Advanced Engineering Informatics xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Applying artificial immune systems to collaborative filtering for movie recommendation

Meng-Hui Chen [a], Chin-Hung Teng [b,c], Pei-Chann Chang [a,c,*]

[a] Department of Information Management, Yuan Ze University, Chung-Li 32023, Taiwan
[b] Department of Information Communication, Yuan Ze University, Chung-Li 32023, Taiwan
[c] Innovation Center for Big Data and Digital Convergence, Yuan Ze University, Chung-Li 32023, Taiwan

ARTICLE INFO

Article history:
Received 24 August 2014
Received in revised form 16 April 2015
Accepted 20 April 2015
Available online xxxx

Keywords:
Recommendation system
Collaborative filtering
Artificial immune system

ABSTRACT

Collaborative filtering is a widely used recommendation technique and many collaborative filtering techniques have been developed, each with its own merits and drawbacks. In this study, we apply an artificial immune network to collaborative filtering for movie recommendation. We propose new formulas in calculating the affinity between an antigen and an antibody and the affinity of an antigen to an immune network. In addition, a modified similarity estimation formula based on the Pearson correlation coefficient is also developed. A series of experiments based on MovieLens and EachMovie datasets are conducted, and the results are very encouraging.

1. Introduction

Recommender systems have been an important research topic over the past twenty years. These systems learn users' preferences in order to make recommendations to them. As e-commerce continues to grow, more and more products are being purchased online, and there is an increasing customer demand for the large number of items available on websites to be filtered so that they can more easily find specific items that actually interest them. Recommender systems were developed for this purpose, and can predict user preference for an item. Companies that successfully employ recommender systems in their e-commerce business include Amazon and Netflix. They learn user needs and try to provide the necessary information to users via the recommender system, thus increasing sales and profits.

Personalized recommender systems obtain useful information from historical data, such as a user's interests and purchasing behavior, in order to recommend relevant information or products to that user. Personalized recommendation was first proposed by Robert Armstrong [1] in 1995, and in the same year Henry Lieberman [2] presented an intelligent personalized navigation system for web browsing. Yahoo announced a personalized web entry point called My Yahoo the following year, and since then many novel personalized recommendation concepts have been proposed, and many recommendation algorithms have been developed.

Recommender systems can be roughly classified into four types: content-based recommendation, knowledge-based recommendation, collaborative filtering recommendation, and hybrid recommendation. In this research, we focus on collaborative filtering recommendation because it is a widely used algorithm in this field. Collaborative filtering algorithm was introduced in the 1990s, and has been effectively used in many recommender systems [3,4]. It predicts a rating for a user based on the rating preferences of similar users. Most collaborative filtering algorithms operate by finding similar users and then predicting a rating of an item based on the preferences and previous ratings of those users.

Collaborative filtering techniques can be further categorized into two types, model-based and memory-based, depending on how the data are processed. Model-based collaborative filtering techniques aim at building a model to represent user rating data, and use that model to predict user preference for a specific item. On the other hand, memory-based algorithms employ all user rating data to predict a missed user rating of an item. Memory-based techniques can also be classified as user-based and item-based collaborative filtering. User-based collaborative filtering is the first automatic collaborative filtering method [5,6]. It works by finding other users with rating preferences similar to those of the target user, and uses their ratings to predict the target user's rating of the item in question. In contrast to user-based collaborative filtering, item-based collaborative filtering is developed from the perspective of the item. Item-based collaborative filtering was first introduced by Sarwar et al. [7], and has been used by Amazon.com [8].

The dataset applied in this research is MovieLens dataset and it was created in 1997 by the GroupLens. GroupLens is a research lab

* Corresponding author at: Department of Information Management, Yuan Ze University, Chung-Li 32023, Taiwan.
E-mail address: iepchang@saturn.yzu.edu.tw (P.-C. Chang).

http://dx.doi.org/10.1016/j.aei.2015.04.005
1474-0346/

in the Department of Computer Science and Engineering at the University of Minnesota. There are five important fields in the lab, one of those is recommendation. Liu et al. [28] proposed a collaborative filtering based recommendation system. One of the three datasets in the paper is MovieLens. Their research results show that the collaborative filtering framework established by user interest could make the forecast accuracy higher. Sarwar et al. [20] focused on the development of different correlation coefficients. The improved correlation coefficient derives the higher accuracy in MovieLen datasets than others.

Artificial immune system (AIS) is a technique that simulates the mechanism of a biological immune system fighting foreign pathogens. It has been successfully used in optimization problems and scheduling [9]. AIS has also been used in collaborative filtering for recommending items [10,11]. For example, Acilar and Arslan [10] employed AIS to solve data sparsity and scalability problems. However, as a model-based approach, it has limited accuracy; lower than some state of the art techniques.

In tradition, the user's rating in collaborative filtering is calculated based on Pearson Coefficient only. However, the predicted results of user's rating should not be confined by this rule only. In this research, a model based approach by combing AIS and collaborative filtering is proposed and applied to predict the user's rating instead. Using this model, the immune network of antibodies will be employed as a classification rule to predict the user's rating. Even the user in the same class but his rating can be calculated from different immune networks which will be more accurately to reflect the user's interest.

The remainder of this paper is organized as follows. First, we will give a simple review of related work in Section 2. Following this, Section 3 introduces our AIS algorithm for user or item classification, and describes our new AIS-based collaborative filtering system. The experiments are given and their results are discussed in Section 4. Finally, we conclude in Section 5.

2. Literature review

In this paper, we mainly focus on collaborative filtering techniques, thus detailed research work on collaborative filtering is surveyed in the following:

2.1. Collaborative filtering

A typical collaborative filtering scenario is one in which, given a set of users and a set of items, the users may rate a subset of items, while the system must predict a missing user rating for an item. Based on their information processing approaches, collaborative filtering techniques can be classified into two types: model-based and memory-based. Memory-based approaches predict the missed rating from a group of users or items with similar profiles. In this type of approaches, similarity calculation is a critical step for finding a group of similar users or items.

Well-known similarity metrics include the Pearson correlation coefficient [5], constrained Pearson [15], weighted Pearson correlation [16] and cosine similarity [17]. Although these similarity measures have been used in many collaborative filtering algorithms, some researchers are dissatisfied with their performance. As a result, novel similarity models are continuously being put forward [18,19]. For example, Liu et al. [19] proposed a similarity model taking the local context information of user ratings and the global preference of user behavior into account. They claim that this new model is more effective, especially in under cold user conditions.

Because memory-based collaborative filtering techniques achieve recommendation based on a group of similar users or items, they are also called neighbor-based methods. Neighbor-based approaches can be further classified unto two types, user-based [5,6] and item-based [7], according to their similarity calculation methods. User-based approaches filter information based on a group of similar users, while item-based approaches compute the similarity between items instead of user similarity.

In contrast to neighbor-based approaches, model-based approaches achieve item recommendation by first constructing a model and then predict user ratings based on this model. Many model-based approaches, such as SVD [20], factor analysis [21], neural networks [22], PCA [23], Bayes networks [17] and latent class models [24,25] have been proposed. Typically, model-based approaches tend to have lower prediction time than neighbor-based methods. However, many models are very complex, with large number of parameters to be estimated, and thus require a long time to learn the models. On the other hand, neighbor-based approaches are typically much simpler and easy to implement, and can produce reasonable accuracy if sufficient user rating information is available. Therefore, this research will adopt the neighbor-based approach by applying AIS in collaborative filtering for movie recommendation.

2.2. Artificial immune system

Artificial immune system is derived from the mechanism of a biological immune system fighting foreign pathogens. By using the adaptive immune response, this algorithm can be used to search for the solution to an optimization problem. Immune network theory was first proposed by Jerne [12] in 1974.

In 2003, Dasgupta et al. [13] explained that an immune system is a complex system. It has a strong information processing characteristics, such as feature selection, pattern recognition, learning and memory recall. Three major immunological principles are typically applied in an immune system. They are immune network theory, negative selection and clonal selection. In 2005, Alatas and Akin [14] proposed using an artificial immune system algorithm to mine fuzzy classification rules in order to improve classification accuracy. They used the AIS algorithm to find the best classification rule in that category, and saved it in the database to improve the classification accuracy. AIS has also been used in collaborative filtering [10,11]. As described in Section 1, its advantages lie in data reduction, and the reported accuracy cannot therefore compare with many state of the art collaborative filtering techniques.

In this research, we apply AIS in collaborative filtering for movie recommendation. The differences of our approach among other earlier approaches are 1. We propose new formulas in calculating the affinity between an antigen and an antibody and the affinity of an antigen to an immune network. This new formula can be applied either in user-based or item-based approaches. 2. A modified similarity estimation formula based on the Pearson correlation coefficient is also developed. 3. We also derive a prediction formula suitable for the resulting classification. From experimental results on the MovieLens dataset, we found that our system is able to produce prediction accuracy comparable to state of art techniques in terms of mean absolute error. In addition, the precision and recall rate of our system are also very high.

3. Proposed system

The main idea in this research is to develop a rating model by applying AIS in collaborative filtering for movie recommendation. As mentioned above, there are two types of neighbor-based collaborative filtering approaches; one is user-based [5,6], and the other is item-based [7]. The rating model can be used in both approaches. In this study, we call this approach as user- or

Please cite this article in press as: M.-H. Chen et al., Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.04.005

item-based AIS collaborative filtering system. By applying the rating model, we can predict users' rating for a movie.

As shown in Fig. 1, there are two phases in our proposed model; one is training phase, and the other is testing phase. In the training phase, we treat each record of user rating data as an antigen and the similarity is defined as the affinity in AIS. It is calculated by using the Pearson correlation coefficient. When the antigens invade the immune system, there are antibodies generated. Then, the immune network is constructed by the first K antibodies generated.

Once the initial immune network is generated, we calculate the affinity of each training data and the immune network. If the affinity is higher than the threshold, the antigen will extend the immune network. Otherwise, the antigen will form a new immune network. By the way, there will be many more immune networks generated in the training phase.

After all the immune networks are generated in the training phase, we apply collaborative filtering technique to predict a user's rating. To achieve this, we find a set of nearest neighbour to the target user and the associated immune networks. We then compute the affinities of the target user to its nearest neighbors and its associated immune networks, and then predict the rating by using the computed similarities as the prediction weighting.

In this proposed forecasting model, we defined each training data as an antigen invading the immune system. After the training phase, AIS will generate many different classification rules through the evolution process. As explained earlier, the predicted results of user's rating must not be confined by one single rule and as inspired from Willke [26] and Hermann [27], immune network and collaborative filtering were proposed and applied to calculate the affinity and predict the user's rating. In AIS, each user's rating will be calculated by the associated antibodies. In addition, each antibody has its specificity in immune network; therefore, the predicted results are not just to follow the antibody with highest affinity, but following the related antibodies in the immune network. By the way, the forecasting strategy is similar to group decision making. The user's rating is based on the group decision making instead of by one-side message only.

In the following subsections, we give the detailed procedures of our proposed model and explain how we predict user ratings for specific items based on the trained immune networks.

3.1. Generating immune networks

In this research, a record of data in the training set is treated as antigens. We will use these antigens to generate antibodies and associated immune networks. The immune network generation flowchart is depicted in Fig. 2 and explained in the following.

We first define terms used in this paper.

- Antigen: We define all unclassified training data as antigens, i.e. $AG = \{Ag_1, Ag_2, Ag_3, \ldots, Ag_v\}$, where v is the number of users (or items if item-based approach is applied) in the training data. Each antigen includes two different types of data: the basic data $F = \{f_1, f_2, f_3, \ldots, f_m\}$ and the rating data $R = \{r_1, r_2, r_3, \ldots, r_n\}$. Basic data F contain the basic information about the antigens such as age, gender and career if the antigen is a user, and movie title and movie type if the antigen is an item, such as a movie. The rating data R are simply user ratings of items.
- Antibody: Antibodies are generated in the training process to react to antigens. In this study, antibodies are generated by copying antigens. However, unlike antigens, antibodies are organized in networks called immune networks. Each antibody therefore has its associated immune network.
- Immune Network: When an antigen enters a human body, an immune response will be triggered and an immune network will be generated. The antibodies in an immune network can unite to fight off antigens. In this paper, the immune networks are denoted by $N = \{n_1, n_2, n_3, \ldots, n_k\}$, where each network $n_k = \{Ab_1, Ab_2, Ab_3, \ldots, Ab_s\}$ is organized by a set of antibodies.

We now describe how the immune networks are generated and how the affinities between antibodies and antigens are computed.

3.2. Training immune networks

As shown in Fig. 2, in order to generate entire immune networks we must first produce several initial networks. This simulates the human immune response, which has some inherent antibodies belonging to different immune networks before the invasion of antigens. We first randomly select an antigen to create the first immune network. Later, we repeat selecting the antigens which produce the worst affinity to existing networks in order to generate other initial networks until a fixed number of networks are created. This can prevent the generated networks from being created by similar antigens, which may lead to training failure due to insufficient immune ability. The steps for generating initial immune networks are summarized in the following.

Step 1: Randomly select an Ag_ξ from AG and produce the first immune network n_1 by copying Ag_ξ as Ab_1, the first antibody of n_1.

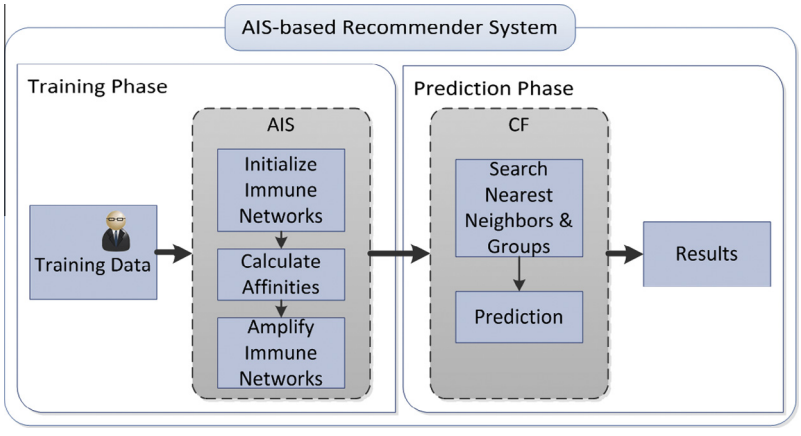


Fig. 1. The proposed AIS-based recommender system framework.

Please cite this article in press as: M.-H. Chen et al., Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.04.005

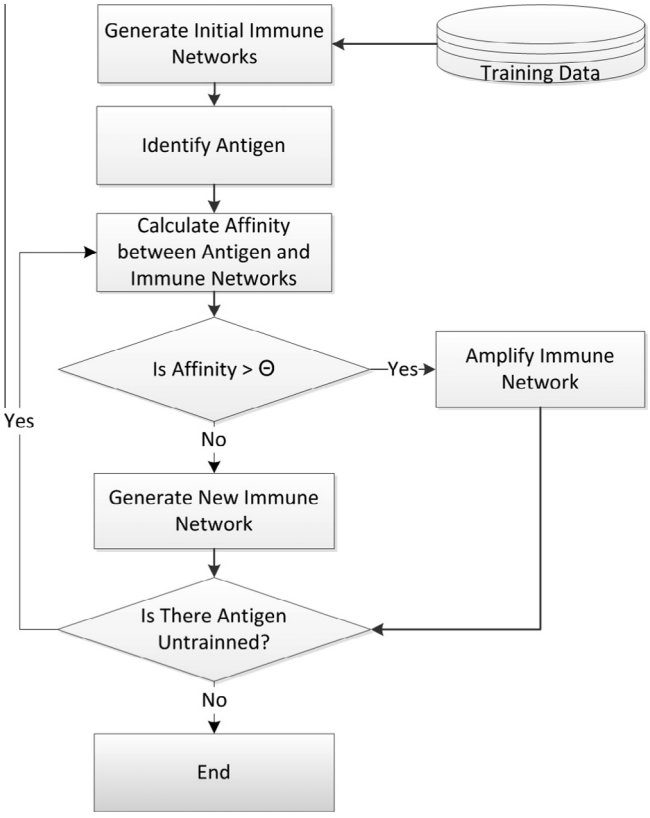


Fig. 2. Flowchart for generating immune networks.

Step 2: Compute the affinities of all antigens Ag_v to the generated immune networks.
Step 3: Find the Ag_v with the worst affinity to the immune networks. Produce a new immune network n_k by copying Ag_v as Ab_1 and add Ab_1 to n_k.
Step 4: If the number of immune networks (denoted by $|N|$) is smaller than a predefined number such as L, repeat Steps 2 and 3 until L initial immune networks are produced.

Fig. 3 shows the pseudo code for generating the initial immune networks.

Once the initial immune networks have been generated, we then evolve the networks by including the remaining antigens in the system so as to expand the immune networks or produce other new networks to increase the immune capability of the system, as shown in Fig. 2. A threshold θ is applied to decide if the next antibody is to join the current immune network or not. If the similarity between the antibody and the center of the immune network is greater than θ, the new antibody will be added into this immune network. Because there are a significant number of combinations of threshold, we will apply a greedy approach to determine the value of θ. In this paper, the threshold value is decided according to the experimental results by set the value from 0.1 to 0.9.

The steps for producing the final immune networks are summarized in the following.

Step 1: For each antigen Ag_v, compute its affinities to all immune networks.
Step 2: Find the immune network n_k with the best affinity with Ag_v.
Step 3: If the best affinity is greater than a threshold θ, copy Ag_v as Ab_s and add Ab_s to n_k. Otherwise, create a new network and copy Ag_v to this new network as an antibody.

```
1.  Input: antigens AG and the desired number of initial immune networks L
2.  Output: initial immune networks N={n1,…, nL}
3.  Start
4.      ξ ←random number
5.      Generate n1, copy Agξ as Ab1, and add Ab1 to n1
6.      Whlie |N| < L Do
7.          For each antigen Agv Do
8.              For each nk Do
9.                  Calculate Affinity(Agv,nk)
10.             End For
11.         End For
12.         Find the Agv with the smallest affinity
13.         Generate new nk, copy Agv as Ab1 , and add Ab1 to nk
14.     End While
15. End
```

Fig. 3. Pseudo code for generating initial immune networks.

Step 4: Repeat Steps 1 to 3 until all antigens are processed.

The pseudo code for producing the final immune networks is shown in Fig. 4.

3.3. Calculating affinity

Affinity calculation plays a critical role in our immune network generation. In this research, affinity is merely the similarity between antigen and antibody. As mentioned above, each antigen has two types of data, the basic data F and the rating data R. According to the data characteristics, we introduce different formulas for calculating the affinities of these two types of data.

Basic data F: Basic data is a type of nominal data, such as the gender of a user, and the affinity of this type of data cannot be evaluated using traditional similarity measures such as Pearson correlation coefficient. For this type of data, we develop a new formula by employing the Hamming distance [29] as our similarity calculation.

The function h is the Hamming distance measure between the vth antigen and the xth antibody in component f_i. It is defined as follows:

$$h(Ag_{vf_i}, Ab_{xf_i}) = \begin{cases} 1, & \text{if } Ag_{vf_i} = Ab_{xf_i} \\ 0, & \text{otherwise} \end{cases}. \quad (1)$$

where Ag_v and Ab_x denote the vth antigen and the xth antibody, respectively, subscript f_i represents the ith component of the basic data associated with that antigen or antibody.

```
1.  Input: antigens AG, initial immune networks N, and threshold θ
2.  Output: final immune networks N={n1,…, nM}
3.  Start
4.      For each antigen Agv Do
5.          For each nk Do
6.              Calculate Affinity(Agv,nk)
7.          End For
8.          Find the nk with the best affinity
9.          If (Affinity(Agv,nk) > θ ) Then
10.             Copy Agv as Abs and add Abs to nk
11.         Else
12.             Generate new nk, copy Agv as Abs , and add Abs to nk
13.         End If
14.     End For
```

Fig. 4. Pseudo code for producing final immune networks.

Then, the affinity between an antigen, Ag_v and an antibody, Ab_x is calculated as follows:

$$H(Ag_{v,F}, Ab_{x,F}) = \frac{\sum_{i=1}^{m} h(Ag_{v,f_i}, Ab_{x,f_i})}{m}, \quad (2)$$

m is the total number of components in the set F.

Finally, the affinity of an antigen Ag_v to an immune network n_k, is evaluated using the average Hamming distance:

$$Affinity(Ag_v, n_k) = \frac{\sum_{i=1}^{s} H(Ag_{v,F}, Ab_{i,F})}{s}, \quad (3)$$

where s is the number of antibodies in the immune network n_k.

To demonstrate the feasibility of this new formulation, a simple case including three users watching 10 different movies is shown in Table 1. Assume that User1 have seen all these 10 movies while User2 only two and User3 seven. The rating data for each movie by these three users is given in Table 1.

Each user and each item are considered as different immune networks (n_k). The user-based approach following Eq. (1), the affinity between User2 and User1, is calculated as follows:

$$H(u_2, u_1) = \frac{0+0}{2} = 0$$

The average of affinity between User2 and User1 (n_{k_1}), User2 and User3 (n_{k_3}) is as follows:

$$Affinity(u_2, n_k) = \frac{\left(\frac{0+0}{2}\right) + \left(\frac{0}{1}\right)}{2} = 0, n_k = n_{k_1}, n_{k_3}$$

The item-based approach following Eq. (1), the Hamming distance between Item3 and Item1 is as follows:

$$H(i_1, i_3) = \frac{0+0}{2} = 0$$

The average of affinity between Item2 and Item1 (n_{k_1}), Item2 and Item3 (n_{k_3}) is as following:

$$Affinity(i_2, n_k) = \frac{\left(\frac{0}{1}\right) + \left(\frac{0}{1}\right)}{2} = 0, n_k = n_{k_1}, n_{k_3}$$

Rating data R: For rating data we can apply the widely-used Pearson correlation coefficient [30] as the similarity measure, as follows:

$$P(Ag_{v,R}, Ab_{x,R}) = \frac{\sum_{i\in I}(Ag_{v,r_i} - \overline{Ag_{v,I}})(Ab_{x,r_i} - \overline{Ab_{x,I}})}{\sqrt{\sum_{i\in I}(Ag_{v,r_i} - \overline{Ag_{v,I}})^2}\sqrt{\sum_{i\in I}(Ab_{x,r_i} - \overline{Ab_{x,I}})^2}} \quad (4)$$

where I is the intersection set of the rating data R of Ag_v and Ab_x, and subscript r_i denotes the ith rating data in R. $\overline{Ag_{v,I}}$ and $\overline{Ab_{x,I}}$ are the means of the rating data of antigen Ag_v and antibody Ab_x on the set I, respectively. According to Table 1, the user-based similarity of User2 and User1 is shown as following:

$$P(u_2, u_1) = \frac{((5-4.5)*(4-3.5)) + ((4-4.5)*(3-3.5))}{\sqrt{((5-4.5)^2 + (4-4.5)^2)} + \sqrt{((4-3.5)^2 + (3-3.5)^2)}} = \frac{0.5}{0.5} = 1$$

The item-based similarity of Item3 and Item1 is shown as following:

$$P(i_1, i_3) = \frac{((5-4.5)*(4-3.5)) + ((4-4.5)*(3-3.5))}{\sqrt{((5-4.5)^2 + (4-4.5)^2)} + \sqrt{((4-3.5)^2 + (3-3.5)^2)}} = \frac{0.5}{0.5} = 1$$

The Pearson correlation coefficient is similar to treat the data as a vector and calculate the correlation in a normalized form. However, we found that its capability in determining the affinity of two user rating data is limited. For example, following the rating data in Table 1, the Pearson correlation coefficients for User1 and User2 and for User1 and User3 are 1 and 0.7454, respectively. However, User1 and User2 only have the ratings for 2 common items while User1 and User3 have ratings for 7 common items with 3 identical rating values. Obviously, User1 and User3 are more similar than User1 and User2, but the Pearson correlation coefficient does not reflect this phenomenon.

To address this problem, we develop a modified Pearson correlation coefficient by taking previous points into account as follows:

$$P_{revised}(Ag_{v,R}, Ab_{x,R}) = w\frac{\sum_{i\in I}(Ag_{v,r_i} - \overline{Ag_{v,I}})(Ab_{x,r_i} - \overline{Ab_{x,I}})}{\sqrt{\sum_{i\in I}(Ag_{v,r_i} - \overline{Ag_{v,I}})^2}\sqrt{\sum_{i\in I}(Ab_{x,r_i} - \overline{Ab_{x,I}})^2}}, \quad (5)$$

where w is defined as the number of the movies with the same rating between user A and user B, i.e.

$$w = \sum_{i\in I}\delta(Ag_{v,r_i}, Ab_{x,r_i}) \quad (6)$$

where

$$\delta(a,b) = \begin{cases} 1, & \text{if } a = b \\ 0, & \text{if } a \neq b \end{cases} \quad (7)$$

However, if w is 0, w is reset to its default value 1 instead. That is the original Pearson Coefficient will be employed in this case.

Following the case in Table 1, the number of identical ratings of items between User1 and User3 is 3, and w is equal to 3. The number of identical ratings of items between User1 and User2 is 0, and w is set to 1 as the default value of w is 1. Owing to this weight difference, i.e., 3 and 1, the affinity for User1 & User3, is higher than User1 & User2. If not considering this weight, affinity of User1 & User2 will be higher than User 1 & User3.

The w for User1 and User2 is 1, thus, the user-based approach in Eq. (5) is as follows:

$$P_{revised}(u_1, u_2) = 1*\frac{((5-4.5)*(4-3.5)) + ((4-4.5)*(3-3.5))}{\sqrt{((5-4.5)^2 + (4-4.5)^2)} + \sqrt{((4-3.5)^2 + (3-3.5)^2)}} = \frac{0.5}{0.5} = 1$$

The w for Item1 and Item3 is 1, therefore the item-based approach in Eq. (5) is as follows:

$$P_{revised}(i_1, i_3) = 1*\frac{((5-4.5)*(4-3.5)) + ((4-4.5)*(3-3.5))}{\sqrt{((5-4.5)^2 + (4-4.5)^2)} + \sqrt{((4-3.5)^2 + (3-3.5)^2)}} = \frac{0.5}{0.5} = 1$$

In generating immune networks, we must compute the affinity of an antigen to an immune network. In this case, the affinity is

Table 1
A simple case including three users watching ten movies.

	Item1	Item2	Item3	Item4	Item5	Item6	Item7	Item8	Item9	Item10
User1	5	2	4	4	1	4	3	4	5	2
User2	4	–	3	–	–	–	–	–	–	–
User3	5	–	–	4	1	3	2	1	4	–

Please cite this article in press as: M.-H. Chen et al., Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.04.005

computed by averaging the revised Pearson correlation coefficients on the network, as follows:

$$Affinity(Ag_v, n_k) = \frac{\sum_{i=1}^{s} P_{revised}(Ag_{v,R}, Ab_{i,R})}{s}, \quad (8)$$

where s is the number of antibodies in the immune network n_k.

Following Table 1, for the user-based approach in Eq. (8), the average of affinity between User2 and User1 (n_{k_1}), User2 and User3 (n_{k_3}) is as follows:

$$Affinity(u_2, n_k) = \frac{1 * \frac{((5-4.5)*(4-3.5))+((4-4.5)*(3-3.5))}{\sqrt{((5-4.5)^2+(4-4.5)^2)}+\sqrt{((4-3.5)^2+(3-3.5)^2)}} + 1 * 0}{2}$$
$$= \frac{1 * \frac{0.5}{0.5} + 1 * 0}{2} = 0.5, n_k = n_{k_1}, n_{k_3}$$

For item-based approach, the average of affinity between Item2 and Item1 (n_{k_1}), Item2 and Item3 (n_{k_3}) is as follows:

$$Affinity(i_2, n_k) = \frac{1 * 0 + 1 * 0}{2} = 0, n_k = n_{k_1}, n_{k_3}$$

3.4. Predicting user ratings

Once the immune networks have been generated, we then use these networks to predict a user rating for a specific item. Because the immune networks are trained by all the antigens, we know to which network each antigen (or user) belongs. We let $G = \{g_1, g_2, g_3, \ldots, g_k\}$ denote the set of trained immune networks with each $g_k = \{u_1, u_2, u_3, \ldots, u_s\}$ consisting of a set of users. A threshold θ' is applied to decide if the antibody u_x in immune network g_i is to join the current immune network g_j or not. θ'' is the threshold value which decides if the immune network g_i is to join the current immune network g_j or not. Again, these two thresholds are decided by a greedy approach. In this paper, the threshold value is decided according to the experimental result by set the value from 0.1 to 0.9.

The prediction process first finds a set of the target user's nearest neighbors, based on the immune networks, and then predicts the rating using the similarities between groups and users. We summarize the procedures in the following.

Step 1: For each group g_k, compute the similarity between the group of the target user u_v to g_k.
Step 2: If the similarity is greater than a predefined threshold θ'', then for each user u_s in this group, compute the similarity between u_s and u_v.
Step 3: If the similarity between u_s and u_v is greater than a predefined threshold θ', include the user u_s in the set U.
Step 4: Repeat Steps 1 to 3 until all qualified users are found.
Step 5: Predict user's rating of an item based on the set U.

The pseudo code for searching the nearest neighbors is shown in Fig. 5.

In this study, the process of prediction is classified into two cases.

Case 1: If the set U contains the users who have rated the item that target user u_v wants to rate.

In this case, the prediction is achieved by the following formula:

$$P_{u_v,r} = \overline{u_v} + \frac{\sum_{i=1}^{U} GSim_{revised}(u_{v,g}, u_{i,g}) \times Sim_{revised}(u_v, u_i) \times (u_{i,r} - \overline{u_i})}{\sum_{i=1}^{U} |GSim_{revised}(u_{v,g}, u_{i,g}) \times Sim_{revised}(u_v, u_i)|} \quad (9)$$

where $\overline{u_v}$ and $\overline{u_i}$ are the average ratings of user u_v and u_i, respectively. $GSim_{revised}$ and $Sim_{revised}$ are the group similarity and user similarity, respectively, based on Eq. (5). $u_{v,g}$ and $u_{i,g}$ denote the groups to which u_v and u_i belong, and $u_{i,r}$ is the rating of u_i for the item to be rated by user u_v.

```
1.  Input: user u_v, item i_k, immune networks N, threshold of user
    similarity θ', threshold of group similarity θ"
2.  Output: rating prediction of user u_v on item i_k
3.  Define U=Ø
4.  Start
5.      For each g_k Do
6.          If (group similarity(u_{v,g},g_k)>θ") Then
7.              For each u_s Do
8.                  If (similarity(u_v,u_s)>θ') Then
9.                      U←U ∪ u_s
10.                 End If
11.             End For
12.         End If
13.     End For
14.     Prediction(u_v, i_k, U)
```

Fig. 5. The pseudo code for searching the nearest neighbors.

Thus, if we want to predict the rating in Item2 of User2 in user-based approach and there are two different immune networks. The rating can be calculated as following:

$$P(u2, i2) = 3.5 + \frac{(1 * 1 * 2 - 3.4)}{|1 * 1|} = 2.1$$

Before making an example for item-based approach, we set the Item1 and Item3 as the members in the same group. The group is with high similarity to Item3, so the rating in Item3 of User3 is calculated as following:

$$P(i3, u3) = 3.5 + \frac{(1 * 1 * 5 - 4.7)}{|1 * 1|} = 3.8$$

Case 2: If the set U does not contain the users who have rated the item that target user u_v wants to rate.In this case, we have no information about the target item. Thus, we simply use the average rating of user u_v as the predicting value, i.e.:

$$P_{u_v,r} = \overline{u_v} \quad (10)$$

4. Experiments

4.1. Datasets and error metrics

In this research, we employ datasets from MovieLens (http://movielens.org) and EachMovie to evaluate the performance of the proposed system. MovieLens contains three datasets, the 100 K, 1 M and 10 M datasets, and the data sparsities for these datasets are 93.69%, 95.78% and 99.87%, respectively. Because the 100 K dataset is widely used in many research papers, we also use this dataset to test the performance of our system. The 100 K dataset contains 100,000 ratings collected from 943 users on 1682 movies. The rating value for this dataset is ranged from 1 to 5, with 5 denoting the most satisfactory rating. In addition to the rating data, this dataset also contains information about the users and movies, such as age, gender, career and movie types. We use these data as the basic data of antigens, as described in Section 3.

The MovieLens 100 K dataset has been divided into 5 folds for performance evaluation purposes. Typically, one fold is used as the testing set and the remaining folds are treated as the training

sets. Performance evaluation is achieved by computing some error metrics on the testing set until all 5 folds are tested.

The EachMovie dataset is also a commonly used dataset. However, the dataset is no longer available, and we had to use previously obtained data for this test. Because the original dataset has some jump in user and movie IDs, we have re-organized this dataset. Currently, the dataset we used has 2,811,983 ratings from 61,265 users on 1623 movies. The rating value in this dataset is ranged from 0 to 1, namely 0.0, 0.2, 0.4, 0.6, 0.8 and 1.0. Because this dataset only contains the user ratings for the movies, we do not have the basic data for AIS training. We thus only use the rating data to construct the AIS immune networks for this dataset.

The first error metric we used is the mean absolute error (MAE) [31], defined as follows:

$$MAE = \frac{\sum_{u_i,m} |R_{u_i,m} - P_{u_i,m}|}{N} \tag{11}$$

where $R_{u_i,m}$ and $P_{u_i,m}$ are the real and predicted ratings of user u_i for item m, respectively. MAE is widely used in recommender systems, and it is able to estimate the accuracy of rating prediction on average.

In addition to MAE, another widely used error measure in this field, precision and recall, is used in this study. To define precision and recall, we should first define which rating value indicates that an item should be recommended to a user. In this research, if the rating value is between 3 and 5, the item is recommended to the user; otherwise, the item is not recommended. Based on this definition, we have four different conditions: true positive (TP), false negative (FN), false positive (FP) and true negative (TN). True positive indicates that the real rating is to recommend the item, and that the predicted rating also supports this recommendation. If the predicted rating does not support this recommendation, the condition of false negative occurs. On the other hand, false positive indicates that the real rating does not support the recommendation, but the predicted rating supports the recommendation. Finally, if both real and predicted ratings do not support the recommendation, true negative occurs. According to these four conditions, the precision and recall [32] are defined as follows:

$$precision = \frac{TP}{TP + FP} \tag{12}$$

$$recall = \frac{TP}{TP + FN} \tag{13}$$

In addition to precision and recall, another widely used performance index, the F_1 measure [32], is also applied in this research to evaluate the accuracy of the precision-recall of the proposed system.

$$F_1 = 2\frac{precision \times recall}{precision + recall} = \frac{2TP}{2TP + FP + FN} \tag{14}$$

A higher value of F_1 indicates better system accuracy.

Table 2
Precision and recall under different θ.

	Consider the only target network		Consider other networks	
	User-based	Item-based	User-based	Item-based
θ	0.2	0.5	0.2	0.4
TP	79,652	79,682	79,947	80,219
FP	2868	2838	2573	2301
FN	12,925	13,069	13,081	13,387
TN	4555	4411	4399	4093
Precision	0.965245	0.965608	0.96882	0.972116
Recall	0.860386	0.859096	0.85939	0.856986
F_1	0.909804	0.909243	0.91083	0.910927

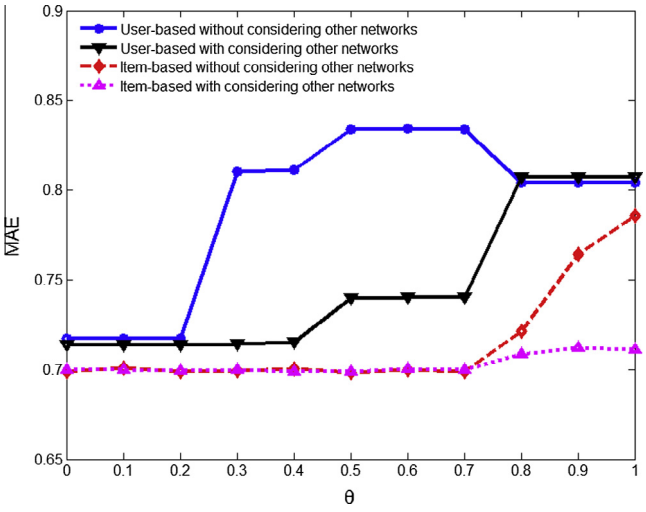
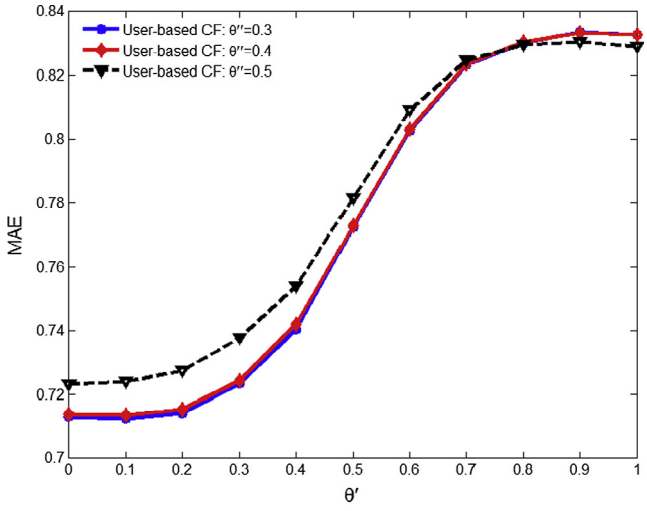
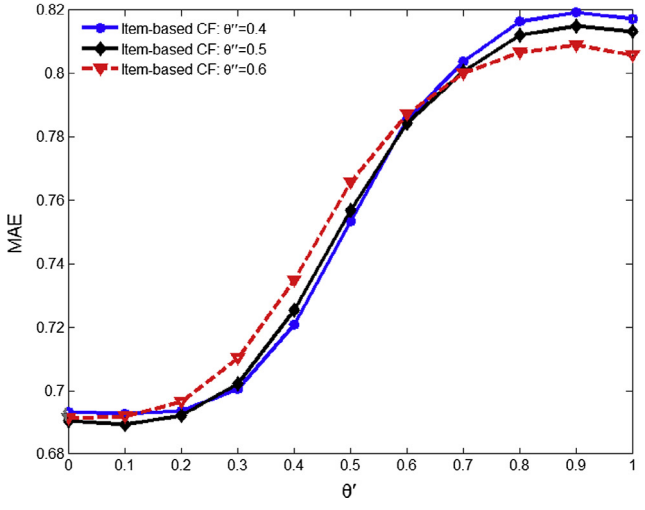


Fig. 6. MAE for different approaches under different θ.



(a) User-based AISCF



(b) Item-based AISCF

Fig. 7. MAE for our user- and item-based AIS collaborative filtering (CF).

Table 3
MAE comparison under applying different similarity measure formulas.

Based	Equation	MAE					
		Fold1	Fold2	Fold3	Fold4	Fold5	Mean
User-based	(4)	0.7393	0.7319	0.7188	0.7185	0.7283	0.7273
	(5)	0.7237	0.7169	0.7063	0.7058	0.7144	0.7134
Item-based	(4)	0.7224	0.7168	0.7184	0.7106	0.7203	0.7177
	(5)	0.7066	0.6961	0.6957	0.6936	0.7013	0.6986

Table 4
Comparison of AIS classification and *K*-means algorithm.

Based	Clustering	MAE					
		Fold1	Fold2	Fold3	Fold4	Fold5	Mean
User-based	AIS	0.7237	0.7169	0.7063	0.7058	0.7144	0.7134[a]
	K-mean	0.7218	0.7169	0.7122	0.7133	0.7096	0.7148
Item-based	AIS	0.7066	0.6961	0.6957	0.6936	0.7013	0.6986[a]
	K-mean	0.7405	0.7203	0.7134	0.717	0.7189	0.722

Note: The best performance.

4.2. Setting experimental parameters

Before applying our system in real datasets, several parameters need to be setup. The first parameter is the number of initial immune networks, L (see Fig. 3). Because our system generates new immune networks during its training process (see Fig. 4), L is not critical, and we empirically set it to 3.

The other parameters to be setup are the thresholds θ, θ', and θ''. As shown in Figs. 4 and 5, these three parameters are applied for the expansion of an immune network. AIS collaborative filtering system can be applied either in user-based approach or item-based approach. Therefore, in the following, we will show both the results of applying our AIS techniques and prediction formulas in user- and item-based system for comparison.

Fig. 6 shows the MAE of our user- and item-based AIS collaborative filtering system, with θ ranged from 0 to 1 using the MovieLens 100 k dataset. In this experiment, we also consider situations when we only use users (or items) in the same immune network to construct the neighbor set (U in Fig. 5), or take users (or items) in the other networks into consideration to construct the set. From the results of this experiment, we can obtain two results. First, in most cases the MAEs obtained when considering other immune networks are better than those obtained when only using the target network (i.e., the network containing the target user or item) to construct the neighbor set. Second, if we use other immune networks to construct the neighbor set, the θs of the optimal MAE are located at 0.2 and 0.4 for user- and item-based AIS collaborative filtering system, respectively.

Table 2 displays the precision, recall and F1 measure of our user- and item-based AIS collaborative filtering system at the θ of the optimal MAE. We also show the results obtained when considering only the target immune network, and considering other networks for comparison. This table again confirms that using the users (or items) in other networks produces better results. The F1 measures for both user- and item-based AIS collaborative filtering system when considering other immune networks are higher than those obtained when only considering the target network. Thus, in the following experiments we will employ users (or items) in other immune networks to construct the neighbor set, and the thresholds θ for user- and item-based AIS collaborative filtering system are set to 0.2 and 0.4, respectively.

When we take the users (or items) in other networks into consideration, two thresholds, θ' and θ'', must be determined (see Fig. 5). We used many combinations of θ' and θ'', and tried to find the values that led to the best MAE. Fig. 7 shows the curves that produced the best MAE for our user- and item-based AIS collaborative filtering system. From this figure we can observe that, for the user-based AIS collaborative filtering system, the best MAE occurs at $\theta' = 0.1$ and $\theta'' = 0.3$. For the item-based AIS collaborative filtering system, the best MAE occurs at $\theta' = 0.1$ and $\theta'' = 0.5$. Therefore, in the following experiments, we used these threshold values to set up our AIS collaborative filtering systems.

Table 5
Compare with other approaches [25].

	Fold1	Fold2	Fold3	Fold4	Fold5	Mean
Proposed approach	0.7002	0.6805	0.6909	0.6868	0.6881	0.6893
Pearson(all)	0.7225	0.7133	0.7062	0.7063	0.7130	0.7122
Euclidean(all)	0.7306	0.7195	0.7181	0.7210	0.7211	0.7220
Pearson(10)	0.7367	0.7297	0.7230	0.7270	0.7311	0.7295
Euclid(10)	0.7532	0.7354	0.7410	0.7448	0.7488	0.7446
Pearson(25)	0.7185	0.7071	0.7065	0.6998	0.7082	0.7080
Euclidean(25)	0.7306	0.7192	0.7237	0.7213	0.7272	0.7244
Pearson(50)	0.7157	0.7049	0.7133	0.7107	0.7102	0.7110
Euclidean(50)	0.7373	0.7314	0.7315	0.7335	0.7305	0.7328
Pearson(75)	0.7140	0.7002	0.7027	0.6982	0.7043	0.7039
Euclidean(75)	0.7260	0.7147	0.7157	0.7160	0.7205	0.7185
DM	0.7580	0.7418	0.7284	0.7509	0.7497	0.7458
ML + IMDB(EQ)	0.7304	0.7206	0.7069	0.7201	0.7209	0.7198
Triadic	0.7500	0.7369	0.7306	0.7328	0.7324	0.7365
FA/U($q = 1$)	0.7324	0.7280	0.7257	0.7279	0.7208	0.7269
FA/I($q = 1$)	0.8048	0.8051	0.8039	0.8000	0.8067	0.8041
SVD($q = 5$, $\lambda = 0$)	0.7005	0.6909	0.6971	0.6918	0.6992	0.6959
SVD($q = 4$, $\lambda = 0.01$)	0.6987	0.6876	0.6899	0.6893	0.6926	0.6916
CF($\|U_p\| = 1$, $\|M_i\| = 2$, $\tau = 1/25$)	0.6837	0.6869	0.6846	0.6861	0.6826	0.6848

Table 6
The best precision and recall of proposed approach.

TP	FP	FN	TN	Precision	Recall	F_1
79501	3019	12408	5072	0.96341	0.865	0.91156

4.3. Experimental results

This section shows the results of applying our AIS collaborative filtering system using the thresholds discussed above. We first compare the MAE of employing different similarity formulas, i.e., the traditional Pearson correlation coefficient (Eq. (4)) and the

Table 7
Results of EachMovie dataset.

	Fold1	Fold2	Fold3	Fold4	Fold5	Mean
	0.182053	0.181526	0.182025	0.182005	0.18165	0.181852
TP	FP	FN	TN	Precision	Recall	F_1
1,725,748	248,844	347,845	48,9546	0.873977	0.83225	0.852603

Table 8
Performance comparison with other algorithms [28] based on rating results.

Split./Alg	RSVD	ItemRank	CF	LDA	iExpand	AIS + CF
10–90	0.887	0.845	0.919	0.909	**0.844**	0.846
20–80	0.798	0.796	0.822	0.825	**0.795**	0.840
30–70	**0.770**	0.775	0.787	0.792	0.777	0.787
40–60	0.760	0.749	0.772	0.778	0.769	**0.743**
50–50	0.751	0.755	0.756	0.767	0.759	**0.714**
60–40	0.748	0.754	0.751	0.765	0.759	**0.703**
70–30	0.747	0.748	0.744	0.758	0.753	**0.696**
80–20	0.740	0.749	0.741	0.759	0.755	**0.682**
90–10	0.739	0.757	0.746	0.764	0.763	**0.680**

Bold values denotes the best performance among all different methods.

revised Pearson correlation coefficient (Eq. (5)). Table 3 lists the MAE of applying different similarity equations in our user- and item-based AIS collaborative filtering system. We used the MovieLens 100 k dataset, and the MAEs for the 5 folds and the mean MAE are also listed in the table. From the table, we can observe that for the user-based AIS collaborative filtering system, Eq. (5), the revised Pearson correlation coefficient, is slightly better than Eq. (4), the traditional Pearson correlation coefficient. For the item-based AIS collaborative filtering system, a similar conclusion can be drawn. This confirms that our modified Pearson correlation coefficient can indeed produce better movie prediction in terms of mean absolute error.

Since our AIS collaborative filtering system can be treated as a classification approach, we compare our performance with a widely used clustering algorithm, the *k*-means algorithm. The cluster number of *k*-means is 4. The parameter is decided by running *K*-means from 2 to 10, and the best is 4. Table 4 shows the MAE obtained when applying our AIS approach and the *k*-means algorithm in user- and item-based AIS collaborative filtering system. For the user-based AIS collaborative filtering system, the difference in performance of our AIS technique over the *k*-means algorithm is not evident. However, for the item-based AIS collaborative filtering system, the superior results obtained using our AIS collaborative filtering system is obvious. This experiment demonstrates the effectiveness of the proposed AIS collaborative filtering system approach compared with a traditional classification algorithm.

To further demonstrate the performance of our proposed system, we compared it with some existing techniques. Table 5 shows the MAEs of many reported collaborative filtering techniques. Again, the MovieLens 100 k dataset was used in this evaluation. From this table we can see that, with the exception of the final approach, a latent model [25], our system's MAE is superior to those of other approaches. According to Table 2 and Fig. 7, the parameter of θ is set 0.4, θ' is set 0.1 and θ'' is set 0.3. Compared with the best approach, in Table 5, the difference in MAE is only 0.0045. Therefore, the performance of the proposed system is comparable with state of the art approaches in terms of MAE.

Table 6 lists the precision, recall and F_1 measure of the proposed system. The precision is 0.96341, and the F_1 reaches a very high value of 0.91156. This confirms the good performance of the proposed system in terms of precision and recall.

We also used the EachMovie dataset to test our system. Because the original dataset has no partition, we partitioned it into 5 folds as the MovieLens dataset, and evaluated the performance using an approach similar to the above. The MAE, precision, recall and F_1 measure results are shown in Table 7. Because the rating in the EachMovie dataset is ranged from 0 to 1, the MAE value is much smaller than that of the MovieLens dataset. The average MAE is 0.181852, and the precision, recall, and F_1 are about 0.85. The precision, recall, and F_1 are also very high, although slightly worse than that of the MovieLens dataset.

We have further compared our proposed model with other approaches [28] by different sparsity levels. The parameter of AIS collaborative filtering is the same as those applied in Table 5. The training sets are from 10% to 90% and the test sets are from 90% to 10%. The results show the performance of our proposed approach is better than others. By increasing the training sets from 10% to 90% in our model, there are more and more antibodies to be accommodated in the immune network. Therefore, the forecasted rating will be more accurate. As a result, the performance measure of our model is getting better and better as shown in Table 8.

5. Conclusion

In this paper, we presented an AIS collaborative filtering system for rating prediction and recommendation. We employed an artificial immune algorithm to train a set of immune networks. The rating data was treated as antigens, and a number of immune networks were generated by copying the antigens as the antibodies of the immune networks. These immune networks were then used as the basis for finding the nearest neighbors for a target user or item. A revised Pearson correlation coefficient was also introduced in this paper, and its effectiveness was confirmed experimentally. A prediction formula based on the generated immune networks was also devised, and the performance of our AIS collaborative filtering system using this prediction formula was evaluated. The results are encouraging, as the performance of our system is comparable to some state of the art techniques in terms of mean absolute error. In addition to mean absolute error, the precision and recall of our system on some well known datasets was also evaluated. Our system produces very high precision and recall for these datasets. Thus, if the movie company can understand or predict what movie the customers need in advance, the company can adopt more effective marketing strategy to the customers.

Although our system was tested on movie datasets, it can easily be applied in other datasets. As a memory-based collaborative

Please cite this article in press as: M.-H. Chen et al., Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.04.005

filtering technique, our system still suffers some memory-based approach problems, such as cold start and data scalability. In addition, the recommendation systems developed can be used by E-commerce sites to suggest products to their customers and to provide consumers with information to help them decide which products to purchase. Our future work will be to improve the system, to make it less sensitive to these issues. For the overspecialization problem, the user's rating will be biased by certain groups of users. Actually, by combining immune network and Pearson Coefficient our proposed model has the advantage in avoiding this problem since the user's rating is calculated based on the immune network. In the future research, to solve the cold-start problem, we could develop a mechanism to accommodate more features from the user such as "age" and "profession" to search for related users in the same group for product rating.

References

[1] R. Armstrong, D. Freitag, T. Joachims, T. Mitchell, WebWatcher: a learning apprentice for the world wide web, AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, 1995, p. 6–12.
[2] H. Lieberman, Letizia: an agent that assists web browsing, Joint Conference on Artificial Intelligence, 1995, p. 924–929.
[3] F. Cacheda, V. Carneiro, D. Fernandez, V. Formoso, Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender systems, ACM Trans. Web 5 (1) (2011). article 2..
[4] J. Bobadilla, F. Ortega, A. Hernando, J. Bernal, Generalization of recommender systems: collaborative filtering extended to groups of users and restricted to groups of items, Exp. Syst. Appl. 39 (1) (2012) 172–186.
[5] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, Grouplens: an open architecture for collaborative filtering of netnews, in: Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'94), ACM, New York, NY, 1994, pp. 175–186.
[6] U. Shardanand, Social information filtering for music recommendation, M.S. thesis, Massachussets Institute of Technology, 1994.
[7] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: Proceedings of the 10th International Conference on World Wide Web, 2001, pp. 285–295.
[8] G. Linden, B. Smith, J. York, Amazon.com recommendations: item-to-item collaborative filtering, IEEE Internet Comput. 7 (1) (2003) 76–80.
[9] S. Darmoul, H. Pierreval, S.H. Gabouj, Scheduling using artificial immune system metaphors: a review, in: Proceedings of IEEE Conference on Service Systems and Service Management, 2006, pp. 1150–1155.
[10] A.M. Acilar, A. Arslan, A collaborative filtering method based on artificial immune network, Exp. Syst. Appl. 36 (2009) 8324–8332.
[11] S. Cayzer, U. Aickelin, A recommender system based on idiotypic artificial immune networks, J. Math. Modell. Algor. 4 (2) (2005) 181–198.
[12] N.K. Jerne, Towards a network theory of the immune system, Ann. Immunol. (Paris) 125C (1–2) (1974) 373–389.
[13] D. Dasgupta, Z. Ji, F. Gonzalez, Artificial immune systems research in the last five years, in: Proceedings of the Congress on Evolutionary Computation Conference, Canberra, 2003, pp. 8–12.
[14] B. Alatas, E. Akin, Mining fuzzy classification rules using an artificial immune system with boosting, Adv. Databases Inform. Syst. 3631 (2005) 283–293.
[15] U. Shardanand, P. Maes, Social information filtering: algorithms for automating "word of mouth, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'95), 1995, pp. 210–217.
[16] J.L. Herlocker, J.A. Konstan, A. Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, in: Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1999, pp. 230–237.
[17] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence, 1998, pp. 43–52.
[18] H.J. Ahn, A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem, Inform. Sci. 178 (1) (2008) 37–51.
[19] H. Liu, Z. Hu, A. Mian, H. Tian, Z. Zhu, A new user similarity model to improve the accuracy of collaborative filtering, Knowledge-Based Syst. 56 (2014) 156–166.
[20] B. Sarwar, G. Karypis, J. Konstan, J. Ridel, Application of dimensionality reduction in recommender systems-a case study, in: Proceedings of the ACM WebKDD Workshop, 2000.
[21] J. Canny, Collaborative filtering with privacy via factor analysis, in: Proceedings of the 25th Annual International ACMSIGIR Conference on Research and Development in Information Retrieval (SIGIR'02), 2002, pp. 238–245.
[22] D. Billsus, M.J. Pazzani, Learning collaborative information filters, in: Proceedings of the 15th International Conference on Machine Learning, 1998, pp. 46–54.
[23] K. Goldberg, T. Roeder, D. Gupta, C. Perkins, Eigentaste: a constant time collaborative filtering algorithm, Inform. Retrieval 4 (2) (2001) 133–151.
[24] T. Hofmann, Latent semantic models for collaborative filtering, ACM Trans. Inform. Syst. 22 (1) (2004) 89–115.
[25] H. Langseth, T.D. Nielsen, A latent model for collaborative filtering, Int. J. Approx. Reason. 53 (4) (2012) 447–466.
[26] H. Willke, Organisierte Wissensarbeit, Zeitschrift für Soziologie 27 (3) (1998) 161–177.
[27] S. Hermann, Produktive Wissensarbeit – Eine Herausforderung, in: S. Hermann, (Ed.), Ressourcen strategisch nutzen. Wissen als Basis für den Dienstleistungserfolg, vol. 204, Fraunhofer IRB Verlag, Stuttgart. p. 207–228.
[28] Q. Liu, E. Chen, H. Xiong, C.H.Q. Ding, J. Chen, Enhancing collaborative filtering by user interest expansion via personalized ranking, IEEE Trans. Syst., Man, Cybernet., Part B: Cybernet. 42 (1) (2012) 218–233.
[29] R.W. Hamming, Error detecting and error correcting codes, Bell Syst. Tech. J. 29 (2) (1950) 147–160.
[30] K. Pearson, Notes on regression and inheritance in the case of two parents, in: Proceedings of the Royal Society of London, vol. 58, 1895, pp. 240–242.
[31] R.J. Hyndman, A.B. Koehler, Another look at measures of forecast accuracy, Int. J. Forecast. 22 (4) (2006) 679–688.
[32] V. Labatut, H. Cherifi, Accuracy measures for the comparison of classifiers, in: 5th International Conference on Information Technology Amman, Jorda, 2011.

Please cite this article in press as: M.-H. Chen et al., Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.04.005