ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

Music Recommendation Algorithm based on Contex-tual Semantics

Yang Zhou, Cuiju Luan

School of Information Engineering, Shanghai Maritime University, Shanghai 201306, China.

Abstract

User listening to songs is sequential, while traditional music recommendation algorithms do not con-sider the relationship between listening behaviors and their context, so a music recommendation algo-rithm based on contextual semantics is proposed. Different from the traditional collaborative filtering recommendation algorithm based on user-scoring matrix, this algorithm abstracts the listening records of different users into time-series text sequences, and uses natural language models to capture songs in the process of song conversion. Context semantic relations. For quantized songs, it can be mapped to a high-dimensional space, and the similarity relationship can be obtained by calculating the distance between the songs. For the user, through the user's music listening records, the user can be portrayed to the same dimensional space as the song. Similarly, by calculating the distance between the user and the song, the similarity between the user and the song can be obtained, so as to achieve the purpose of recommendation. Compared with traditional recommendation algorithms, this algorithm can not only capture contextual semantic relationships that traditional recommendation algorithms cannot capture, but also has better overall top-n recommendation performance than traditional algorithms.

Keywords

Word2vec; Recommendation Algorithm; Music Recommendation; Contextual Semantic.

1. Introduction

With the development of music industry, the behavior of music lovers has changed greatly. In the past, people bought tapes to listen to music, but now people prefer music streaming services, such as foreign Amazon Music, Apple Music and domestic NetEyun Music, QQ Music and so on. These music service platforms provide people with a faster and more convenient way to listen to songs, and users can directly search and find their favorite songs. It is troublesome for users to find a large number of songs they like and create playlists. Therefore, how to find users' favorite songs quickly and accurately in the massive music data information has become particularly important and valuable. As a result, the recommendation system has become a bridge between meeting the needs of users and content [1]. Through the recommendation system, we can not only meet the preferences of users, but also find potential users for each song.

Nowadays, when users open the music software, different users display different playlists. This process is the application process of recommendation algorithm. At present, the commonly used music recommendation algorithms are mainly divided into three types:Popularitybased(PB), Contentbased (CB), and CollaborativeFiltering(CF) [2,3,4].

In general, most recommended algorithms are integrated with existing recommended algorithms and other algorithms. Improve the efficiency of the recommended algorithm by mixing the recommended form. This recommendation algorithm is good, but the interference of the parameter selection in the

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

matrix of the matrix and the parameter selection of the algorithm, when the user-rating matrix is extremely sparse, the efficiency of the algorithm is difficult to guarantee, and the recommended effect on TOP-N is not well. Therefore, this paper proposes a recommended algorithm based on context semantics, and verifies the favorable and depletion of the algorithm by comparative experiments.

2. Related Research

2.1 Music Recommended Algorithm based on Popularity

The most popular music recommended algorithm is the recommendation of the calculation of the flocking. There are many definitions and computational methods, such as the number of playings of music listening to songs in a period of time, or using more complex probability-based calculation methods. However, regardless of how the pattern is calculated, time factors and spatial factors are two main factors affecting popularity.

For example, for the number of music samples n, the number of user samples is N, the popularity of the song 'i' is the number of songs of song 'i', which is the number of songs 'i', then the weight of the song 'i' is defined as:

$$weight(i) = \frac{\frac{1}{n}\sum_{n}^{j=1} popularity(j)}{popularity(i)} \tag{1}$$

Then the similarity between user A and user B is defined as:

$$similarity(A,B) = \frac{\sum_{i \in N(A) \cap N(B)} weight(i)}{\sum_{i \in N(A) U N(B)} weight(i)} \tag{2}$$

Note the number of times that user k plays the song 'i' as $nums(i)$, and consider the influence of popularity, then user k scores the song 'i' as follows:

$$score(i) = \frac{nums(i)}{log(1+popularity(i))} \tag{3}$$

Finally, the user k scores each song 'i', taking Top-n as a recommendation.

2.2 Music Recommended Algorithm based on Content

The basic idea of this kind of recommendation algorithm is to analyze the characteristics of the user's listening behavior according to the user's historical music listening records, get the user's listening style preferences, and then recommend the songs of the same feature type to the changed users. For this kind of music recommendation algorithm, there are label content-based recommendation algorithm and music feature-based music recommendation algorithm [5].

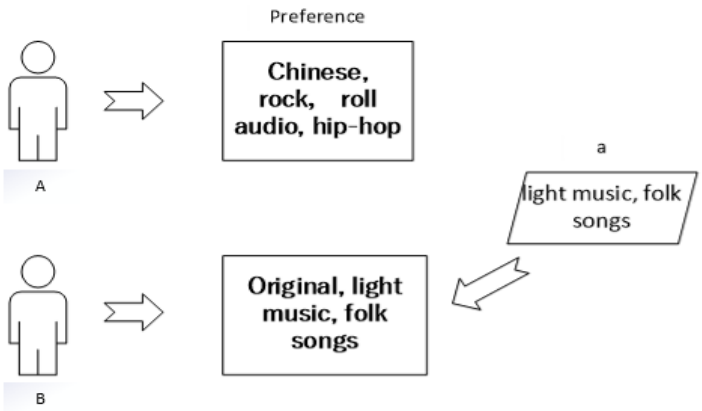

Figure 1. Content based music recommendation algorithm

For example, user A's listening style prefers to listen to music such as Chinese, rock, electric and hip-hop. User B likes to listen to songs such as original, light music and folk songs. Then songs a that

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

belong to light music and folk songs will be recommended to priority user B. This is the content-based music recommendation algorithm.

2.3 Collaborative Filtering Music Recommendation Algorithm

The core idea of collaborative filtering recommendation algorithm is to make use of the similarity or similarity of user preferences to recommend content. The algorithm mainly includes three types: song-based collaborative filtering recommendation algorithm, user-based collaborative filtering recom-mendation algorithm and model-based collaborative filtering algorithm [6].

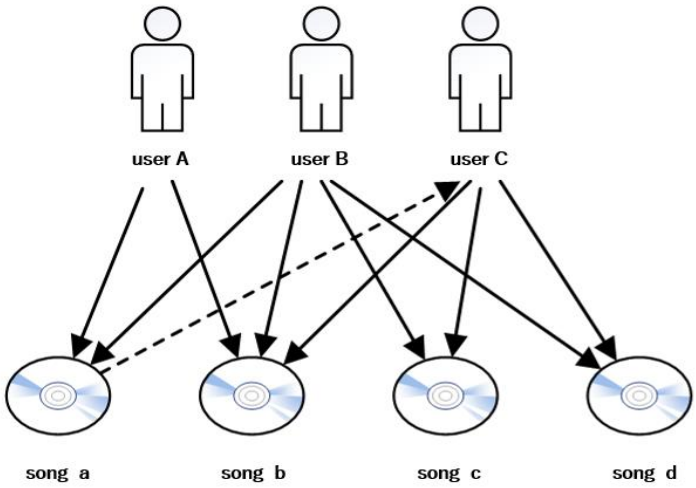

Figure 2. User-based music recommendation algorithm

For example, there are three users: user A, user B, and user C. It is not difficult to see from the Figure 2. that the songs listened to by user B and user C are more similar, so the song a can be recommended for user C from user B's history.

For the music recommendation algorithm based on popularity, the complexity of the algorithm is low in practice, and only needs to count the music playback times in a specific practice or space, so the running times of the algorithm is fast, and there is no cold start problem. Due to the lack of information involved, the effect of recommendation is not very good. For the content-based music recommendation algorithm, the algorithm needs to tag the music type in advance, but human errors will be introduced in the process of tagging. And the accuracy of the limited song tags is also worth considering. For the us-er-based music recommendation algorithm, the most fatal defect is the sparsity of the matrix and the cold start problem.

In order to solve the above problems, this paper proposes a music recommendation algorithm based on context semantics, which can not only start cold and migrate to different music platforms after one-time training, but also capture the context semantic relationship between songs.

3. Music Recommendation Algorithm based on Contex-tual Semantics

3.1 Algorithm Framework

The recommended algorithm based on context semantics mainly includes the following three parts. First, in accordance with different users, the data sets are collected, collect each user's listening song, and then encode different songs, use the CBOW in Word2VEC to train. After getting the space of each song, you have all the listened songs for each user to get the user's spatial vectors. Finally, by calculat-ing the most recent song in the space, it can be recommended as a Top-n recommended.

Traditional Word2Vec is developed by Tomasmikolov on Google in 2013, which is one of the most common technologies that use shallow neural networks in several natural language processing (NLP) cases. A good class ratio is to represent the color using the RGB value in the industry. In the three-

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

dimensional coordinate space, the spatial vector is constructed by the three dimensions of red (R), Green (G), and Blue (B), and any coordinate point on this space can represent a color. For example, "black" can be associated with (0, 0, 0) and "white" (255, 255, 255).

Like other supervisory learning algorithms, neural networks need to mark data for training. How-ever, in the form of a word sequence (ie, a word) or a song sequence (ie, a playlist) is there is no target or data tag. Therefore, the network is trained by creating a so-called "forged" task. We will not be interested in the input and output of the network, and only the weight between the input layer and the hidden layer can be extracted into a vector (ie the associated spatial vector). Briefly, embedded goals can be classified as non-supervised learning, but the embedded process in Word2VEC is implemented through a neural network. This is the illustration of a general Word2Vec architecture:

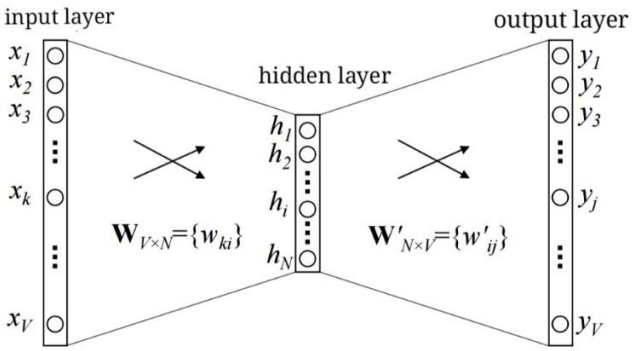

Figure 3. Word2vec architecture

Wherein, the input layer is a vector that passes the one-hot encoded. W_(V×N) is the weight matrix of projecting input x to the hidden layer. The hidden layer contains N neurons, which only copies the weighted sum of the input to the next layer. 〖W"" 〗_(V×N) is the weight matrix that maps the hidden layer output to the final output layer. The output layer is also a V length vector having a softmax activation function.

There are two ways of Word2Vec use the same architecture: skip-gram is through a given target word, the model will try to predict the context word. CBOW (Continuous Bag of Words) is based on a given context vocabulary prediction target word.

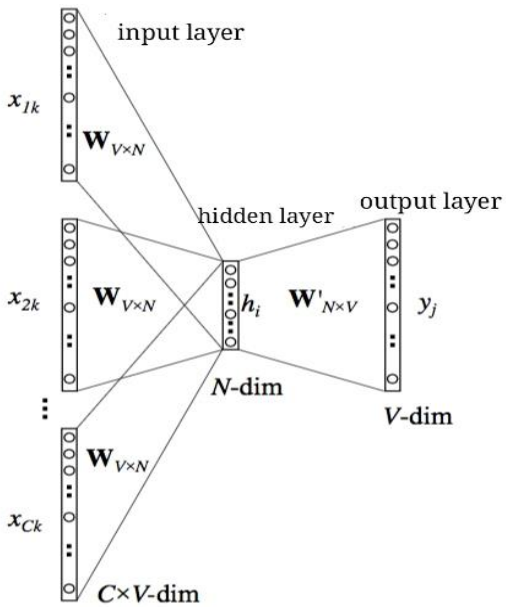

Figure 4. CBOW model structure

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

In the context-based semantic music recommendation algorithm, the CBOW architecture will be used, and the CBOW training is faster and can capture frequent songs. Assuming that the target song played between context songs is similar. If the playlist is designed by the user or service as a specific type, the song embedding will logically contain more information about this type.

The generated music playlist training sample is converted into an one-hot vector $x_1, x_2, \ldots, x_C$ (context) of the input layer, and therefore, the size is $C \times V$. All vector x is then multiplied by $W_{V\times N}$ and then the mean of embedded vents. The hidden layer will be multiplied by $W'_{V\times N}$ to obtain V. $\hat{y}$ is expressed as the probability of weighing and transformation through the softmax function, y_i represents the song that the user actually hesperesses. The error between the output and each context word is $\hat{y} - y$. Repeat the above steps to continuously iterate to reduce the error to achieve the degree capable of acceptability. Where the softmax function is defined as follows:

$$softmax(y_i) = \frac{e^{y_i}}{\sum_{V}^{y=1} e^{y_j}} \tag{4}$$

The expression of the optimization function ζ is:

$$\zeta = \sum_{W_i} lbp(W_i | Content(W)) \tag{5}$$

Where W_i is a target word, $Content(W)$ is a context.

The similarity between songs A,B is calculated using a cosine similarity. It measures the cosine between the two vectors A and B projected in the multidimensional space. The song vector having similar contexts occupies a tight spatial position, and the cosine value between such vectors should be close to 1, that is, the angle is closer to 0, the smaller the angle between the two versions of A and B, the lower the cosine similarity.

$$similarity(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} \tag{6}$$

3.2 Algorithm Implementation

3.2.1 Dataset

In this article, using an American radio playlist from Yes.com and a song tag from Last.fm, each playlist is treated as a sentence, and each song in the playlist is treated as a word. There are 15842 playback records and 75262 songs in the dataset.

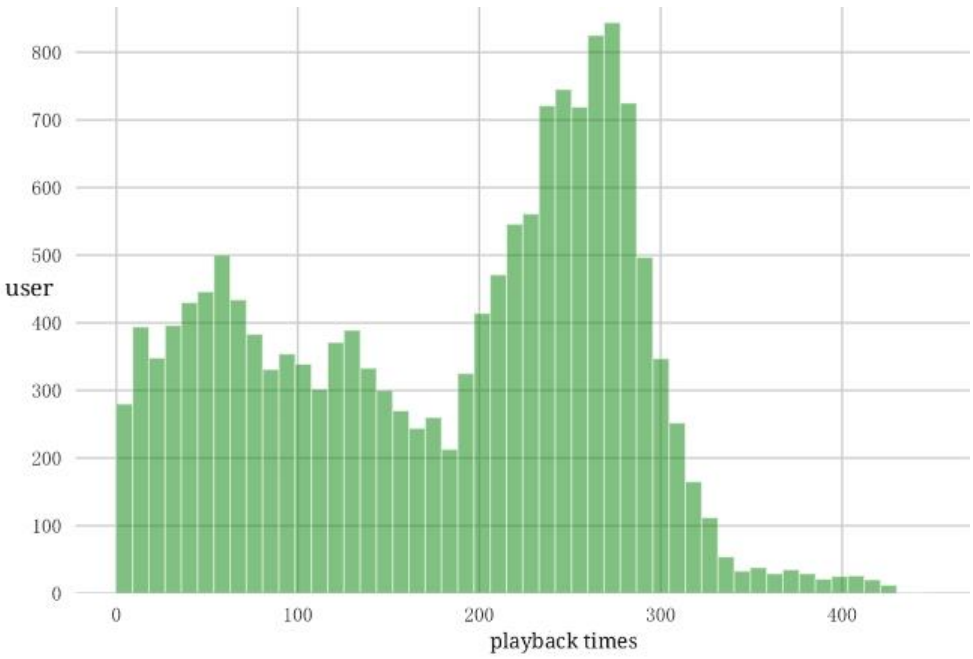

Figure 5. Distribution of user play volume

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

Figure 5. shows that, most users' playback is concentrated within 300. The number of users in which the playback is between 200 and 300 is most.

3.2.2 Algorithm Implementation Process

Make sure the loss function image is recorded in the iterative process and draws the loss function image. The closer loss, the model is predicted by the performance song in the case of a given environment song, the model is predicted. Therefore, the resulting song vector is more meaningful. As can be seen from the figure, the error reduction during the top 20 iterations, the error reduction in the 20 to 800 iterations tends to slow, and it tends to stabilize.

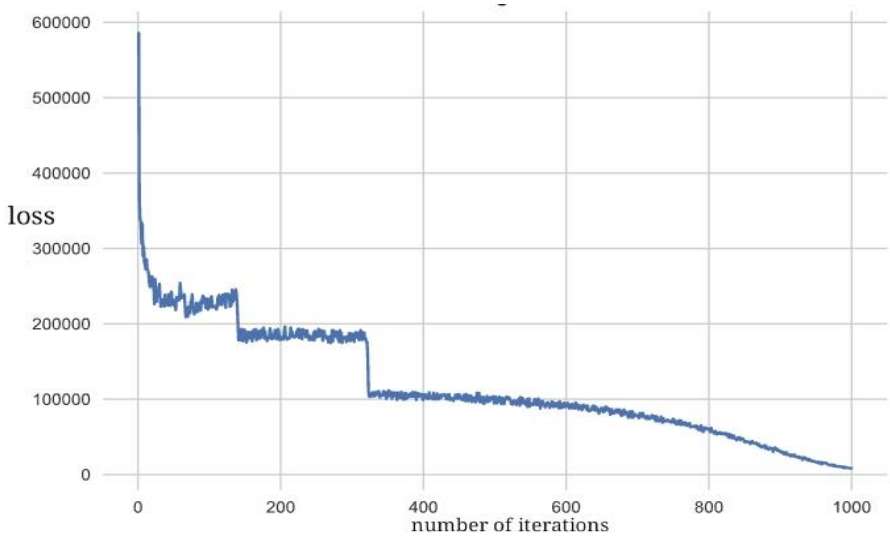

Figure 6. Error after each iteration

The song vector can be visualized using the color gradient. This model uses 256 dimensional training, so each song will have 256 color bars, representing the elements in the vector.

The picture shows song id 4162 by Magic Lihong (Maroon5, American pop rock band) singing SheWillBeLoved. You can see that the songs closest to it are ThisLove and Misery, which are also sung by Mok Lihong with a similarity of 0.555 and 0.456, respectively.

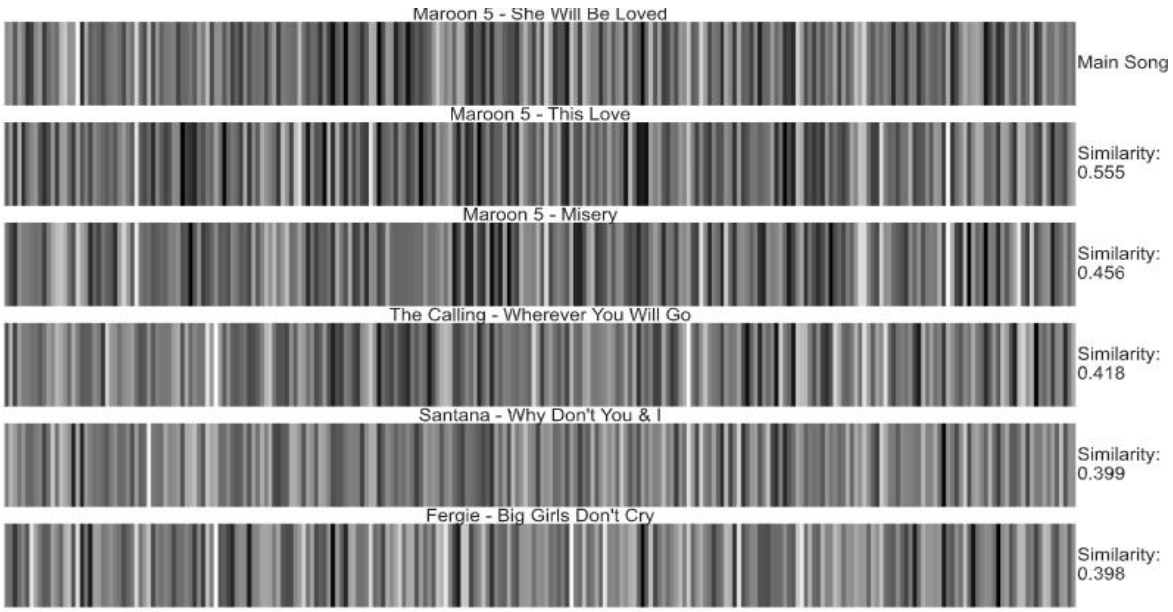

Figure 7. Visualization of song vectors

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

After embedding the words of each song in the training set, you can use these song vectors to recommend for a playlist. So how to recommend it? One way is by averaging all the song vectors in each playlist as the playback vector of the whole playlist, or as a portrait of a user in the lyrics embedded in the vector space. These vectors are used to generate queries to find similar songs based on cosine similarity.

Table 1 shows the playlist with id 305. You can see that there are three songs Selena-Como La Flor, The Texas Tornados-Who Were You Thinkin' Of, and Selena-Sentimientos in the playlist. Taking the mean vector of the three song vectors as the portrait of the user, the user is recommended. Take the top five songs with the highest similarity:

Table 1. The five songs most similar to id 305

Similarity	Song
0.763	Little Joe Y La Familia - Borrachera
0.751	Lorenzo Antonio - Con La Misma Espina
0.745	Tierra Tejana - Eres Casado
0.742	Jennifer Y Los Jetz - Me Piden
0.730	The Texas Tornados - (Hey Baby)Que Paso

4. Experiment

4.1 Process

In order to evaluate the recommendation effect of Top-n songs recommended based on context semantic relationship, through the comparative experiments of the recommendation algorithm based on context semantic relation, the recommendation algorithm based on SVM, the recommendation algorithm based on item similarity and the recommendation algorithm based on popularity, the hit rate, accuracy, recall, F1 value and running time of each al-gorithm were compared.

4.2 Results and Analysis

For Top-n recommendation, hit rate (HR) is a commonly used metrics. If the results of TOP-N recommend we previously on the test set music ID, then we think it is "hit", otherwise it is not intended. The figure below is the hit rate of TOP-25, which can be seen that the hit rate of the recommended al-gorithm based on the context-oriented semantics proposed is better than other algorithms.

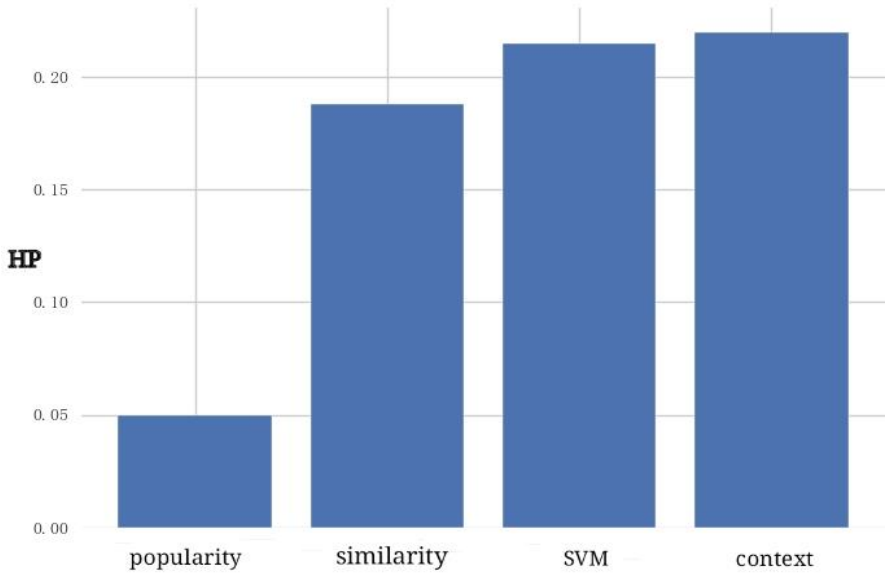

Figure 8. Hit rate comparison

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

Test the recommended algorithm based on context semantic relationships and the accuracy of other algorithms based on different n (recommended numbers). As can be seen from the following figure, the recommendation algorithm proposed in this paper has better performance performance compared to other algorithms, and with the increase of the number of recommendations, the accuracy of the algo-rithm Rendering a smooth rise.

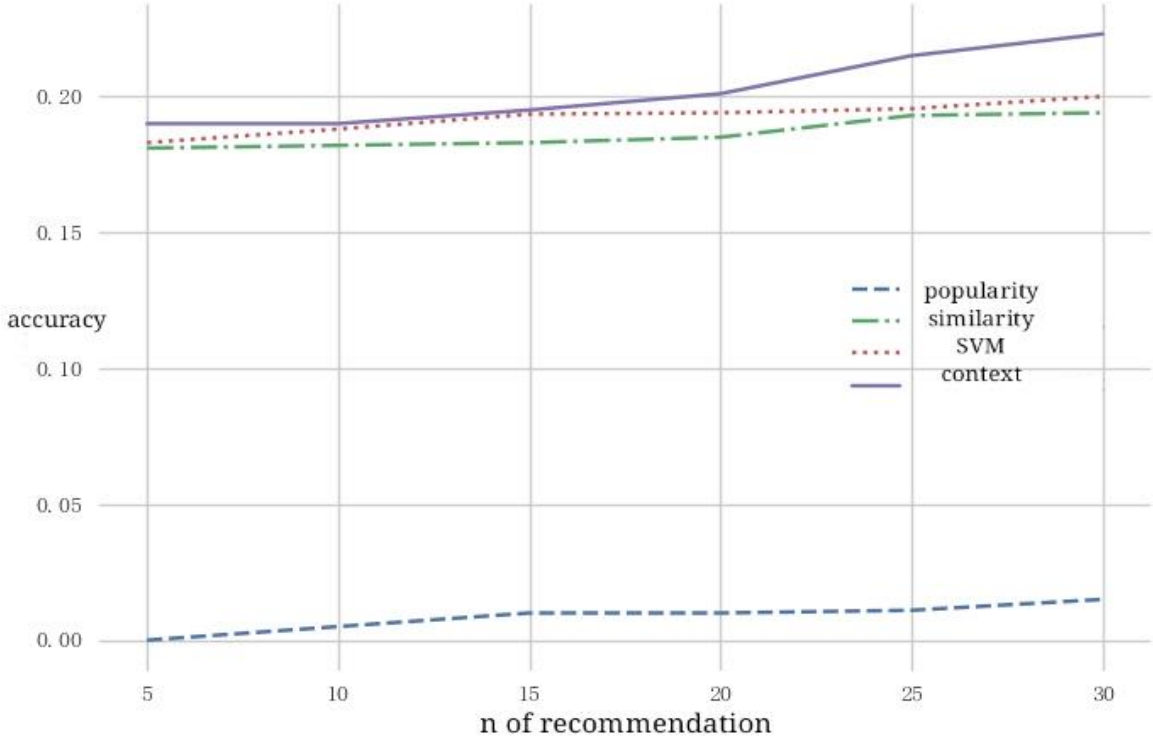

Figure 9. Comparison of accuracy

Table 2 is based on the context semantic recommendation algorithm and other algorithms in the accuracy, the recall , and the f1 value in the data set on the data set, and the algorithm runs once.

Table 2. Algorithm performance comparison

algorithm	accuracy	recall	F1	t (s)
Context	0.1983	0.1281	0.1642	5586
SVM	0.1883	0.1174	0.1597	6374
Similarity	0.1876	0.1169	0.1565	5862
popularity	0.0142	0.0096	0.0113	0.447

From the experimental data, we can see that the recommendation algorithm based on context semantics is better than other algorithms in the recommendation performance of Top-n. From the point of view of the running time of the algorithm, compared with the algorithm based on popularity, it is much more complex, because the popularity algorithm only counts the playing times of different songs, so the running time of the algorithm is equivalent to traversing the data set. So the time complexity is linear. However, the efficiency of the algorithm is not ideal. Compared with the recommendation algorithm based on SVM and item similarity with better recommendation effect, it is also more time-saving.

5. Conclusion

This paper makes a comprehensive discussion and research on the proposed recommendation algorithm based on context semantics. By capturing the context semantic relationship of song playback

ISSN: 2414-1895 DOI: 10.6919/ICJE.202107_7(7).0014

and constructing the user's spatial portrait according to the user's listening records, the user's Top-n recommendation can be realized. The experimental results show that the running time, hit rate, accuracy, recall rate and F1 value of this model are better than other algorithms.

References

[1] Zhang Xiaolei. A review of research on collaborative filtering-based recommendation systems[J]. Digital World, 2021(01): 8-9.

[2] Shi Yuan. Overview of personalized recommendation algorithms [J]. Intelligent Computers and Applications, 2020, 10(08): 110-112.

[3] Shi Fenggui. Word2Vec word vector application based on natural language processing[J]. Journal of Heihe University, 2020, 11(07): 173-177.

[4] Jia Xiaofang. Research on recommendation model based on deep learning [D]. Dalian Maritime University, 2020.

[5] AI Bi. Design and implementation of personalized music recommendation system [D]. Chengdu: University of Electronic Science and technology, 2018

[6] Yang Kai, Wang Li, Zhou Zhiping, et al. Personalized recommendation of scientific literature based on content and collaborative filtering[J].Information Technology,2019,43 (12):11-14.

[7] Ruan Guangce, Xie Fan, Tu Shiwen. Application Research on the Diversity of Library Recommendation System Based on Word2vec[J]. Library Journal, 2020, 39(03): 124-132.

[8] Zhang Huawei. Neural network collaborative recommendation model based on Word2Vec[J]. Cyberspace Security, 2019, 10(06): 25-28.

[9] Huang Ran. Research on MOOC recommendation algorithm based on content and word2vec [D]. Shandong Normal University, 2019.

[10] Wu Tao. Research and application of recommendation algorithm in recommendation system[D]. Beijing Jiaotong University, 2019.

[11] Liang Shunpan, Wang Chen, Yuan Fuyong, Zhang Fuzhi. A matrix factorization recommendation algorithm combining mean segmentation and word2vec [J]. Small Microcomputer System, 2019, 40(05): 978-983.

[12] Ye Gongbing. Research on movie recommendation system based on collaborative filtering [D]. South China University of Technology, 2019.

[13] Zhang Yu. Research and implementation of recommendation system based on user ratings and comments [D]. South China University of Technology, 2019.

[14] Wang Yu. Research on matrix completion and recommendation algorithm based on word2vec [D]. Jilin University, 2019.

[15] Gao Yang. Research and application of sentiment analysis based on Word2Vec method [D]. Xiamen University, 2019.

[16] Li Xiaoyu. Overview of collaborative filtering recommendation algorithms[J]. Journal of Shangqiu Teachers College, 2018, 34(09): 7-10.

[17] Wang Qinglong. Application research of text content monitoring and analysis based on word2vec and SVM [D]. Nanchang University, 2018.

[18] Li Xiao, Xie Hui, Li Lijie. Research on the calculation of sentence semantic similarity based on Word2vec[J]. Computer Science, 2017, 44(09): 256-260.

[19] Huang Xiaohang. Research review of recommender systems[J]. Science and Technology Getting Rich Guide, 2014(30): 206+222.