A Hybrid Recommendation Algorithm with LDA and SVD++ Considering the News Timeliness

Junsong Luo1, Can Jiang2, Peng Tian2 and Wei Huang2,3

1College of Information Science & Technology, Chengdu University of Technology, Chengdu 610059, Sichuan, China
2School of information and software engineering, University of Electronic Science and Technology of China, Chengdu 610054, Sichuan, China
3Department of Computer Science, Guiyang University, Guiyang 550005, GuiZhou, China

Abstract
Collaborative filtering-based recommender systems, which had been applied successfully in E-commerce domain, could filter various types of information and recommend information beyond people’s ever interest. However, recommender systems have a common drawback is low accuracy. In order to further improve the accuracy of the recommendation algorithm, and verify practicability of the model. We proposed a new hybrid recommendation algorithm. Study of the Latent Dirichlet Allocation (LDA) and Singular Value Decomposition algorithm (SVD), and then researching the parameters of news’ implicit feature, time, and recommendation accuracy. We obtained data through crawl. First, we used LDA to classify the news text and used SVD++ algorithm to fuse implicit feedback in SVD model. To incorporate the importance of news timeless, we studied recommendation strength by constantly adjusting the time window of what was affected by news implicit features. Next, we processed the news recommendation sequences of SVD++ with the method of weighting similarity. In the process, we found that although the performance is not good when we only use LDA algorithm, when we combined LDA and SVD++, our model shows a better consequence. In the end, the recommendation accuracy has been improved.

Key words: Implicit Feedback, Latent DirichletAllocation, Singular Value Decomposition Plus Plus, Weighting Similarity, Time Window

1. INTRODUCTION

Network news is the originator of the news media. The depth and speed of its spread were far ahead of traditional media. With the vast network news content, how to push network news effectively remains an important issue for network news. In recent years, recommendation systems, such as Collaborative Filter(CF) (Kondo, Nakatani and Hiromi, 2015) have received growing research interest. The research in the field originated with Amazon (Lika, Kolomvatsos and Hadjieftymiades, 2014), Netflix (Tkalicic, Odic, Kosir and Tasic, 2013), MovieLens, and others. These systems collect users’ feedback and implicit feedback, including users’ ratings, browsing records, purchasing records, evaluating information, and then, they assess users’ preferences to recommend additional products for them.

As a special commodity, network news are related to these other systems but also have their own characteristics. Related websites include Topix.net, Redit, Digg, and others. In the study of news recommendation methods, the commonly used method is Group Lens (Chienchin and Yunchun, 2016), an early collaborative filtering system. Content-based, CF was a successful technique for building a recommendation system (You and Muhong, 2013). However, it also has shortcomings in recommendation news, such as, it is limited by the content of news. So, many other techniques have been developed, including combining collaborative filtering with semantic analysis and weighted TF-IDF technology, or using a clustering algorithm and Latent Dirichlet Allocation for the recommendation of news.

In this paper, we propose a novel news recommendation method. We combine the sub topic classification method based on the LDA model (Qiu and Xu, 2013) and the Singular Value Decomposition algorithm based on matrix decomposition used in Prize Netflix video recommendation with excellent results, and then we apply the Singular Value Decomposition plus (SVD++) algorithm to improve news recommendation. The main strategy of this method is to use the LDA model to deal with the news text first, and then fuse the SVD++ model with implicit feedback in SVD. We studied the effect of timeliness on the news using time window analysis. Finally, we used weighting similarity for the SVD++’s recommendation sequence. Our results indicate that the method function well for news recommendation.

2. RELATED WORKS

The recommendation system is a way to determine users’ interest. News recommendation system is one of
the major applications of a recommendation system in the field of journalism. Tang made a detailed summary and analysis of the characteristics of news recommendation processes. News recommendation is a complex application but lacks high recommendation accuracy. Different solutions were originally proposed. Collaboration filtering remains one of the most successful recommendation system and is still widely used today. Basically, collaborative filtering methods use ratings to recommend events useful to users. For example, Wang et al. proposed a news recommender based on users’ multidimensional tastes. This approach used multidimensional vectors to characterize the user’s interest and analyzed the impact of leader-follower network structure and quality factors. Although this method incorporated the influence of news, it ignored many other important factors, such as news’ topics.

Chen et al. used multiple user profiles to present user’s interests and their algorithms effectively improved the precision of recommendation. Yang et al. used a memory model focused on news’ subject and proposed an improvement plan which depended on user access to determine the different interest degrees based on frequency attenuation speed. Wu et al. also proposed a new topic detection and topic-based hot news recommendation algorithm combining LDA and Affinity Propagation (AP). Ba et al. proposed an approach combining a clustering algorithm with SVD algorithm. They found the approach increased the efficiency and scalability of a system. Liu et al. proposed several methods for user behavior analysis and modelling based on the SVD++ model and showed the method was effective (Gemci and Peker, 2013). These methods reflected changes in users’ interest, but they ignored the timeliness of news. The recommendation of outdated news to users significantly reduces the recommendation accuracy.

Song et al. investigated the effect of a time window on recommendation algorithm and found adapting approximately 12.56% recent rating records, the calculation complexity could be significantly reduced. Yang et al. also found that time can impact users’ interests. With time, different users’ preferences can drift dynamically, and each individual user showing interest in an item may be occur in the same time segment. They improved quality of recommendation system by adding time weight. Most approaches to study recommendation systems used LDA, SVD++ or other machine learning algorithms. However, only a few approaches utilized both LDA and SVD++. An algorithm will have lower precision if using only a LDA model and will improve if a model can combine LDA and SVD++. Using this approach, we consider the important factor of news’ timeliness.

3. PROCESSING OF NEWS

3.1. Dealing with News Text by LDA

LDA is a three-layer Bayesian probability model (Guannan, Duanbing and Yan, 2013). The process of modeling is as follows:

1) Extracting word (w) from news text in the topics of news in accordance with:

\[ p(w_i) = \sum_{i=1}^{T} p(w_i|z_i = j)p(z_i = j) \]  

2) Computing the probability of words in the news text in accordance with:

\[ p(w|d) = \sum_{i=1}^{T} p(w|z = j)p(z = j) \]

3) Repeat the first two steps until all the words from the news text have been extracted.

Where, \( z_j \) is latent variable, representing the word belongs to the topic, \( p(w_i|z_i = j) \) represents the probability of \( w_i \) belongs to the topic \( j \), \( p(z_i = j) \) represents the probability of the topic \( j \) belongs to the document \( d \). The value of \( \alpha \) and \( \theta \) are given randomly. After classifying the news text, according to the probability of each word in the news text, the processing is as follows: For each news item that is browsed by a user \( (u) \), if the word comes out, using 5 multiplies the probability as new rating (we assume that the highest rating is 5).

3.2. Dealing with News Data by SVD

In SVD the existing news rating is used to determine the degree of the individual’s tastes for each factor, then news data is analyzed to forecast results. It can be expressed as follows:

\[ R_{m\times n} = P_{m \times m} \Sigma_{m \times n} Q_{n \times n}^T \]

Here, R is a rating matrix with m dimensions and n columns. The matrix of \( \Sigma \) has diagonal elements with descending order and other elements are zero. After the dimension reduction, the formula of SVD can be written as follows:

\[ R_{m \times n} \approx P_{m \times k} \Sigma_{k \times k} Q_{k \times n}^T \]

Here, k is the dimensions after reduction. The predicted rating is computed by:

\[ \hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u \]
Where $\mu$ is the global average rating, or the sum of ratings divided by the number of ratings. $p_u$ is a vector that matrix $P$ corresponds to a user $u$, $q_i$ is a vector that matrix $Q$ corresponds to an item $i$. $b_i$ represents the degree of vector $i$ deviating from $\mu$, $b_u$ represents the degree of vector $u$ deviating from $\mu$. $\gamma$ is a parameter.

We use stochastic gradient descent to get the solution function as follows:

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$ (6)

$$b_u \leftarrow b_u + \gamma(e_{ui} - \lambda \cdot b_u)$$ (7)

$$p_u \leftarrow p_u + \gamma(e_{ui} - \lambda \cdot p_u)$$ (8)

$$q_i \leftarrow q_i + \gamma(e_{ui} - \lambda \cdot q_i)$$ (9)

3.3 Dealing with News Data by SVD++

SVD++ includes implicit feedback, using users’ historical rating data and browsing data, items’ historical rating data, and browsing data as new parameters. It is predicted by:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T(p_u + \frac{1}{\sqrt{|R_{ui}|}} \sum y_j)$$ (10)

Using the method of stochastic gradient descent, the solution function is as follows:

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$ (11)

$$b_u \leftarrow b_u + \gamma(e_{ui} - \lambda \cdot b_u)$$ (12)

$$b_i \leftarrow b_i + \gamma(e_{ui} - \lambda \cdot b_i)$$ (13)

$$p_u \leftarrow p_u + \gamma(e_{ui} \cdot q_i - \lambda \cdot p_u)$$ (14)

$$q_i \leftarrow q_i + \gamma(e_{ui} \cdot (p_u + \frac{1}{\sqrt{|R_{ui}|}} \sum y_j) - \lambda \cdot q_i)$$ (15)

$$y_j \leftarrow y_j + \gamma(e_{ui} \cdot \frac{1}{\sqrt{|R_{ui}|}} q_i - \lambda \cdot q_i)$$ (16)

Where, the meaning of each symbol is the same as in the SVD algorithm.

![Figure 1. Program flowchart](image-url)
3.4. Algorithm Procedure

After sorting news data by SVD, we can easily find the significance of the three matrices P, Σ, and Q.

<table>
<thead>
<tr>
<th>Table 1. The significance of the matrix for news recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>P</td>
</tr>
<tr>
<td>Σ</td>
</tr>
<tr>
<td>Q</td>
</tr>
</tbody>
</table>

Network news has a large amount of information, short reading time, and the news content is updated rapidly. These factors increase the difficulty and uncertainty of news recommender because of a constantly changing number of news topics. To reach convergence requires multiple iterations of the LDA algorithm. We use rating as the input in our algorithm.

From Table1, we see that news are associated with users in the matrix Σ. Users’ implicit features include the number of clicks, residence time on webpages, and conversion rate (Qilong, Xiaoyong and Zhongying, 2013). News’ implicit features include time, authenticity and times viewed. Time is the most important feature. There are a lot of uncertainties in these characteristics, or data noise. SVD can filter data noise to boost the accuracy of news recommendation result. As a supplement to SVD, we used SVD++ to recommend news. Fig.1 is the flow chart of our algorithm.

3.5. Optimizing Algorithm Considering News Timeliness

As a special commodity, the renewal speed of news is quite fast. Time the decisive factor in the measurement of news. How to deal with time is the key to improve the accuracy of a recommendation algorithm. Here, we grouped users in accordance with user_id in the rating matrix. We sorted the news according to news browsing time, with the most recently browsed items ranked first. In the matrix Q, time is one of the most important hidden features. Because we do not know the actual value of the feature in advance, we must adjust the time window to observe the influence of timeless news on the news recommending process. Using time windows is a way to divide the training set. This can be done using division into absolute time windows or relative time windows. Absolute time windows require the start and end time value to train algorithm in the range. The relative time window approach divides based on selection of a certain percentage of training set (e.g. 15%), then uses these divisions to train the algorithm and recommend. In order to get more accurate results, here, we used the absolute time window. To make the algorithm more effective, we used weighting. The basic idea is as follows: since the similarity is sorted by descending order, some news may be at the top but is browsed less in the recommended sequence. Therefore, it is feasible to deal with the similarity sequence by weighting. To do this, for each news item, we counted the number of times it was viewed (SumOfRead) and the most frequently viewed news items (MaxNewsRead). According to the number of times browsed, the score of the corresponding news item was weighted as follows:

weight = \frac{\text{SumOfRead}}{\text{MaxNewsRead}} \quad (17)

4. EXPERIMENTAL EVALUATION

4.1. Dataset and Pre-processing

Here, we used crawler technology to obtain randomly selected news browsing records from 10,000 users in March 2014 from a well-known financial news website in China. We used this dataset to evaluate our methods. Each news record includes user_id, news_id, browsing time, and news content (the user_id was kept anonymous).

The purpose of the experiment is to predict the last news viewed by the 10,000 users. Here, we selected the one before the last news as the test set, and others as the training set for the model. Firstly, we pre-processed the dataset as follows:

1) Remove the news’ theme which is empty or “404 Not Found”.
2) Delete the users with greater than nine user_ids, as in the dataset, they are abnormal, the user_ids only nine digits. One problem is the random missing data because the number of browsed news items for each user are variable. In the rating matrix, the column gives the biggest news browsing number based on the sorting time. There were some users who didn’t browse any news, and their scores were set to zero in the rating matrix. These scores for news by users read can be written as p(w|d)×5.

4.2. Experimental Metrics and Evaluation Method

Three metrics and their shifts were used to evaluate the algorithms: mean absolute error (MAE), converge, and F-measure. These are widely used metrics. For recommendation systems, MAE represents the accuracy of the predicted rating, an important metric for users. Converge indicates how well the system discovers other items...
that are desirable to a user. Converge is similar to the recall for an information retrieval domain; however, converge does not make sense if the recommendation precision is too low. Therefore, the comprehensive indicator, F-measure, which considers both precision and recall, is used to measure the recommendation algorithm accuracy. Here, we used F-measure (F) as our evaluation method. F considers both precision (P) and recall (R) of the test as follows:

\[ F_\beta = \frac{(1+\beta^2)(P \cdot R)}{\beta^2 P + R} \]  \hspace{1cm} (18)

Where P is precision:

\[ P = \frac{\sum_{u \in U} \text{hit}(u)}{\sum_{u \in U} \text{T}(u)} \]  \hspace{1cm} (19)

R is recall:

\[ R = \frac{\sum_{u \in U} \text{hit}(u)}{\sum_{u \in U} \text{T}(u)} \]  \hspace{1cm} (20)

Accuracy is the important metric for recommendation systems. In order to fully evaluate the algorithm, we must combine P and R. Here, U is the set of users in the dataset, \( \text{hit}(u) \) represents the number browsed by users in recommending news, \( \text{T}(u) \) represents the length of the recommended list to the user \( u \), \( \sum_{u \in U} \text{T}(u) \) represents the number browsed by users in the test set. \( \beta \) (beta) is a regular known value of 0.5, 1, or 2.

There are three methods commonly used to calculate similarity: Related Cosine Similarity (RCS), Pearson Correlation Coefficient (PCC), and Adjusted Related Cosine Similarity (A-RCS). RCS is one of the most commonly used methods, so, we adopted RCS to measure the recommendation algorithm as follows:

\[ \text{sim}(i,j) = \cos(i,j) = \frac{\sum_{u \in I} r_{u,i} r_{v,j}}{\sqrt{\sum_{u \in I} r_{u,i}^2 \sum_{u \in I} r_{v,j}^2}} \]  \hspace{1cm} (21)

Where i and j represent two news, \( r_{u,i} \) and \( r_{v,j} \) respectively represent the rating user u and user v to news i, and \( I_{uv} \) represents the rating set of user u and user v.

4.3. Experimental Result and Analysis

We evaluated the changes of precision, recall, and F in different time windows.

Table 2. The experimental results for different time windows

<table>
<thead>
<tr>
<th>Number</th>
<th>Time Window</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(-160,-130)</td>
<td>0.30</td>
<td>0.60</td>
<td>0.40</td>
</tr>
<tr>
<td>2</td>
<td>(-130,-100)</td>
<td>0.46</td>
<td>0.91</td>
<td>0.61</td>
</tr>
<tr>
<td>3</td>
<td>(-100,-70)</td>
<td>0.65</td>
<td>1.29</td>
<td>0.86</td>
</tr>
<tr>
<td>4</td>
<td>(-70,-40)</td>
<td>2.76</td>
<td>5.52</td>
<td>3.68</td>
</tr>
<tr>
<td>5</td>
<td>(-40,-10)</td>
<td>6.74</td>
<td>13.48</td>
<td>8.99</td>
</tr>
<tr>
<td>6</td>
<td>(-10,20)</td>
<td>6.64</td>
<td>13.27</td>
<td>8.85</td>
</tr>
<tr>
<td>7</td>
<td>(20,50)</td>
<td>0.74</td>
<td>1.48</td>
<td>0.99</td>
</tr>
<tr>
<td>8</td>
<td>(50,80)</td>
<td>0.77</td>
<td>1.54</td>
<td>1.03</td>
</tr>
<tr>
<td>9</td>
<td>(80,110)</td>
<td>0.70</td>
<td>1.40</td>
<td>0.93</td>
</tr>
</tbody>
</table>

Figure 2. Precision distribution histogram for different time windows
Table 2 summarizes the results shown in fig.2 to fig.4. The time window has a great impact on the recommendation accuracy. Precision, recall, and F-measure follow a normal distribution. Using six time windows of (-160, -130), (-130, -100), (-100, -70) and (20, 50), (50, 80), and (80, 110), the algorithm’s precision, recall, and F-measure were very low. We can predict that most recommended news are located in (-70, 20). The experimental results showed that the efficiency of the algorithm is the highest in that interval. Therefore, the next experiments were carried out for that time window (-70, 20). We found that beta can affect F-measure over a range of values.

As can be seen from Fig. 5, under the same experimental conditions, SVD and SVD++ both outperform LDA. Because SVD++ is based on SVD and also includes feedback about historical browsing and rating data for users and for news, the SVD++ is better than SVD. As beta increases, the weight of the recall in the measure increases. When beta=1, the F-measure is the harmonic mean of the precision and the recall.

As can be seen from Fig. 6, the weighted similarity can improve the accuracy of the algorithm. Due to the stochastic gradient descent component of the method the time complexity of the algorithm is increased.
5. Conclusions

With the popularity of the Internet, the problem of information overload has developed. The ability to determine users’ potential interest and recommend accordingly will allow great social and economic value. Here, we processed a dataset, solving the problem of having no scores. LDA is commonly used to process text, but has a large cost. For time windows, we weighted the recommendation sequence from the SVD++ algorithm, which improved the accuracy of results, but also increased the time complexity of the entire algorithm. There are several avenues of research for future efforts: 1) at the level data processing, classify into likes and dislikes according to browsing, not only simply set these scores to zero; 2) investigate alternative weighting methods; 3) develop an improved machine learning algorithm to improve the recommendation accuracy and decrease the complexity of the algorithm.

ACKNOWLEDGMENTS

The first author acknowledges the financial support by Science and Technology Project of Department of Science and Technology of Guizhou Province (China) (QJH LKM[2011] 15). The second and third authors acknowledge the financial support by Science and Technology Project of Department of Science and Technology of Guizhou Province (China) (QKHJZ[2014] 7194). The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

REFERENCES