

We increase the relevance class weight of error class $G_m(x)$ for the basic relevance set so that the weighted value decreases correctly. When this happens, we find those samples that are more difficult to classify and thus have no effect on diabetics. Therefore, we can remove those factors. Among them, Z_m is a normalization factor introduced by Adaboost that enables D_{m+1} to be a probability distribution:

$$Z_m=\sum\nolimits_{i=1}^{N} w_{mi}\exp\left(-a_m y_i G_m\left(x_i\right)\right). \tag{6}$$

Finally, we assign weights to all of the associations and make a classification as well. Again, each category is weighted and classified, which means that each set is combined as follows:

$$f\left(x\right)=\sum\nolimits_{m=1}^{M}a_{m}G_{m}\left(x\right). \tag{7}$$

The final classification of relevance is then expressed as follows:

$$G_{m}\left(x\right)=\operatorname{sign}\left(f\left(x\right)\right)=\operatorname{sign}\left(\sum\nolimits_{m=1}^{M}a_{m}G_{m}\left(x\right)\right). \tag{8}$$

3.4 Description algorithm

In the probabilistic matrix factorization, each r_{ij} is independent, and all of the non-zero r_{ij} are classified by an Adaboost classifier. So the final classifier $G_m(x)$ obtained by Adaboost algorithm is introduced into the probability matrix decomposition namely "Users-Food Characteristics Correlation" matrix, to replace the position of r_{ij} to processr_{ij}, thus $G_m(r_{ij})$ can be generated. Each type of association contributes to users' personalized diet, which corresponds to different indicators. The algorithm balances the weight relationship between personal preferences and nutritional needs.

By combining Eqs. (2) with (8), we obtain the following:

$$P\left(R\,|\,U,V,\sigma_{R}^{2}\right)=\prod\nolimits_{i=1}^{M}\prod\nolimits_{j=1}^{N}\left[N\left(G_{m}\left(r_{ij}\right)|\,U_{i}^{T}V_{j},\sigma_{R}^{2}\right)\right]^{I_{ij}}. \tag{9}$$

Our algorithm does not require an artificial threshold set, thus avoiding the loss of valuable data due to pruning. The problem of non-modeling (i.e., null solutions) cannot occur even with the addition of new users as the association between food attributes and user's preferences is bound. At the same time, our algorithm analyzes loosely constrained, multi-attribute problems without empty solutions. Hence, it will always recommend its best results in the existing range available to diabetics, satisfying both nutritional requirements and personal preferences.

When the screening and classifying of all of the non-zero r_{ij} values are completed, the final set of associations is the upper bound of error. The screening and classifying processes exclude non-compliant relevance values, reducing the error e in the process of elimination via the test:

$$\frac{1}{N}\sum\nolimits_{i=1}^{N}I\left(G\left(r_{ij}\right)\neq y_{i}\right)\leq\frac{1}{N}\sum\nolimits_{i}\exp\left(-y_{i}f\left(r_{ij}\right)\right)=\prod\nolimits_{m}Z_{m}\quad . \tag{10}$$

Although there are many different types of algorithms for diet recommendation, the content-based and collaborative filtering algorithms are most common. Specifically, the content-based dietary recommendation requires both basic food attributes (calories, protein, fat, and carbohydrates) and users' dietary records. Collaborative filtering requires data for each edible food score, food weight, and food caloric value. We show the steps for the three algorithms in Algorithms 1–3.

Algorithm 1: Diabetic diet recommendation algorithm based on collaborative filtering

```
Input: Scoring matrix, caloric value, and weight of food
Output: Result set of food for simulation
Take $K$ nearest neighbors.
Divide the dataset into test and training sets, using $1/m$ of the data as the test set. Using different $K \leq m-1$, obtain different test and training sets under the same random seed condition.
Calculate the similarity between foods under simulated conditions and calculate the quantified similarity between vectors ItemA and ItemB to form the similarity matrix.
Sort the similarity matrix in the simulation and calculate the degree of interest matrix using $K$ nearest neighbors.
Recommend the results according to the interest matrix.
```

Algorithm 2: Diet recommendation based on the content

```
Input: The basic attributes of food and diabetic diet records
Output: Food result set for simulation
Model the attributes of the food ($V$).
Model the historical diet records of users ($U$).
if $V = U$, return 1
Recommend result set.
```

Algorithm 3: Our algorithm

```
Input: Scoring matrix, basic attributes of food, physical indicators of the user, dietary preferences
Output: Food result set for the recommended simulation Q
The basic attributes of food $V = \{v_1, v_2, \ldots, v_m\}$ and dietary preference characteristics of users $U = \{u_1, u_2, \ldots, u_n\} \to r_{ij}$
while True
    $f(x) = \sum_{m=1}^{M} a_m G_m(x) \to f(r_{ij})$
    if $\frac{1}{N} \sum_{i=1}^{N} I\left(G\left(r_{ij}\right) \neq y_i\right) \leq \prod_m Z_m$ , return 1
    $f(r_{ij}) G_m(r_{ij}) = \mathrm{sign}(f(r_{ij}))$
  if $\mathrm{sign}(f(r_{ij})) == 0$ break
end while
Analyze all food, return Q
```

Algorithm 1 carries out the AB test constitute a similarity matrix and then constitutes a matrix of interest degree. Its time complexity is $3\times \mathrm{O}(n^2)$. Algorithm 2 generates a matrix of historical diet records then matches the food attributes and historical diet records based on the n-model level. Its time complexity is $2\times \mathrm{O}(k)+\mathrm{O}(n^2)$. Algorithm 3 forms the relevance matrix with a time complexity $\mathrm{O}(n^2)+\mathrm{O}(k)$.

4 Experiments

4.1 Introduction to the data set

To highlight the accuracy of our algorithm in food recommendation compared with others, we selected a particular group of diabetic users and collected their data. However, diabetic patients can be non-compliant, as they may go to another hospital to receive treatment after leaving one. When this occurred, the patient's data was distributed across multiple hospitals, which led to incomplete data. In addition to collecting data from various hospitals, we interviewed and surveyed many patients. This collation and analysis was intended to make the data as reasonable as possible. For patients, we also analyzed the corresponding food characteristics including quantified measures for taste, texture, calories, fats, proteins, and carbohydrates. We also measured patients' physical indexes, including weight, blood pressure, glycosylated hemoglobin level, plasma glucose level, and working intensity.

4.2 Experimental evaluation indicator

Recommendation algorithms are traditionally evaluated according to accuracy rate, recall rate, and coverage rate [Wang, Gong, Li et al. (2016); Han and Yamana (2017)]. However, for the health field, we calculate only the precision rate and diversity:

$$\text{Precision}=\frac{\sum_u \left|R(u)\cap T(u)\right|}{\sum_u \left|R(u)\right|} \tag{11}$$

$$\text{Diversity}=\frac{2}{n(n-1)}\sum_{u,v\in U}\frac{\left|L_u\cap L_v\right|}{\left|L_u\right|}. \tag{12}$$

4.3 Comparison of experimental methods

We compared the content-based diet recommendation algorithm (CB), the algorithm using collaborative filtering and our algorithm for analysis (CF). We varied the number of nearest neighbors *K* as 20, 40, 60, 80, 100, 120, 140, 160, 180, and 200 to represent an increasing number of users. We also set *K* as 5, 10, 20, 40, 80, 160, and 320 to compare the diversity of result sets recommended by each algorithm.

Because our interest is in diabetic diets, we compared the accuracy of health and personalized indicators of each algorithm. We selected a diabetic patient at random for simulation, with 4 hours as the basic time interval. We compared the recommended food with the food selected by the participating users to calculate the accuracy of the diet's compatibility with user preferences. For nutritional requirements and health indicators, we compared the nutritional content of recommended foods with user needs to calculate accuracy. Then we analyze the relationship between diversity and accuracy to determine their trend.

Finally, we tracked test data results. By simulating the physical indexes of users (only plasma glucose and body weight), we compared each algorithm's impact on users' physical indexes.

4.4 Experiments and result analysis

We performed time-based random generation to simulate the physical indicators of each diabetic patient and their diet preferences According to the dietary preferences, we modeled the users with the recommended food result set and calculated the diversity of the food result sets. In the individualized dietary areas, the extent to which individual algorithms have been personalized in this study is shown in Figs. 1 and 2 below.

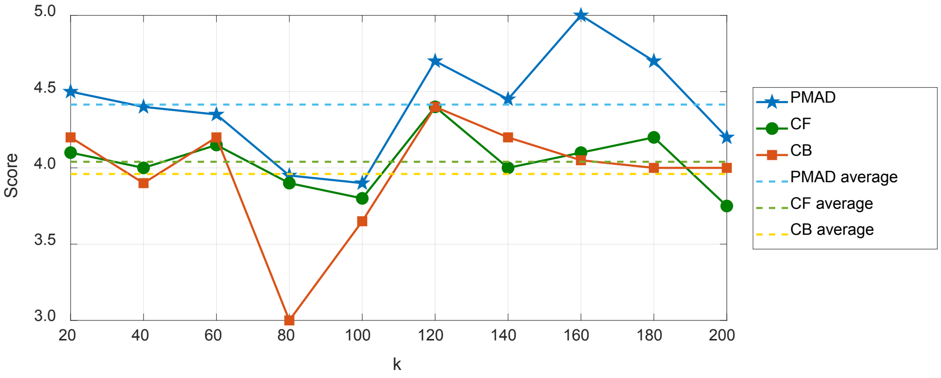

Figure 1: Score distribution

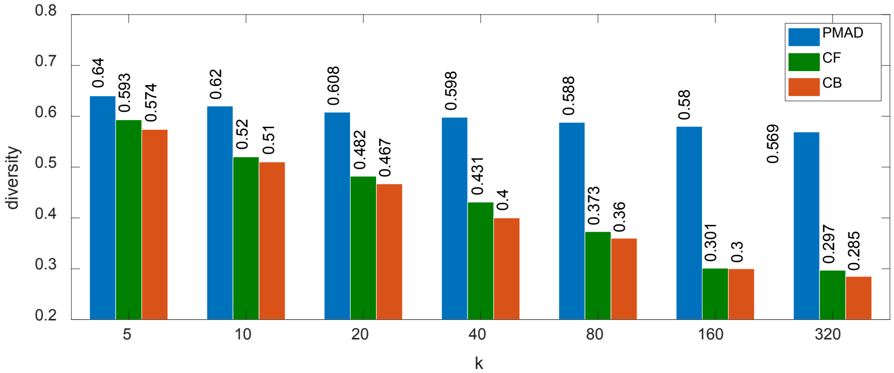

Figure 2: Comparison of diversity

Fig. 1 shows that the food result set recommended by our algorithm was favored by simulated users with a relatively high average score. The content-based diabetic diet recommendation algorithm had a large gap when K=80 due to its inability to handle newly added users.

Fig. 2 shows that the diversity of our algorithm was higher than the others, indicating better support for complex attributes. Our algorithm simulated users' physical indicators and diet preferences, and we analyzed the recommended results comprehensively. The other two algorithms did not perform well with complex properties.

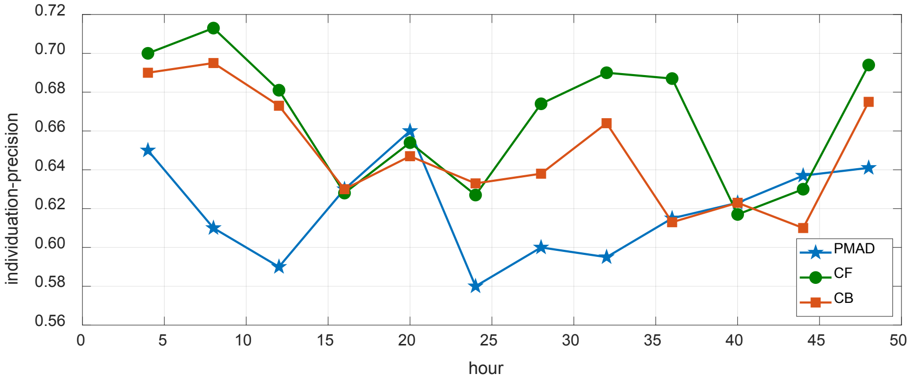

Figure 3: Distribution of the personalized accuracy rate

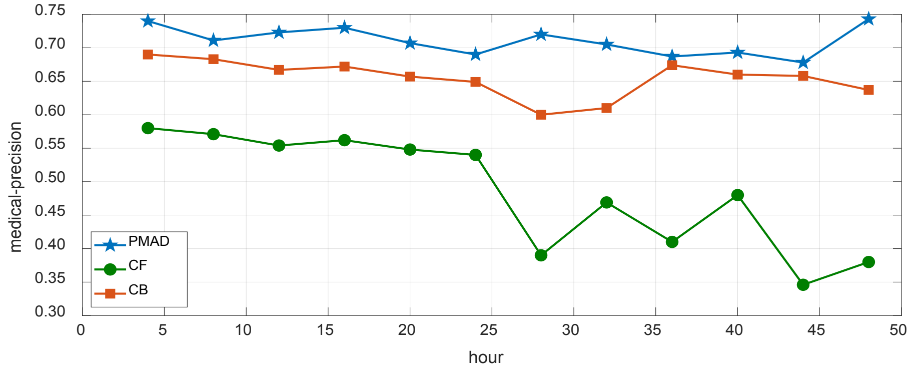

Figure 4: Distribution of the health indicators accuracy rate

Fig. 3 shows that the personalized accuracy rate of our algorithm was lower than that of others. This occurred because the diet preferences drive users to choose their favorite ingredients in the selection process, regardless of the ingredients' nutrient proportion and value.

We emphasize that the diversity and personalization results from our recommendation algorithm are two different concepts, reflecting different needs. A comparison of Figs. 2 and 3 shows that diversity is the relationship between the recommended food result set and all of the food suitable for diabetics. The individualization is the relationship between users' food preferences and recommendation set.

Fig. 4 shows the good therapeutic effect of our algorithm. The other algorithms placed too much emphasis on the patients' dietary preferences at the expense of nutritional suitability.

In this study, we initially set the body weight and plasma glucose level of each patient as 59 kg and 8.5 mmol/L, respectively, in the initial simulation, with patients taking food every 4 hours. In the simulation, the patient consumed 0.005 kg of body weight per hour, with the plasma glucose levels increasing by 0.001 mmol/L per hour. The physical indicators of randomly sampled simulated users were detected within 72 hours, as shown in Figs. 5 and 6.

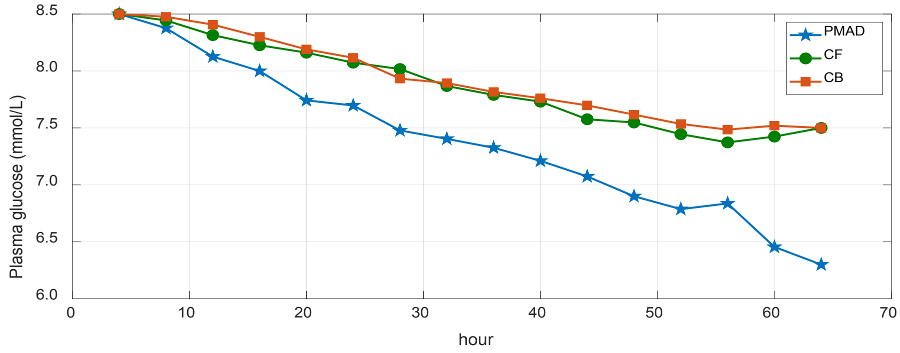

Figure 5: Plasma glucose level

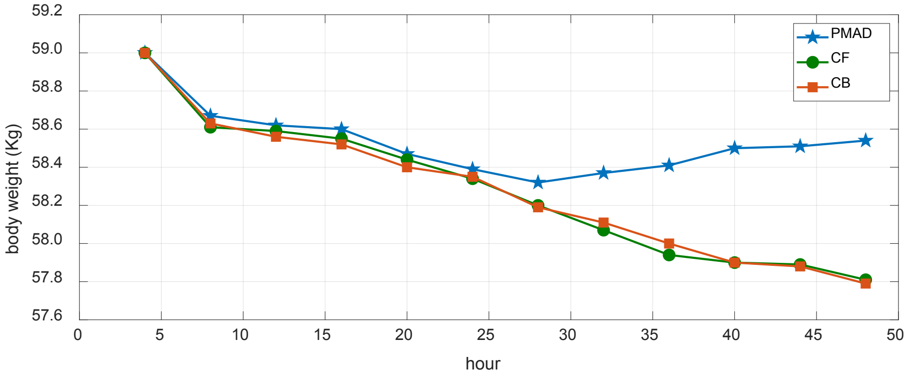

Figure 6: Body weight

Fig. 5 shows that although all of the three algorithms reduced the plasma glucose level, our algorithm offered better performance than the other two. By rating the plasma glucose content (excellent 4.4-6.1, good≤7.0, poor>7.0, in mmol/L), we conclude that our algorithm reduced the plasma glucose of participating users to healthier levels.

By the end of the second day, a turning point appeared. Our algorithm increased users' body weight and maintained the body weight balance for some time, while the other algorithms only minimized weight loss.

5 Conclusion

Making use of traditional dietary recommendation algorithm research, we incorporate the Adaboost classifier into our algorithm to strengthen the probabilistic matrix factorization. The algorithm fully considers of the constraint relationship between users' own attributes and nutritional characteristics of foods. Considering many complex factors in our algorithm, the recommended food result set meets both the health standards and users' dietary preferences. Therefore, our algorithm improves personalized diet recommendations, tailoring them to both patient preferences and nutritional needs. We anticipate that improved data collection methods using wireless networks [Wang, Zuo, Shen et al. (2015); Ullah, Abdullah, Kaiwartya et al. (2017)] will further improve the portability and scalability of our algorithm in the future.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China (51679105, 51809112, 51939003, 61872160); "Thirteenth Five Plan" Science and Technology Project of Education Department, Jilin Province (JJKH20200990KJ).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. (2013): Recommender systems survey. *Knowledge-Based Systems*, vol. 46, pp. 109-132.

Brown, K. N. (2015): Design and evaluation of a constraint-based energy saving and scheduling recommender system. *International Conference on Principles and Practice of Constraint Programming*, vol. 2015, pp. 687-703.

Chen, C. H.; Karvela, M.; Sohbati, M.; Shinawatra, T.; Toumazou, C. (2018): PERSON-personalized expert recommendation system for optimized nutrition. *IEEE Transactions on Biohealth Circuits and Systems*, vol. 12, no. 1, pp. 151-160.

Dinakaran, S.; Thangaiah, P. R. J. (2017): Ensemble method of effective AdaBoost algorithm for decision tree classifiers. *International Journal on Artificial Intelligence Tools*, vol. 26, no. 3, pp. 1-21.

Djenouri, Y.; Comuzzi, M. (2017): Combining apriori heuristic and bio-inspired algorithms for solving the frequent itemsets mining problem. *Information Sciences*, vol. 420, pp. 1-15.

Felfernig, A.; Schippel, S.; Leitner, G.; Reinfrank, F.; Isak, K. et al. (2013): Automated repair of scoring rules in constraint-based recommender systems. *AI Communications*, vol. 26, no. 1, pp. 15-27.

Han, J.; Yamana, H. (2017): A survey on recommendation methods beyond accuracy. *IEEE Transactions on Information & Systems*, vol. 100, no. 12, pp. 2931-2944.

Hao, F.; Park, D. S.; Woo, S. Y.; Min, S. D.; Park, S. (2016): Treatment planning in smart health: A sustainable strategy. *Journal of Information Processing Systems*, vol. 12, no. 4, pp. 711-723.

Liu, G. S.; Meng, K.; Ding, J. C.; Nees, J. P.; Guo, H. Y. et al. (2019): An entity-association-based matrix factorization recommendation algorithm. *Computers, Materials & Continua*, vol. 58, no. 1, pp. 101-120.

Liu, L.; Yu, S.; Wei, X.; Ning, Z. (2018): An improved apriori-based algorithm for friends recommendation in microblog. *International Journal of Communication Systems*, vol. 31, no. 2, pp. 1-11.

Oh, Y.; Kim, Y. (2017): A customized healthy menu recommendation method using content-based and food substitution table. *KIPS Transactions on Software and Data Engineering*, vol. 6, no. 3, pp. 161-166.

Porcel, C.; Ching-López, A.; Bernabé-Moreno, J.; Tejeda-Lorente, A.; Herrera-Viedma, E. (2017): Fuzzy linguistic recommender systems for the selective diffusion of information in digital libraries. *Journal of Information Processing Systems*, vol. 13, no. 4, pp. 653-667.

Qi, S.; Li, S.; Zhou, H. (2017): Collaborative filtering algorithm based on forgetting curve and long tail theory. *Advances in Materials, Machinery, Electronics*, vol. 1820, no. 1-5.

Ren, X.; Song, M.; E, H.; Song, J. (2017): Context-aware probabilistic matrix factorization modeling for point-of-interest recommendation. *Neurocomputing*, vol. 241, pp. 38-55.

Shi, J.; Zheng, X.; Yang, W. (2017): Survey on probabilistic models of low-rank matrix factorizations. *Entropy*, vol. 19, no. 8, pp. 1-33.

Ullah, F.; Abdullah, A. H.; Kaiwartya, O.; Kumar, S.; Arshad, M. M. (2017): Medium access control (MAC) for wireless body area network (WBAN): superframe structure, multiple access technique, taxonomy, and challenges. *Human-Centric Computing and Information Sciences*, vol. 7, no. 34, pp. 1-39.

Wang, J.; Zuo, L.; Shen, J.; Li, B.; Lee, S. (2015): Multiple mobile sink - based routing algorithm for data dissemination in wireless sensor networks. *Concurrency & Computation Practice & Experience*, vol. 27, no. 10, pp. 2656-2667.

Wang, S.; Gong, M.; Li, H.; Yang, J. (2016): Multi-objective optimization for long tail recommendation. *Knowledge-Based Systems*, vol. 104, pp. 145-155.

Wang, Y.; Liu, F.; Xia, S. T.; Wu, J. (2017): Link sign prediction by variational Bayesian probabilistic matrix factorization with student-t prior. *Information Sciences*, vol. 405, pp. 175-189.

Wen, J. H.; Shu, S. (2014): Improved collaborative filtering recommendation algorithm of similarity measure. *Computer Science*, vol. 41, no. 5, pp. 68-71.

Xu, A. B.; Xie, D. (2017): Low-rank approximation pursuit for matrix completion. *Mechanical Systems & Signal Processing*, vol. 95, pp. 77-89.

Yin, C. Y.; Ding, S. L.; Wang, J. (2019): Mobile marketing recommendation method based on user location feedback. *Human-centric Computing and Information Sciences*, vol. 9, no. 1, pp.1-17.

Zhang, P. B.; Yang, Z. X. (2016): A novel AdaBoost framework with robust threshold and structural optimization. *IEEE Transactions on Cybernetics*, vol. 48, no. 1, pp. 64-76.

Zhang, S.; Yang, W.; Xu, S.; Zhang, W. (2017): A hybrid social network-based collaborative filtering method for personalized manufacturing service recommendation. *International Journal of Computers Communications & Control*, vol. 12, no. 5, pp. 728-740.