# Summary

## Collaborative Filtering Recommendation System Correcting User Bias Based on Comments-Rating Relationship

When NN(Neural Network) is used in sentimental analysis, the positive/negative rating that the comment holds can be corresponded into a value of 1 to 5. The returned value can stand how similar the target value and the predicted value are. When the learning process is well done by big data, the target value and the predicted value will be same for the majority of the data. However, minority will have a difference, and it can be interpreted as a bias of the particular user. In this study, similar users are defined as users whose comments are similar. Users who have similar comments have similar ratings obtained by NN. Therefore, using the obatained, or unbiased rating at the recommendation system is a better way compared to using the rating that the users have given.

Figure 1. is a diagram of the recommendation system suggested by this study.

CNN is trained by users' comments, and the unbiased rating trains the recommendation system. Detailed procedure is shown in Figure 2.

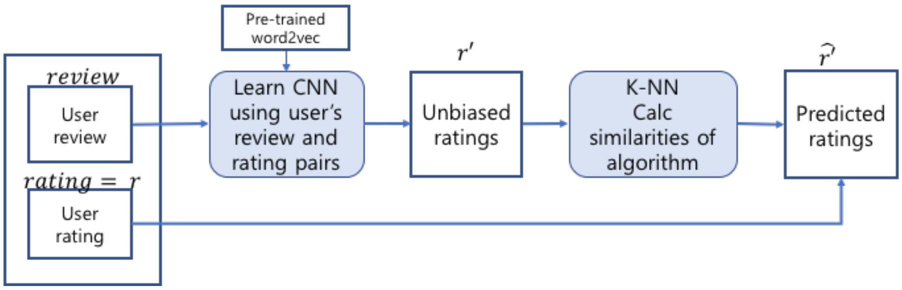


Figure 1. Diagram of the recommendation system

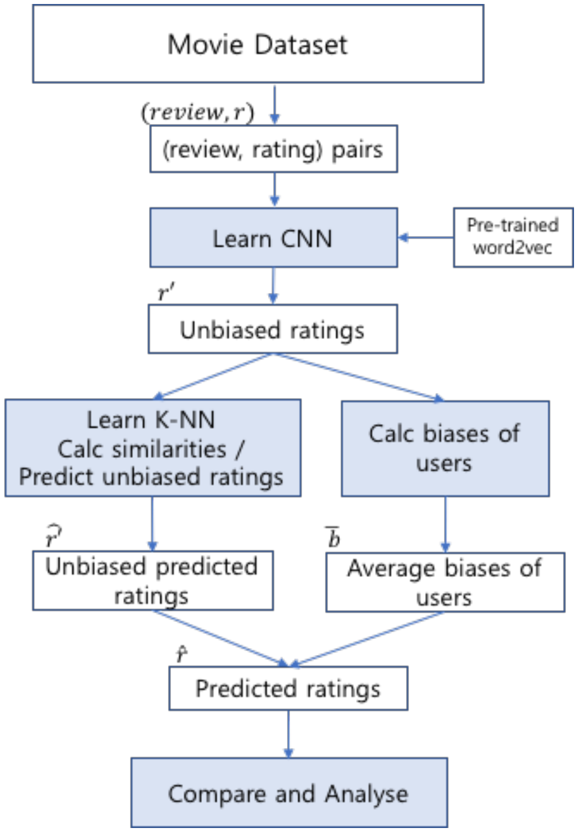


Figure 2. Procedure of the recommendation system

In this study, data collected by SNAP Project, Stanford University was used. The data contains information of comments and points about the movies from Amazon.

At CNN train step, CNN was trained by entire comment $review_A$ and rating $r_A$ that user A has given. 3, 4, 5 were used as the filter size. Dropout value was 0.5, L2 and regularization lambda was 0.1. Batch size was 128, and 150 epoch was trained, resulting the accuracy up to 0.9247.

Then by the trained CNN, new rating, or unbiased rating $r'_A$ was obtained. If A has a bias, let bias for an each comment is $b_A$. Then, $r'_A$ can be thought as $b_A$ added to $r_A$ and can be written as (1).

$$r'_A \simeq r_A - b_A \tag{1}$$

Then, average bias of an user can be written as (2).

$$\overline{b_A} = \frac{\sum_{i}^{n_A}(r'_{A,i} - r_{A,i})}{n_A} \tag{2}$$

The suggested recommendation system was trained by the unbiased rating of all users. The system calculated the neighbors by the Pearson-correlation written as (3).

$$w'(A,B) = \frac{\sum_{i=1}^{q}(r'_{A,i} - \overline{r'_A})(r'_{B,i} - \overline{r'_B})}{\sqrt{\sum_{i=1}^{q}(r'_{A,i} - \overline{r'_A})^2 \sum_{i=1}^{q}(r'_{B,i} - \overline{r'_B})^2}} \tag{3}$$

Predicted unbiased rating can be obtained by the Pearson-correlation above. Predicted unbiased rating of movie i of user A can be written as (4).

$$\widehat{r'_{A,i}} = \overline{r'_A} + \frac{\sum_{j=1}^{k} w'(A,j)(r'_{j,i} - \overline{r'_j})}{\sum_{j=1}^{k} |w'(A,j)|} \tag{4}$$

Predicted biased rating was calculated as the predicted unbiased rating added to average bias of the user. It can be written as (5).

$$\widehat{r_{A,i}} \simeq \widehat{r'_{A,i}} + \overline{b_A} \tag{5}$$

Several different recommendation algorithms in surprise, a 3rd party library in Python, were used for the comparison.

The result when k-NN algorithm was used is shown in Table 1. When Pearson-correlation was used, RMSE was 0.5080 and MAE was 0.3180 at the original algorithm, but RMSE was 0.4981 and MAE was 0.3065 at the suggested algorithm. Similarly, when cosine was used, RMSE and MAE both had decreased. This means that when CNN is used at finding neighbors, more similar neighbors were found.

Table 1. RMSE and MAE values when k−NN algorithm was used

| Recommendation system | | RMSE | | MAE | |
|---|---|---|---|---|---|
| Algorithm | Similarity function | original | suggested | original | suggested |
| k−NN | Pearson | 0.5080 | **0.4981** | 0.3180 | **0.3065** |
| | Cosine | 0.5215 | **0.4997** | 0.3257 | **0.3071** |

RMSE and MAE were decreased for all other algorithms based on k-NN, such as

KNN with Means, KNN with ZScore, KNN baseline, shown in Table 2.

Table 2. RMSE and MAE values of other k−NN algorithms

| Recommendations system | | RMSE | | MAE | |
|---|---|---|---|---|---|
| Algorithm | Similarity function | origianl | suggested | original | suggested |
| k−NN Baseline | Pearson | 1.0182 | **0.9837** | 0.7321 | **0.6894** |
| | Cosine | 1.0121 | **0.9735** | 0.6866 | **0.6460** |
| k−NN with Means | Pearson | 0.3697 | **0.3640** | 0.1057 | **0.1042** |
| | Cosine | 0.3905 | **0.3653** | 0.1303 | **0.1196** |
| k−NN ZScore | Pearson | 0.3474 | **0.3436** | 0.0962 | **0.0958** |
| | Cosine | 0.3930 | **0.3651** | 0.1325 | **0.1217** |

Finally for other recommendation algorithms, suggested algorithm was more efficient for Normal Predictor, Baseline, and Slope One. However for Co-Clustering, SVD, and NMF, the efficiency was similar or lower. It is shown in Table 3.

Table 3. RMSE and MAE values when other recommendation algorithms were used

| Recommendation algorithm | RMSE | | MAE | |
|---|---|---|---|---|
| | original | suggested | original | suggested |
| Normal Predictor | 1.3097 | **1.2453** | 0.9145 | **0.8575** |
| Baseline | 0.5194 | **0.4978** | 0.3279 | **0.3093** |
| Slope One | 0.4032 | **0.3825** | 0.1282 | **0.1188** |
| Co−Clustering | 0.4195 | 0.4849 | 0.1778 | 0.2479 |
| NMF | 0.3346 | 0.3626 | 0.2406 | 0.2537 |
| SVD | 0.3382 | **0.3364** | 0.1658 | **0.1619** |
| SVD++ | 0.2940 | 0.2968 | 0.1211 | **0.1180** |

In this study, a recommendation system was suggested in order to improve the collaborative filtering. Original collaborative filtering, which uses only the rating, doesn’t consider the users’ bias. Therefore, comments were used at the system because comments contain the emotion and purpose of the user. Comments and ratings were used to train CNN and the obatined rating, or the unbiased rating was collaborated the original recommendation algorithms. Through a comparison between the original algorithms and the suggested algorithm, the suggested algorithm performed better.