Alleviating Item-Side Cold-Start Problems in Recommender Systems Using Weak Supervision

SANG-MIN CHOI1,2, KIYOUNG JANG1,6, TAE-DONG LEE3, ABDALLAH KHREISHAH4, (Senior Member, IEEE), AND WONJONG NOH5, (Member, IEEE)

1Department of Computer Science, Yonsei University, Seoul 03722, South Korea
2Research Team, The Level Inc., Seoul 06160, South Korea
3Department of Electrical Engineering, Korea University, Seoul 02841, South Korea
4Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA
5School of Software, Hallym University, Chuncheon 24252, South Korea
6Tmax Soft, Seongnam 13590, South Korea

Corresponding author: Wonjong Noh (wonjong.noh@hallym.ac.kr)

This work was supported by the Hallym University Research Fund, in 2020, under Grant HRF-202008-016.

ABSTRACT Recently, recommender systems have been used in various fields. However, they are still plagued by many issues, including cold-start and sparsity problems. The cold-start problem occurs when users are unable to make recommendations to other users owing to a complete lack of information about certain items. This problem can exist both at the user side and the item side. User-side cold-start problems occur when new users access the systems; item-side cold-start problems occur when new items are added to databases. In this study, we addressed the item-side cold-start problem using the concept of weak supervision. First, a new process for identifying feature based representative reviewers in a rater group was designed. Then, we developed a method to predict the expected preferences for new items by combining content-based filtering and the preferences of representative users. Through extensive experiments, we first confirmed that in comparison to existing methods, the proposed approach provided enhanced accuracy, which was evaluated by determining a mean absolute error for the average ratings. Then, we compared the proposed scheme with the collaborative filtering (CF) and neural CF approaches (NCF). The estimation by the proposed approach was 21% and 38% more accurate than CF and NCF in terms of mean absolute error (MAE), respectively. In future, the proposed framework can be applied in various recommender systems as a core function.

INDEX TERMS Cold-start problem, recommender system, representative user, social group, weak supervision.

I. INTRODUCTION

Since the late twentieth century, the number of Internet users has increased rapidly; many of them interact through online social networks, effectively making the web community similar to actual society. Over the years, there have also been some important changes in the online social network. First, in the past, the general public mainly accepted information from the mass media. However, in recent times, they also receive information from opinion leaders who exert significant influence over others. Therefore, many online articles posted by influential bloggers, because of their significant influence on other users, are used as corporate marketing tools or political advertisements. Second, members of a social network receive, provide, and transmit a large amount of information. The development of the web has allowed many people to access abundant information. These Internet users can upload their data and download desired information using web platforms.

However, to maximize the benefits of online social networks, it is necessary to solve certain issues. First, it is very necessary to identify the influential users in an online society. Recently, there have been advances in research [1]–[6] that identifies the representative reviewers or opinions within social networks. Here, the term “representative reviewers” denotes those users who provide a good indication of ratings from other users. Second, in response to their uploading behaviors, users need to be protected from generally
unwanted data such as spam. Due to these reasons, a web-based recommendation system has been actively developed through various research; this system generally utilizes the similarities between users or items [7], [8] or the associative methods exploiting the similarities of both [9], [10]. These efforts have significantly improved recommendation accuracy. However, many unsolved issues remain in the recommendation system, including cold-start and sparsity problems [11], [12]. The cold-start problem occurs when users are unable to make recommendations owing to a complete lack of information about certain items; the sparsity problem occurs when the database only contains a small amount of information [11]–[14], which affects the recommendation accuracy. The cold-start problem in recommender systems exists both at the user side and the item side [11]–[13], [15]. The user-side cold-start problem occurs when new users access the systems. As new users do not have registered preferences, they are excluded from recommendation processes and do not receive any recommendation results [15]–[17]. The item-side cold-start problem occurs when new items are added to databases. As these items are not registered as preferred by any users, they are excluded from recommendation processes and cannot be promoted to users. In this study, we focused on addressing the item-side cold-start problem.

A. RELATED WORK

1) ITEM-SIDE COLD-START PROBLEMS IN RECOMMENDER SYSTEMS

Various research studies have analyzed the item-side cold-start problem; one of the best-known approaches is the use of item features such as category information [13], [18], [19]. Choi and Han [20] studied the prediction of preferences for new items using the option of representative users extracted from user rating networks. Gunawardana and Meek [21] addressed the item-side cold-start problem based on the relations between the item attributes and proposed the use of the Boltzmann machine for recommendation. Using a movie database as a case study, Gantner and Griffiths [22] attempted to cluster new items using the attribute data. This cluster method based on the attribute data mitigated the item-side cold-start problem. Anand et al. [23] proposed the metric method based on the user action history to alleviate the item-side cold-start problem. Sun et al. [24] clustered the items using the attribute data and preferences and created a decision tree that can be applied to new items and existing items and can predict preferences for new items. Feil et al. [25] examined different gamification patterns and their effect on the number of the report ratings provided by the average user. They provided results for the positive effect of instantiating gamification patterns. Volkovs et al. [26] combined content- and neighbor-based models to address the cold-start problem in recommender systems. The approach produced consistent results in actual testing. Wang et al. [27] deployed information from an ad platform in an online shopping domain. They attempted to build a cross-domain system. Based on the shared users in the system, Wang et al. attempted to resolve cold-start problems.

2) RECOMMENDER SYSTEMS BASED ON DEEP NEURAL NETWORKS

In many studies, supervised learning-based machine learning approaches have been used to address the cold-start problem in recommender systems [28]–[30]. However, this is difficult because recommending an item under cold-start conditions presents similar problems to processing unlabeled data. Owing to the recent popularity of deep learning, it has been applied by various studies for recommender systems [31]–[34]. A typical example is the neural collaborative filtering (NCF), which utilizes a neural network as a collaborative filtering (CF) technique [35]. The approach is based on learning not only the parameters of the model, such as in the traditional deep learning method [31], [32], but also the embedding vector, similar to the Word2vec technique [36]. Based on these embedding vectors, the NCF techniques derive prediction results through two methods: generalized matrix factorization (GMF) and multi-layer perceptron (MLP). The GMF predicts preferences after the user and item vectors have been concatenated. Similar to the GMF, the MLP also concatenates the user and item vectors; however, it additionally forms a fully connected layer. After the data passes through the configured layer, predictions can be derived. Some studies have also focused on a neural matrix factorization (NeuMF) approach [35], which incorporates both the GMF and the MLP methods. In this technique, a prediction is achieved through the concatenation of the last layer of the GMF and MLP. The training of the model begins with the extraction of the user and item vectors. Initially, each vector is initialized with random values, and the vectors are learned through the training data. The resulting user and item vectors are used to predict results. The advantage of this approach is that the MLP can be used to learn the higher-dimensional meaning of latent calculations, rather than simple ones, that predict results as the product of user and item vectors using the matrix factorization (MF) method [37]. Recommender system based on deep learning have been applied to YouTube platforms [38]. The landing pages of YouTube have different configurations for each user account, i.e., a personalized recommender system is used to provide contents. The techniques used on YouTube consist of two modules: the candidate generation module and ranking module. The candidate generation module extracts video candidates using the user profile information and video viewing history information. Next, the ranking module predicts the video candidates, considering recommendation as a classification problem. This technique considers each video as one class; it can handle extremely large class classifications. To improve the learning efficiency, the module addresses important sampling. In addition, the recommendation result is predicted based on the user information and video viewing history information for candidate videos extracted from the candidate generation module in the ranking module. The prediction processes are similar to...
the NCF because these ranking modules predict recommendation results through the MLP by embedding the users and items. In addition, these techniques have exhibited considerable performance improvements. Thus, deep learning-based research is currently very active in the field of recommender systems, and has already found real applications.

B. CONTRIBUTIONS AND ORGANIZATIONS
Currently, most recommender systems are based on the application of CF approaches to predict user preferences. However, the CF methods that are based on user preferences do not make use of weak supervision [39], [40], which has the advantage of reducing time and costs in building training sets for unlabeled data by empirically generating training data using an external knowledge base [41]. Furthermore, weak supervision can be used when the user preferences cannot be employed or when the recommender system needs to be constructed using information other than preference [39], [42], [43]. Therefore, unlike legacy works, this study addresses the item-side cold-start problems using the concept of weak supervision, which can be used in a content-based recommendation system such as news recommendations. This prediction can provide useful and previously inaccessible information on new items and minimize problems related to the exclusion of new items from the recommendation process.

Our contribution can be summarized as follows.
• We proposed a weak supervision-based novel approach that exploits content-based filtering and the activities of representative users to predict the preferences for new items in cold-start situations. First, we employed the content-based filtering approach, which utilizes item features, such as category information, to recommend items. This study focuses on only the genre information as the item feature because its uncertainty (or entropy) is the smallest in the movie data [44], which means that it comparatively guarantees stable results. Second, we applied a method that identifies representative users for one of the item features, that is, we identified the representative users for each genre in this study. We conjectured that the average ratings for new items calculated through the choices of representative users enable more reliable predictions than the average of the ratings of other users.
• Through various tests, we showed that exploiting the ratings of the proposed representative users generates more precise results than other approaches. Furthermore, we found the optimal number of representative users through extensive simulations.
• We proved that the proposed approach outperforms the typical collaborative filtering-based approaches: CF and NCF. The proposed approach was 21% and 38% more accurate than CF and NCF in terms of mean absolute error (MAE).

This paper is organized as follows. The proposed algorithm is presented in Section II. The experiment and results are detailed in Section III. IV is the conclusion.

| Dataset     | Attribute       | Explanation                  |
|-------------|-----------------|------------------------------|
| Movie       | MovieID, Title, Genre | There are 3,883 movies in total |
| User        | UserID, Gender, Age, Occupation, Zip-code | There are 6,040 users in total |
| Rating      | UserID, MovieID, Rating, Timestamp | There are 1,000,209 ratings in total |

| No | Genre     | No | Genre     |
|----|-----------|----|-----------|
| G1 | Action    | G10 | Film-Noir |
| G2 | Adventure | G11 | Horror    |
| G3 | Animation | G12 | Musical   |
| G4 | Children  | G13 | Mystery   |
| G5 | Comedy    | G14 | Romance   |
| G6 | Crime     | G15 | Sci-Fi    |
| G7 | Documentary | G16 | Thriller  |
| G8 | Drama     | G17 | War       |
| G9 | Fantasy   | G18 | Western   |

In this section, we address how to alleviate the item-side cold-start problems for new items within recommendation systems using the concept of representative reviewers within the rater groups. Toward that end, in this study, we employed a movie database provided by GroupLens, as shown in Table 1, which comprises 3,883 items and 6,040 users. The only item feature provided by the GroupLens movie database is the genre information. Each movie in the database belongs to at least one genre, i.e., a Toy Story movie in the database falls under animation, children, and comedy. Table 2 shows all 18 genres of the database.

The proposed approach can be summarized in the three steps in Fig. 1: deriving genre preference, identifying representative users, and generating average preference for an item.

A. DERIVING EACH USER’S GENRE PREFERENCES FROM USER-ITEM RATINGS
We derived the genre preferences by utilizing the user ratings on the GroupLens database. To achieve this, the user’s genre
selection tendency was first calculated; subsequently, based on this, each user’s genre preferences were derived.

1) SELECTION TENDENCY
Fig. 2 shows an example of the drawing selection tendency using movie items. User A has selected eight movies, each of which is representative of a mixture of different genres. The first movie, called Movie 1, is associated with Genres $G_1$, $G_2$, and $G_4$. This means, for example, that $G_1$ is an animated film, $G_2$ is a children’s film, and $G_4$ is comedy. All the movies on the list have their own genre combination. The selection preference of User A was studied by counting the number of times each genre appears on the list. For example, the selection tendency of Genre $G_1$ is three because three movies, Movies 1, 4, and 6, fall under it. The selection tendency of Genre $G_2$, is six, as Movies 1, 2, 4, 5, 6, and 7 fall under it. We repeated this method for all the genres on the list, following which we organized the selection tendencies of User A. Table 3 illustrates the selection tendencies of User A.

| Genre ID | Selection tendency |
|----------|--------------------|
| $G_1$    | 3                  |
| $G_2$    | 6                  |
| $G_3$    | 5                  |
| $G_4$    | 2                  |
| $G_5$    | 3                  |
| $G_6$    | 5                  |

2) GENRE PREFERENCE
After computing the selection tendency, we used it, together with the ratings for each movie, to determine the genre preference. Fig. 3 illustrates the process; the selection tendency of User A, as shown in Table 3, was examined to determine the preference for Genre $G_2$. We first checked User A’s ratings for each movie, and then allocated ratings to each movie to establish the genre preference for $G_2$. In this example, the genre preference of $G_2$ for Movie 1 is 5. In the movie selection list in Fig. 3, Genre $G_2$ appears a total of six times (Movies 1, 2, 4, 5, 6, 7). Following the same procedure, we then calculated the average score for each genre in all the selected movies, which is referred to as a genre preference.

B. IDENTIFYING REPRESENTATIVE USERS FOR GENRE
We considered the users who were able to represent their group as representative users, i.e., the preferences of some representative raters can represent that of all other raters. We used the average ratings to identify the representativeness of the ratings of these users because average ratings are a convenient way to determine the views of the majority. Hence, many e-commerce web applications utilize average ratings to show the quality of items.

In their previous study, Choi and Han [2], [20] extracted the representative reviewers from the rater groups for movies or music using (1):

$$ U_S = \sum_{i \in A} |R_s(i) - R_{\mu}(i)|, $$

where $A$ is a set of items rated by a user, $S$, $|A|$ is the cardinality of $A$, $R_s(i)$ is the rating of item $i$ by the user, $S$, and $R_{\mu}(i)$ is the average rating of item $i$ in the database. The result of (1) shows the average difference between the item ratings by one user and the average ratings of the items. If a user’s outcome for (1) is 0, its rating is the same as the average rating calculated from other users’ ratings. From this result, we concluded that users with a result close to 0 can represent other users’ ratings for the same items. Therefore, only one user group can represent the rater group.

However, unlike (1), in this study, we identified representative users for each genre using (2). We obtained a total of 18 representative groups, as the GroupLens database provides 18 genres. The representative users for the genres were identified as follows.

We first organized the genre preferences for each user into a user-genre matrix. In Fig. 4, the bottom table shows the user-genre matrix for $n$ users and 18 genres. Each row of the matrix indicates the genre preferences of each user. After composing the user-genre matrix, we applied (2) to each genre:

$$ GU_S = |P_S(i) - P_{\mu}(i)|, $$
where $P_S(i)$ is the preference of genre $G_i$ by User $S$ and $P_\mu(i)$ is the average preference for the genre $G_i$. We obtained a total of 18 result graphs after applying (2) to all the users in the database, as shown in Fig. 5. In the figure, the y-axis indicates the results of (2), and the x-axis represents 6,040 users, in ascending order of their results. For each genre graph in Fig. 5, we selected users with low scores for (2) as representative users of each genre.

The selection process for representative users using a sample genre $G_1$ is summarized in Fig. 6. The user-genre matrix was the same as in the bottom table of Fig. 4. We first calculated the average genre preference for Genre $G_1$. Second, we calculated the difference between the individual user’s genre preference and the average. This calculation enabled us to obtain the results of (2) for Genre $G_1$ and organize the scores in ascending order, as illustrated in Fig. 6. Finally, we selected users with low scores as representative users for Genre $G_1$.

### C. Generating Average Preference for a New Item

Each new movie falls under at least one category. Fig. 7 illustrates the entire process involved in predicting the average preference for a new item. This encompasses a total of three steps: The first step is to extract representative users for each genre that appears related to a new item. In Fig. 7, the genre combination of the new item is $G_1$, $G_2$, and $G_8$. We extracted low scorers, i.e., those users with a low score based on (2), for Genres $G_1$, $G_2$, and $G_8$. One important finding in this context was that each genre had different representative users. For example, in the user-genre matrix of Fig. 4, User $A$ had a different level of preference for Genres $G_1$ and $G_2$, and the overall average preference for the two genres was also different. This indicates that User $A$ can be a representative user for Genre $G_1$, but not for Genre $G_2$. Thus, we had different representative users for each genre.

The second step was to extract the representative users’ genre preferences. We then calculated their average preference for each genre with a new item. In the example, we extracted the genre preferences of representative users for $G_1$, $G_2$, and $G_8$. The preference for each genre varies because each genre has different representative users.

The final step was to score the average preference for a new item. In the example, after calculating the average preference for each genre, with regard to the genre preferences extracted from each representative user, we determined the average preference for a new item by dividing the sum of each average preference for each genre. This encompasses a total of three steps: The first step is to extract representative users for each genre that appears related to a new item. In Fig. 7, the genre combination of the new item is $G_1$, $G_2$, and $G_8$. We extracted low scorers, i.e., those users with a low score based on (2), for Genres $G_1$, $G_2$, and $G_8$. One important finding in this context was that each genre had different representative users. For example, in the user-genre matrix of Fig. 4, User $A$ had a different level of preference for Genres $G_1$ and $G_2$, and the overall average preference for the two genres was also different. This indicates that User $A$ can be a representative user for Genre $G_1$, but not for Genre $G_2$. Thus, we had different representative users for each genre.

The second step was to extract the representative users’ genre preferences. We then calculated their average preference for each genre with a new item. In the example, we extracted the genre preferences of representative users for $G_1$, $G_2$, and $G_8$. The preference for each genre varies because each genre has different representative users.

The final step was to score the average preference for a new item. In the example, after calculating the average preference for each genre, with regard to the genre preferences extracted from each representative user, we determined the average preference for a new item by dividing the sum of each average preference for each genre.
into the number of genres that have new items, as shown in (3),

\[ R_n = \frac{1}{|GC_n|} \sum_{i \in GC_n} \frac{1}{|RU_i|} \sum_{S \in RU_i} P_S(i), \]  

(3)

where \( GC_n \) is the genre combination of a new item \( n \), \( RU_i \) is the set of representative users of the \( i \)th genre, \( S \) is one representative user of the \( i \)th genre, and \( P_S(i) \) is the genre preference extracted from the representative user \( S \) for the \( i \)th genre.

III. EXPERIMENTS AND ANALYSIS

A. EXPERIMENT DESIGN

To test the proposed approach, we randomly selected 100 items from a total of 3,883 items and considered them as a set of new items.

First, we evaluated the accuracy of the proposed approach. To this end, we compared the results obtained through our approach to an average rating. Our comparisons utilize the mean absolute error (MAE [45], as shown in (4):

\[ MAE = \frac{1}{m} \sum_{n \in PI} |R_n - \mu_n|, \]  

(4)

where \( m \) is the number of test items, \( PI \) is the set of test items, and \( R_n \) and \( \mu_n \) are the average ratings, as calculated by the proposed approach, and the real average rating for item \( n \), respectively. Fig. 8 illustrates how test results are obtained for a test item. Where we assume that an item \( I_1 \), selected as the test item, has no user responses, i.e., no user-generated ratings. The example consists of three steps: (i) calculating the average rating based on those provided by the user-item matrix for Item \( I_1 \); (ii) generating the average preference for the test item; and (iii) calculating the MAE between the average rating drawn by the user-item matrix and the average preferences generated by the proposed approach.

Then, we evaluated the proposed approach using different groups of representative users. These tests led to a more precise analysis of the number of representative users for predicting the average preferences for new items. In this analysis, each different group configuration has a different number of users, which is given by (5):

\[ U_{gn} = \frac{|U|}{|U_g|}, \]  

(5)

where \( U \) is the set of all the users in the database, \( U_g \) is the set of the groups, and \( U_{gn} \) is the number of users in the \( n \)th group. For example, if \(|U_g|\) is 10, we created ten groups by dividing all the users. Thus, each group had approximately 600 users, as the database comprised 6,040 users. Subsequently, we generated ten average preferences for new items, one for each group. Fig. 9 provides an example consisting of ten groups and the ten average preferences generated for the tests. Based on the positions of each user determined by (2), we divided each group. Thus, in Fig. 9, we utilized the 18 graphs derived from the previous section. There are ten groups of users for each genre. We then generated the average preference for new items based on these groups of users. We generated the average preferences for new items using all the sorted users. In a previous study [46], the top 10% and bottom 10% users in the graph results (representing the results of the representative scores sorted in ascending order) show distinct differences with regard to the representation of their group. In this test, we created groups the number of 600, 500, 400, 300, 200, 100, 50, 40, 30, 20 groups, i.e., comprising 10 to 300 users per group. In each group, we used 10, 12, 15, 20, 30, 60, 120, 150, 200, and 300 users as the representative number of users.

Finally, we compared our method with the CF and NCF approaches, to verify the accuracy of our results. For the CF approach, we utilized the user-based CF strategy [7], [10], which recommends items in two steps. The first step involves selecting similar users using popular similarity measures such as the Pearson correlation coefficient or cosine similarity [8]. We used 0.5 as the threshold for the Pearson correlation and
cosine similarity. After selecting the similar users, the CF approach predicted the user preferences for some items. For the NCF approach, we applied the stochastic gradient descent algorithm for optimization and set 0.01 as the learning rate. We also set five layers of the MLP module; each layer had 64 dimension factors. We used 100 epochs in each test and selected the lowest MAE as the results.

To learn the implicit data in NCFs [35], such as Bayesian personalized ranking [47], the label for the unseen data was considered 0. However, when the NCF derives its prediction results, including the unseen data whose users’ preferences are considered as 0, can degrade the accuracy of the prediction results. Therefore, we excluded negative sampling that can produce distorted predictions from the experiments.

B. RESULT ANALYSIS

1) VALIDATION ON USE OF REPRESENTATIVE USER

Fig. 10 shows the MAE graphs for each group. In Fig. 10, the y-axis represents the MAE, and the x-axis represents the nth groups. In each group, the MAE gradually increases with the increase in the x-axis. This indicates that a large difference was observed between the average preference generated by high scorers and the real average rating. These test results confirm our assumption that the low scorers identified by eq. (2) can be used to predict the average preference for new items more precisely. Fig. 11 shows the detailed MAE values derived for each group of users. Fig. 12 shows the MAE values for the top five and bottom five groups in each group configuration. As shown in Fig. 12, the following were experimentally verified: (i) The lowest MAE was found in the top 1 of 50 groups configuration. (ii) The MAE values in 50 groups configuration were lower compared to the other groups. This indicates that the most precise average ratings were obtained by selecting 120 users as the set of representative users. (iii) The bottom group was associated with a higher MAE result compared to the top group. The bottom groups, as opposed to the top, contain users whose evaluation scores for each item are not close to the average. This means that the predictions obtained through the users close to the average ratings were more accurate than the results obtained by the bottom groups. Thus, we can conclude that focusing on the representative users, as opposed to other users, leads to a more precise prediction of the average rating for a new item.

2) COMPARISONS OF PROPOSED ALGORITHM WITH THE CF AND NCF APPROACHES

Fig. 13 shows the comparison of the CF, NCF, and proposed approach. In our approach, we predicted group preferences by selecting the representative users. The CF approaches, on the
In this study, we addressed the cold-start problem for new items in the recommendation system by proposing a new approach that can predict average ratings using weak supervision. First, we proposed how to identify representative reviewers in the raters group. Average ratings can be predicted more precisely using the ratings of the proposed representative users compared to the average ratings of other users. Second, we proposed content-based filtering to generate the average ratings of new items. This helped to determine the selection tendency and category preference of users. Finally, through extensive experiments, we proved that the proposed content-filtering-based representative users generated precise results for the cold item recommendation. We compared the proposed approach to the CF and NCF approaches. The results showed that the proposed approach was 21% and 38% more precise than the typical CF and NCF approaches, respectively. As a future work, we will apply this approach to more diverse databases to analyze the results. We will also utilize MF to extract representative users with the user-genre matrix through a more diversified analysis.

IV. CONCLUSION

other hand, predict personalized preferences, rather than group preferences. Therefore, the proposed and compared approaches were evaluated with the number of groups and users as a criterion, as shown in Fig. 13. We ensured the comparison was fair. For instance, if the proposed approach was evaluated on 600 groups, which indicates that 10 users were used as the size of the representative user group, we chose 10 users in the CF approach, i.e., 600 groups in the proposed approach match 10 users in the CF approach.

When the proposed scheme was compared to the CF approach with the Pearson correlation coefficient, they had similar MAE values. However, in the 40 and 50 groups (i.e., 150 and 120 users in each group, respectively), the proposed approach provided more precise results. When the proposed scheme was compared to the CF approach with the cosine correlation coefficient, the proposed approach yielded even more accurate MAE values for all the group configurations. Moreover, the MAE in the NCF were between 0.7 and 0.8, which means that the proposed approach also yielded more accurate MAE values than the NCF approach for all the group configurations. In general, the NCF learns using many parameters compared to the proposed approach or the matrix factorization method. Thus, learning existing information is clearly better. However, predicting cold-start items has high deviation. That is, the recommendation accuracy of the NCF could be better in normal situations, compared to under cold-start conditions, i.e., the prediction by the NCF was more accurate in pairwise learning, compared to in regression-based learning.

These results above indicate that the proposed approach was more effective than the conventional CF and NCF in determining the average ratings for new items. The estimation by the proposed approach was 21% and 38% more accurate than CF and NCF in terms of mean absolute error (MAE). In addition, the predictions of new items using the proposed approach can be utilized, as a result of weak supervision, in deep learning-based approaches such as the NCF. By utilizing the proposed approach, in the deep learning-based approach, the deviation of the results can be reduced with regards to various learning parameters.

FIGURE 13. Comparison of MAE values of CF, NCF and proposed approach.

| Criterion | MAE of Top-5 groups (our approach) | Criterion | MAE by CF approach | MAE by NCF approach |
|-----------|-----------------------------------|-----------|---------------------|---------------------|
| 600 groups | 0.58 0.60 0.60 0.60 0.60 | 10 groups | 0.59 1.00 | 0.71 |
| 500 groups | 0.62 0.64 0.64 0.64 0.64 | 12 groups | 0.59 0.98 | 0.72 |
| 400 groups | 0.68 0.69 0.69 0.69 0.69 | 15 groups | 0.63 1.00 | 0.70 |
| 300 groups | 0.72 0.74 0.74 0.74 0.74 | 20 groups | 0.62 0.97 | 0.74 |
| 200 groups | 0.76 0.78 0.78 0.78 0.78 | 30 groups | 0.62 0.94 | 0.74 |
| 100 groups | 0.81 0.83 0.83 0.83 0.83 | 60 groups | 0.62 1.02 | 0.71 |
| 50 groups | 0.92 0.94 0.94 0.94 0.94 | 120 groups | 0.63 0.93 | 0.72 |
| 40 groups | 0.95 0.97 0.97 0.97 0.97 | 150 groups | 0.63 1.00 | 0.73 |
| 30 groups | 0.99 1.01 1.01 1.01 1.01 | 200 groups | 0.64 0.95 | 0.71 |
| 20 groups | 1.02 1.04 1.04 1.04 1.04 | 300 groups | 0.63 0.97 | 0.72 |

REFERENCES

[1] N. Agarwal, H. Liu, L. Tang, and P. S. Yu, “Identifying the influential bloggers in a community,” in Proc. 1st Int. Conf. Web Search Web Data Mining (WSDM), 2008, pp. 207–218.
[2] S.-M. Choi and Y.-S. Han, “Representative reviewers for Internet social media,” Expert Syst. Appl., vol. 40, no. 4, pp. 1274–1282, Mar. 2013.
[3] C. Durugbo, “Modelling user participation in organisations as networks,” Expert Syst. Appl., vol. 39, no. 10, pp. 9230–9245, Aug. 2012.
[4] Y.-S. Han, L. Kim, and J.-W. Cha, “Computing user reputation in a social network of Web 2.0,” Comput. Inform., vol. 31, no. 2, pp. 447–462, 2012.
[5] J. J. Jung, “Computational reputation model based on selecting consensus choices: An empirical study on semantic wiki platform,” Expert Syst. Appl., vol. 39, no. 10, pp. 9002–9007, Aug. 2012.
[6] H. Kwak, C. Lee, H. Park, and S. Moon, “What is Twitter, a social network or a news media? in Proc. 19th Int. Conf. World Wide Web (WWW), 2010, pp. 591–600.
[7] J. L. Herlocker, J. Konstan, A. Borchers, and J. Riedl, “An algorithm framework for performing collaborative filtering,” in Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR), 1999, pp. 230–237.
[8] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, “Item-based collaborative filtering recommendation algorithms,” in Proc. 10th Int. Conf. World Wide Web (WWW), 2001, pp. 285–295.
[9] K. Honda, A. Notsu, and H. Ichihashi, “Collaborative filtering by sequential extraction of user-item clusters based on structural balancing approach,” in Proc. IEEE Int. Conf. Fuzzy Syst. (Fuzz-IEEE), Aug. 2009, pp. 1540–1545.
[10] J. Wang, A. P. de Vries, and M. J. T. Reinders, “Unifying user-based and item-based collaborative filtering approaches by similarity fusion,” in Proc. 29th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR), 2006, pp. 501–508.
[11] S.-M. Choi and Y.-S. Han, “A content-confusion recommendation system based on category correlations,” in Proc. 5th Int. Multi-Conf. Comput. Global Inf. Technol. (ICCGI), Sep. 2010, pp. 1257–1260.
[12] M. Ishikawa, P. Géczy, N. Izumi, T. Morita, and T. Yamaguchi, “Information diffusion approach to cold-start problem,” in Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.-Workshops (WI-IAT), Nov. 2007, pp. 129–132.
[13] S.-M. Choi, S.-K. Ko, and Y.-S. Han, “A movie recommendation algorithm based on genre correlations,” Expert Syst. Appl., vol. 39, no. 9, pp. 8079–8085, Jul. 2012.
[14] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, “Methods and metrics for cold-start recommendations,” in Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR), 2002, pp. 253–260.
M. Volkovs, G. W. Yu, and T. Poutanen, “Content-based neighbor recommendation,” in Proc. 23rd Int. Conf. World Wide Web (WWW), 2014, pp. 327–336.

P. Pirasteh, J. J. Jung, and D. Hwang, “Item-based collaborative filtering with attribute correlation: A case study on movie recommendation,” in Proc. 6th Asian Conf. Infell. Database Syst. (ACIDBS), 2014, pp. 245–252.

J. Zhang, Q. Peng, S. Sun, and C. Liu, “Collaborative filtering recommendation algorithm based on user preference derived from item domain features,” Phys. A, Stat. Mech. Appl., vol. 396, pp. 66–76, Feb. 2014.

S.-M. Choi and Y.-S. Han, “Identifying representative ratings for a new item in recommendation system,” in Proc. 7th Int. Conf. Ubiquitous Inf. Manage. Commun. (ICUIMC), 2013, p. 64.

A. Gunawardana and C. Meek, “Tied Boltzmann machines for cold-start recommendations,” in Proc. ACM Conf. Rec. Syst. (RecSys), 2008, pp. 19–26.

Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, and L. Schmidt-Thieme, “Learning attribute-to-feature mappings for cold-start recommendations,” in Proc. IEEE Int. Conf. Data Mining (ICDM), Dec. 2010, pp. 176–185.

S. S. Anand and N. Griffiths, “A market-based approach to address the new item problem,” in Proc. 5th ACM Conf. Rec. Syst. (RecSys), 2011, pp. 205–212.

D. Sun, Z. Luo, and F. Zhang, “A novel approach for collaborative filtering to alleviate the new item cold-start problem,” in Proc. 13th Int. Symp. Commun. Inform. Technol. (ISCIT), Oct. 2011, pp. 402–406.

S. Feil, M. E. Kretzer, K. Werder, and A. Maedche, “Using gamification to tackle the cold-start problem in recommender systems,” in Proc. 19th ACM Conf. Comput. Supported Cooperat. Work Social Comput. Companion (CSCW Companion), San Francisco, CA, USA, Feb./Mar. 2016, pp. 253–256, doi: 10.1145/2810582.2860979.

M. Volkovs, G. W. Yu, and T. Poutanen, “Content-based neighbor models for cold start in recommender systems,” in Proc. Rec. Syst. Challenge (RecSys Challenge), 2017, pp. 7:1–7:6. [Online]. Available: http://doi.acm.org/10.1145/3124791.3124792

H. Wang, D. Amagata, T. Maekawa, T. Hara, H. Niu, K. Yonekawa, and M. Kurokawa, “Preliminary investigation of alleviating user cold-start problem in e-commerce with deep cross-domain recommender system,” in Proc. World Wide Web Conf. (WWW), San Francisco, CA, USA, May 2019, pp. 398–403.

A. L. V. Pereira and E. R. Hruschka, “Simultaneous co-clustering and learning to address the cold start problem in recommender systems,” Knowl.-Based Syst., vol. 82, pp. 11–19, Jul. 2015, doi: 10.1016/j.knosys.2015.02.016.

S. Kang, J. Hwang, D. Lee, and H. Yu, “Semi-supervised learning for cross-domain recommendation to cold-start users,” in Proc. 28th ACM Int. Conf. Infell. Knowl. Manage. (CIKM), Beijing, China, Nov. 2019, pp. 1563–1572, doi: 10.1145/3357384.3357914.

F. Peng, X. Lu, C. Ma, Y. Qian, J. Lu, and J. Yang, “Multi-level preference regression for cold-start recommendations,” Int. J. Mach. Learn. Cybern., vol. 9, no. 7, pp. 1117–1130, Jul. 2018, doi: 10.1007/s13042-017-0635-2.

H. Wang, N. Wang, and D. Yeang, “Collaborative deep learning for recommender systems,” in Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Sydney, NSW, Australia, Aug. 2015, pp. 1235–1244.

Z. Batrnaz, A. Yurekli, A. Bilge, and C. Kaleli, “A review on deep learning for recommender systems: Challenges and remedies,” Artif. Intell. Rev., vol. 52, no. 1, pp. 1–37, Jun. 2019.

W. Guo, H. Gao, J. Shi, B. Long, L. Zhang, B.-C. Chen, and D. Agarwal, “Deep neural language processing for search and recommender systems,” in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), Anchorage, AK, USA, Jul. 2019, pp. 3199–3208.

J. Xie, F. Zhu, M. Huang, N. Xiong, S. Huang, and W. Xiong, “Unsupervised learning of paragraph embeddings for context-aware recommendation,” IEEE Access, vol. 7, pp. 43100–43109, 2019.

X. He, L. Liao, H. Zhang, L. Nie, and X. Hu, “Neural collaborative filtering,” in Proc. 26th Int. Conf. World Wide Web (WWW), 2017, pp. 173–182.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in Proc. 27th Annul. Conf. Neural Inf. Process. Syst., Lake Tahoe, NV, USA, Dec. 2013, pp. 3111–3119.

Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.

P. Covington, J. Adams, and E. Sargin, “Deep neural networks for YouTube recommendations,” in Proc. 10th ACM Conf. Rec. Syst. (RecSys), Sep. 2016, pp. 191–198.

S. Okura, Y. Tagani, S. Ono, and A. Tajima, “Embedding-based news recommendation for millions of users,” in Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), Aug. 2017, pp. 1933–1942.

M. Tkalcic, A. Ozic, A. Kosir, and J. Tasic, “Affective labeling in a content-based recommender system for images,” IEEE Trans. Multimedia, vol. 15, no. 2, pp. 391–400, Feb. 2013, doi: 10.1109/TMM.2012.2229970.

A. Ratner, S. H. Bach, H. R. Ehrenberg, J. A. Fries, S. Wu, and C. Ré, “Snorkel: Rapid training data creation with weak supervision,” Proc. VLDB Endowment, vol. 11, no. 3, pp. 269–282, 2017. [Online]. Available: http://www.vldb.org/pvldb/vol11/p269-ratner.pdf

H. Zhang, N. Zhao, X. Shang, H. Luan, and T. Chua, “Discrete image hashing using large weakly annotated photo collections,” in Proc. 30th AAAI Conf. Artif. Intell., Phoenix, AZ, USA, Feb. 2016, pp. 3669–3675.

E. Raisi and B. Huang, “Cyberbullying detection with weakly supervised machine learning,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Sydney, NSW, Australia, Jul./Aug. 2017, pp. 409–416, doi: 10.1145/3110025.3110049.

S. Kim, S.-M. Choi, K. L. Man, and Y.-S. Han, “Analyzing item features for cold-start problems in recommendation systems,” in Proc. 8th Int. Conf. Infell. Hiding Multimedia Signal Process. (IHM-MSP), Aug. 2014, pp. 167–170.

M. Bulner, Principle of Statistics. New York, NY, USA: Dover, 1979.

S.-M. Choi, J.-W. Cha, and Y.-S. Han, “Identifying representative reviewers in Internet social media,” in Proc. 2nd Int. Conf. Comput. Collective Intell., Technol. Appl. (ICCCI), 2010, pp. 22–30.

S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “BPR: Bayesian personalized ranking from implicit feedback,” 2012, arXiv:1205.2618. [Online]. Available: http://arxiv.org/abs/1205.2618
TAE-DONG LEE received the B.S., M.S., and Ph.D. degrees in electronics engineering from Korea University, Seoul, South Korea, in 1996, 2001, and 2006, respectively. Since 2006, he had been Researching with the Visual Display Division, Samsung Electronics, Suwon, South Korea, for a period of ten years. He joined the Information and Communication Technology Institute, Korea University, as a Research Professor, for a period of one year. He is currently a Co-Founder with SW Company (MoiraLab) and a CTO in researching. His research interests include cloud and distributed computing, peer-to-peer computing systems, visual processing, and mobile operating systems.

ABDALLAH KHREISHAH (Senior Member, IEEE) received the B.S. degree (Hons.) from the Jordan University of Science and Technology, in 2004, and the M.S. and Ph.D. degrees in electrical and computer engineering from Purdue University, in 2006 and 2010, respectively. He was with NEESCOM. In Fall 2012, he joined the Electrical and Computer Engineering Department, New Jersey Institute of Technology, as an Assistant Professor and an Associate Professor in 2017. His research projects are funded by the National Science Foundation of USA, the New Jersey Department of Transportation, and the UAE Research Foundation. His research interests include network security, machine learning, wireless networks, visible-light communication, vehicular networks, and cloud and edge computing. He was on the TPC committee of several international conferences, such as the IEEE InfoCom. He served as the TPC Chair for WASA in 2017, BDSN in 2015, the IOTSMS in 2015, the IEEE SNAMS in 2014, and the IEEE SDS in 2014. He serves as the Chair for the IEEE EMBS North Jersey chapter. He serves as an Associate Editor for several International journals, including the IEEE/ACM TRANSACTIONS ON NETWORKING.

WONJONG NOH (Member, IEEE) received the B.S., M.S., and Ph.D. degrees from the Department of Electronics Engineering, Korea University, Seoul, South Korea, in 1998, 2000, and 2005, respectively. From 2005 to 2007, he conducted the Postdoctoral Research with Purdue University, West Lafayette, IN, USA, and the University of California at Irvine, Irvine, CA, USA. From 2008 to 2014, he was a Principal Research Engineer with the Samsung Advanced Institute of Technology, Samsung Electronics, South Korea. From 2014 to 2018, he was an Assistant Professor with the Department of Electronics and Communication Engineering, Gyeonggi University of Science and Technology, South Korea. He is currently an Associate Professor with the School of Software, Hallym University, South Korea. His current research interests include fundamental analysis and evaluations on machine learning-based 5G and 6G wireless communications and networks. He received the Government Postdoctoral Fellowship from the Ministry of Information and Communication, South Korea, in 2005. He was also a recipient of the Samsung Best Paper Gold Award in 2010, the Samsung Patent Bronze Award in 2011, and the Samsung Technology Award in 2013.