Research Article

User Personalized Recommendation Algorithm Based on GRU Network Model in Social Networks

Fangqin Zeng, Rong Tang, and Yibai Wang

School of Information Engineering, Changsha Medical University, Changsha, Hunan 410219, China

Correspondence should be addressed to Yibai Wang; hunanhappywang@163.com

Received 29 March 2022; Revised 19 April 2022; Accepted 27 April 2022; Published 17 May 2022

Academic Editor: Fazli Wahid

Copyright © 2022 Fangqin Zeng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Efficient and accurate personalized recommendation algorithms can effectively improve user experience satisfaction, in order to improve the performance of user personalized recommendation algorithm, this study proposes a user personalized recommendation algorithm based on deep learning network. The algorithm uses Gate Recurrent Unit (GRU) network to build the main model of personalized recommendation algorithm to reduce the influence of over fitting of multi-layer network; The attention mechanism is introduced into GRU network, so that the recommendation model can obtain the feature information of user data more accurately and reduce the influence of irrelevant data on the model; At the same time, due to the introduction of variable length mini-batch allocation method, the model training data is more complete and reliable, which can effectively improve the accuracy of user personalized recommendation. The simulation experiment is based on Amazon dataset and MovieLens dataset. The experimental results show that the proposed method has good personalized recommendation ability.

1. Introduction

In recent years, with the continuous enrichment of the content forms of information media, the continuous improvement of communication technology, and the continuous acceleration of communication speed, in the wave of the Internet, the amount of information generated every day shows a “blowout” growth, hundreds of millions of new information will be generated every day, and people have ushered in the era of “big data.” In the era of “big data,” people can browse a wide variety of information and fully meet everyone’s personalized needs, but the information overload has also become a thick barrier for people in information screening [1, 2]. Personalized recommendation system can effectively alleviate the problem of excessive information, deeply mining user preferences, and provide decision-making suggestions for service providers. It has become the preferred goal of many network service platforms to provide users with personalized services and assist in decision-making [3–5]. For example, large online news platforms such as Netease News, Tencent News, and today’s headlines will regularly collect a large amount of news information from multiple information channels and platforms. Every day, a huge amount of news data will flow into the background database. It is obviously impractical for users to manually filter the content they are interested in from the massive data [6, 7]. Therefore, in recent years, countless researchers and scholars have invested in the research of personalized recommendation algorithm, constantly introducing new ideas and technologies to optimize the algorithm, improve the accuracy of recommendation, and alleviate the problem of information overload.

Traditional recommendation algorithms are mainly represented by collaborative filtering model. The most common collaborative filtering algorithms mainly include neighborhood method and matrix decomposition model, which can achieve good recommendation effect when there are few data samples [8]. However, the current network data has the characteristics of complex and high-dimensional. The traditional methods cannot support the user information extraction behavior in the environment of large data samples, which has the problem of low efficiency of data information extraction.
The deep network model based on multi-layer network structure can continuously extract and learn dynamic user data information and build corresponding data analysis model to realize stable and accurate user personalized recommendation [9]. However, at present, most networks do not consider the characteristics of data samples, so it is difficult to achieve more in-depth and accurate information extraction, which is not enough to meet the completeness and reliability of the network model, which also makes the personalized recommendation network model have a lot of room for improvement. To solve the aforementioned problems, this study proposes a personalized recommendation algorithm based on deep learning. The innovations of the algorithm are as follows.

1. In this study, the attention mechanism module is used to optimize the GRU network model, which can effectively identify and extract the useful feature information in the user data, improve the information processing efficiency and accuracy of the proposed network model, and realize accurate and efficient user personalized recommendation;

2. Considering the complexity and diversity of data samples, this study uses the optimized variable time length mini-batch allocation method to optimize the data processing, so as to adapt to the features of the variable length of user behavior sequence, and further improve the analysis and processing efficiency of the algorithm while ensuring the completeness of data.

2. Related Works

Personalized recommendation belongs to an information filtering system, which can filter different features information in a specific scene. Personalized recommendation plays an important role in various industries, which can promote users’ decision-making process and generate benefits more efficiently [10, 11]. The core of personalized recommendation is to accurately locate users’ points of interest [12, 13]. Therefore, whether to accurately extract users’ hidden preferences from limited user information has become the key to measure the advantages and disadvantages of a recommendation algorithm, and it is also the focus of research. Traditional personalized recommendation algorithms can be divided into two categories: content-based recommendation algorithm and collaborative filtering recommendation algorithm.

By analyzing the common attributes of all the contents browsed by the same user, the content-based recommendation algorithm filters out the candidates that are most closely related to the contents browsed by the user and feeds back to the user [14]. Collaborative filtering recommendation algorithm reflects the rating information between users and all contents in the form of matrix, and fills the rating information between users and contents again in the low-dimensional space to realize users’ personalized recommendation [15]. However, when dealing with data scenes with high sparsity, the traditional recommendation algorithm has a very cumbersome calculation process and high time complexity, and the accuracy and availability of the algorithm in complex situations are also limited [16, 17].

In recent years, deep learning has brought great changes to the recommendation algorithm, and its emergence provides a better method to improve the performance of the recommendation system [18, 19]. The recommendation system based on deep learning overcomes the technical obstacles of the traditional recommendation mode and achieves higher recommendation quality, so it has attracted extensive attention. Reference [20] realizes Basketball Teaching Network Course Resource Recommendation Based on convolutional neural network (CNN); Reference [21] constructs a learner model based on long short-term memory network (LSTM), which can effectively reduce the cold start and sparse problems in the recommendation process and realize the recommendation of individual learning resources; Reference [22] first learned the citation context and the distributed representation of scientific papers based on LSTM network, and then measured the correlation based on the learned citation context and the feature information of scientific papers to realize the recommendation of scientific papers; Reference [23] uses CNN network and Bi-LSTM network to extract user data information and improve the performance of network, so as to realize targeted recommendation for network users. However, it should be noted that the above recommendation algorithm does not consider the features of user data itself, and cannot effectively extract feature information from complex and diverse user data samples. Therefore, the above recommendation algorithm is difficult to build a complete and reliable personalized recommendation network.

To solve the above problems, this study introduces the attention mechanism and variable time mini-batch allocation method to the GRU network, so as to propose an efficient and accurate user personalized analysis and recommendation algorithm to improve user experience satisfaction.

3. Recommendation Model Based on Deep Learning

In order to achieve accurate and efficient user personalized recommendation, this study uses the attention mechanism module to optimize the GRU neural network model, reduce the over fitting problem in the multi-layer network structure, and extract the user data features orderly and accurately, to improve the performance of the recommendation algorithm.

3.1. GRU Neural Network. LSTM shows great advantages in dealing with the emotional classification of corpus with the combination of long and short texts. However, the internal unit structure of LSTM is complex and there are many optimization parameters, so the training time is long, and there may be over fitting for the short text corpus [24]. GRU network model is an effective variant of LSTM network model. Compared with LSTM model, the internal unit structure of GRU network model is relatively simple, and it
also retains the long-term memory function of LSTM structural. The internal unit structure of GRU model is shown in Figure 1.

As shown in Figure 1, like the LSTM network model, the GRU network calculates the hidden layer output \( g_t \) at the current time based on the input \( r_t \) at the current time and the hidden layer output \( g_{t-1} \) at the previous time. However, the internal unit structure of the GRU network is relatively simple, consisting of only update gate \( u_t \) and reset gate \( v_t \). The update of the internal unit of the GRU network is shown in the formulae (1) to (4).

\[
\begin{align*}
v_t &= \sigma(w \cdot [g_{t-1}, r_t]), \\
u_t &= \sigma(w \cdot [g_{t-1}, r_t]), \\
\tilde{g}_t &= \text{Tanh}(w \cdot [v_t \cdot g_{t-1}, r_t]), \\
g_t &= (1 - u_t) \cdot g_{t-1} + u_t \cdot \tilde{g}_t,
\end{align*}
\]

where \( w \) represents the weight matrix, \( \sigma \) is the activation function in the gate calculation results, and sigmoid function is generally used for binary classification problems. The update gate \( u_t \) is used to control the extent to which the state information of the previous time is added to the current time, the larger the value of the update gate, the more information added to the current time; The reset gate \( v_t \) is used to control the extent to which the state information of the previous time is written into the candidate set \( \tilde{g}_t \) of the current state, the smaller the value of the reset gate, the less information to enter the candidate at the previous moment.

3.2. GRU Based on Attention Mechanism. With the continuous influence of the external environment and the gradual passage of time, users' interests and hobbies are also evolving. At the same time, users' interests and hobbies may also be concentrated in a certain period of time. Therefore, in order to model this change of user interest, it is not enough to use only the original one-layer GRU.

The data extraction layer can obtain the interest feature expression of user interaction sequence by using the auxiliary loss function. However, in order to model the evolution trend and concentration of user interest, it is necessary to add a layer of GRU with attention mechanism on the basis of the original GRU network. Combined with the local activation ability of attention mechanism and the sequence learning ability of GRU, the actual evolution of interest is modeled, and the local activation performance simulates the concentration of interest and weakens the impact of irrelevant interest on the final result of the model [25].

The calculation method of attention \( q_t \) is shown in formula (5), where \( s_q \) is the combination of embedding vectors of candidate goods, the model parameter matrix is \( c \in \mathbb{R}^{n_2 \times n_3} \), \( n_2 \) is the dimension size of hidden layer, and \( n_3 \) is the dimension size of embedding vectors of candidate goods. The value of attention \( q_t \) can reflect the correlation between candidate goods and input \( r_t \).

\[
q_t = \frac{\exp(r_t^T s_q)}{\sum_{i=1}^{n} \exp(r_t^T s_q)}.
\]

At present, most scholars use the attention score to directly control the update state of the hidden layer, but they use the attention score \( q_t \) to directly replace the update gate \( u_t \), ignoring the differences between different dimensions.

This study uses the update gate with attention mechanism to seamlessly combine the attention mechanism with GRU, and constructs the GRU with attention update gate (AUG-GRU). Figure 2 shows the specific gating structure of AUG-GRU network.

In the proposed AUG-GRU network, the update gate retains the dimension information of the original input, and scales the attention score to all the update gate dimensions, so that the irrelevant input has the least impact on the model. In other words, AUG-GRU can more effectively avoid the interference caused by the sudden change of interest to the model, and promote the interest extraction of the model more smoothly. The change of AUG-GRU to GRU is shown in the formulae (6) and (7), where \( u_t \) is the update gate of the original GRU, \( \tilde{u}_t \) is the update gate of the redesigned AUG-GRU, \( g_t \) and \( g_{t-1} \) are the hidden layer state of AUG-GRU.

\[
\begin{align*}
\tilde{u}_t &= q_t \cdot u_t, \\
g'_t &= (1 - \tilde{u}_t) \cdot g'_{t-1} + \tilde{u}_t \cdot \tilde{g}_t.
\end{align*}
\]

The structure of deep interest evolution network is shown in Figure 3.

Compared with the traditional recommendation system model, the biggest improvement of the proposed model is to use GRU network model to extract user sequence interest. At the same time, in order to further adapt to the evolution trend of users' interest, a variant GRU structure with attention mechanism is used to ensure the accurate extraction users' sequences features.

3.3. Indefinite Length Sequence Mini-Batch Allocation. When training neural network model, gradient descent algorithm is often used to minimize the loss function. The random gradient descent method is updated according to
only one sample at a time, which has fast calculation speed and low memory requirements. However, due to the use of only one sample at a time, it has great volatility in the gradient descent process, and it is difficult to use the parallelization method to improve the speed. The small batch gradient descent method combines the advantages of batch gradient descent method and random gradient descent method. The small batch gradient descent method selects several samples each time for parameter updating, which can not only use the parallelization method to process the samples, but also carry out gradient descent and update relatively smoothly each time, but also ensure the speed of training. Therefore, this study uses the small batch gradient descent method for parameter updating. Hereinafter, the number of samples selected each time is called batch size.

In order to adapt to the variable length of the input user behavior sequence, a flexible mini-batch allocation method is used in the AUG-GRU model, so that the mini-batch at each time contains not only batch size commodity attributes, but also the length of attention factor corresponding to the viewed commodity. This allocation method ensures time efficiency without data loss, so as to meet the requirements of AUG-GRU network for input data.

Taking the browsing sequence of some users as an example, each sequence on the left of the arrow in Figure 4 represents the browsing information of a user in the dataset. The commodities in the sequence are arranged according to the browsing order of the user. The rectangular width of each commodity represents the length of time the user browses the commodity, and $y_{x,j}$ represents the $j$ commodity in the $x$ user sequence. Firstly, the user’s duration information is transformed into duration attention factor, the original attributes of goods in the sequence are retained, and then the data is composed of mini batch. The processed data is shown in the sequence on the right side of the arrow in Figure 4. The rectangular width corresponding to the processed data no longer represents the duration information, but uses $q_{x,j}$ to represent the duration attention factor corresponding to commodity $y_{x,j}$.

During the allocation of mini-batch, the information corresponding to the first browsing commodity of batch-size series is randomly selected from the training set at time $t$ to form the mini batch at time $t$, which is input into the AUG-GRU network for training. The expected output of the $t$ time network is the commodity number of the next time in each sequence. At time $t$, after all data in mini batch are trained, the hidden state of AUG-GRU network is obtained.

At time $t'$, if there are still Commodities in the sequence corresponding to position $j$ in the mini-batch that have not been trained, select the commodity vector corresponding to the expected output of the position in the mini batch at time $t$ and the duration attention factor of the commodity as the input of the position at time $t'$, and obtain the hidden state corresponding to position $j$ at time $t'$ through the network hidden state trained at time $t$ and the input information trained at time $t'$. If the sequence corresponding to position $j$ in mini-batch at time $t$ has been trained, randomly select a sequence from the untrained sequence in the training set, take the commodity attribute and long-term attention factor of the first browsing commodity in the sequence as the input of position $j$ in mini batch at time $t'$, reset the network hiding state corresponding to position $j$, and only use the input of position $j$ at time $t'$ for training.

The allocation method of mini-batch at the next time is analogized until all samples in the training set are trained. The specific implementation method is shown in Figure 5.

### 3.4. Model Training

The data often used in recommendation systems is usually implicit feedback data, such as browsing history, user behavior log, and so on. These data often have only positive feedback but no negative feedback, and the data is highly sparse.

At present, the recommendation system based on deep learning technology usually adopts negative sampling technology in order to ensure efficiency and speed when training the model. The experiment also uses negative
sampling technology to construct the training set, and marks the news that users have not browsed as negative samples. The loss function of the proposed model is as follows:

$$\mu_k = \frac{\exp(h_{xz})}{\exp(h_{xz}) + \sum_{y=1}^{Y} \exp(h_{yx})},$$

$$\text{loss} = - \sum_{k=1}^{K} \mu_k,$$

where $h_{xz}$ is the probability that user $x$ browses the $z$ positive sample, $h_{yx}$ is the probability that user $e$ browses the $y$ negative sample, $Y$ is the random negative sampling coefficient, and $K$ is the total number of positive samples browsed by user $x$.

During the model training, the verification set can be used to monitor the model training and adjust the pre-set parameters in time to ensure the training efficiency of the model. For each user in the training set, four negative samples will be paired with one positive sample in this experiment.

In this experiment, the stochastic gradient descent (SGD) method is used to continuously optimize the parameters in the network to make the model achieve better prediction effect. SGD is a better way to calculate the gradient in the field of machine learning. The training goal of the model is to minimize the global loss function. The difficulty is that there are too many global samples and the cost of calculating the gradient in each global iteration is too high.

SGD will extract a subset of samples from the training set every time to complete a single iteration. Repeat a single training iteration for many times and take the minimum value of the multiple random gradient optimal solution, which can be regarded as the global optimal solution. The learning speed of SGD is fast and can realize the effect of real-time update. The disadvantage is that the learning rate used for multiple sampling is unique. In the face of parameters that need to be greatly adjusted and fine-tuning parameters, the learning rate cannot achieve dynamic adaptation. In order to solve the shortcomings of SGD, this study uses Adam to dynamically adjust the learning rate, so as to optimize the training process and improve the model accuracy.

### 4. Example Verification and Result Discussion

In order to verify the effect of the model, Tensorflow, a deep learning framework provided by Google, is used to build the model. Tensorflow framework integrates models such as GRU, which can make development simple. Therefore, it has become a popular deep learning framework. The specific experimental environment configuration is shown in Table 1.

Due to the large amount of experimental data required, open datasets are used for experimental analysis. The datasets used mainly include Amazon dataset and MovieLens dataset.

Amazon dataset includes 233.1 million product reviews and related data from Amazon from May 1996 to 2018. Due to the dataset is too large, this experiment mainly uses two parts: books and electronic products. Book Product data includes 5131621 articles and 2935525 commodity data; the data of electronic products includes 20994353 records and 786868 commodity information.

The MovieLens dataset mainly contains users’ relevant ratings and comments on different films, with a total of 20000 260 comments, involving a total of 20729 films, 133814 users, and 21 film categories. MovieLens dataset contains a wide range of data, so it is often used to test recommendation systems and machine learning algorithms.
Table 1: Experimental environment configuration.

<table>
<thead>
<tr>
<th>Environment</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Ubuntu 18.04.3 LTS</td>
</tr>
<tr>
<td>CPU</td>
<td>Intel core i7-4500U 2.4 GHz</td>
</tr>
<tr>
<td>GPU</td>
<td>GeForce RTX3080 Ti</td>
</tr>
<tr>
<td>Memory</td>
<td>32 GB</td>
</tr>
<tr>
<td>Language</td>
<td>Python 3.6</td>
</tr>
<tr>
<td>Platform</td>
<td>Tensorflow</td>
</tr>
</tbody>
</table>

4.1. Evaluating Indicator. In this study, Recall@β and MRR@β are used as the performance evaluation indicator of the recommended algorithm.

4.1.1. Recall@β. After the AUG-GRU model uses the user’s historical browsing information to get the predicted interest score of the user on each commodity at the next time, the effect of the model needs to be evaluated according to the predicted results. The content that the recommendation system can recommend to users is limited. Usually, it can only recommend some products with high scores to users. Therefore, one of the evaluation indexes adopted is whether the β commodities with the highest prediction score of each user include the commodities that users are really interested in, expressed in Recall@β.

If the β items with the highest score include the items that the user actually browses at the next time, the prediction is deemed to be correct, and the Recall@β value of the user is 1. If not, it is deemed to be a prediction error, and the Recall@β value of the user is 0. When the product actually viewed by a user ranks rank_α in the score predicted by the model, the Recall@β value expression of the user is as follows:

\[
\text{Recall}_@β = \begin{cases} 
1, & \text{if rank}_α ≤ β, \\
0, & \text{if rank}_α > β.
\end{cases}
\]  

(10)

4.1.2. MRR@β. The final result of Recall on the model is the average Recall@β value of all users in the test set. Recall@β can only evaluate whether the first β commodities predicted by the model contain the real browsing commodities, while ignoring the evaluation of the ranking position of the predicted commodities. However, in the actual application of personalized recommendation system, the more front-end products are recommended, the more they can attract users’ attention. Therefore, this study also uses MRR (mean reciprocal rank) as the evaluation index of the model.

For a user, the MRR is the value obtained by taking the reciprocal of the ranking position rank_α where the goods are actually browsed. The specific position of the goods with low ranking is almost meaningless to the recommendation system. Therefore, this study limits the ranking threshold β for MRR value. When the actual products are in the β products with the highest prediction score, the MRR@β is the reciprocal of its ranking, and when the ranking exceeds β, the value of MRR@β is 0. The specific calculation method of MRR@β value of each user in the test set is as follows:

\[
\text{MRR}_@β = \begin{cases} 
\frac{1}{\text{rank}_α}, & \text{if rank}_α ≤ β, \\
0, & \text{if rank}_α > β.
\end{cases}
\]  

(11)

4.2. Optimal Design of Recommended Network. In this study, the experimental dataset is divided into training dataset and test dataset according to the ratio of 4:1. This study optimizes the design of the model to ensure that the AUG-GRU network model realizes accurate and efficient personalized recommendation. In the initial training, the forward layer of AUG-GRU model is 256 neurons, the learning rate is 0.0015, the dropout value is 0.25, the number of training epochs is 25, and the batch_size is 64. Adam is the model optimizer, and Sigmoid function is the activation function. The test loss function of the first trained model for Amazon dataset and MovieLens dataset is more than 0.01945. Therefore, this study carries out finetune on the initial model to improve the accuracy of the model. After several adjustments, the loss function and accuracy change trend of AUG-GRU network on the training dataset are shown in Figure 6 respectively.

As can be seen from Figure 6, the loss function of the test set of AUG-GRU network is lower than about 0.0066 after 30 iterations for Amazon dataset and MovieLens dataset. On the whole, AUG-GRU network model shows good fitting effect. Therefore, the parameters of AUG-GRU network are determined as the optimal parameters. The specific parameter setting of this experiment is that the forward and backward layers of AUG-GRU layer contain 256 neurons, respectively, the dropout is 0.30, the batch_size is 32, and the epoch is 30. Adam is the model optimizer and sigmoid function is the activation function.

4.3. Analysis of Test Results. In order to verify the detection effect of the proposed model, comparative experiments were carried out with the models proposed in References [20, 23]. Figure 7 shows the experimental results based on Amazon dataset.

As can be seen from Figure 7, the evaluation indicator Recall@β and MRR@β of the proposed method in this study are 0.8709 and 0.7954, respectively, which are higher than the evaluation indexes 0.022 and 0.0635 in Reference [20], indicating that the proposed model does have a better ability to model the user’s interaction behavior for the next recommendation. Figure 8 shows the experimental results based on the MovieLens dataset.

As can be seen from Figure 8, for evaluation indicator Recall@β, the proposed method is 0.0219 higher than Reference [23] and 0.026 higher than Reference [20]; for evaluation index MRR@β, the proposed method is 0.0256 higher than that in Reference [20]. In conclusion, the personalized recommendation algorithm of the proposed AUG-GRU network model has high user information fitting and product specific recommendation ability. The reason is that compared with GRU model, the network structure of Reference [20] network model is more complex, and there is a
problem of network over fitting; Reference [23] model cannot accurately capture the features of sample data accurately and orderly, and there is a problem that irrelevant information interferes with the accuracy of recommendation. While overcoming the above difficulties, the proposed method adopts GRU network with attention mechanism to extract the feature information of sample dataset accurately and orderly; k—hemini-batchallocationmethodwithvariable time length can better match the feature of complex and diverse sample data, improve the analysis ability of the overall algorithm, and support the algorithm to achieve efficient and accurate information recommendation.

5. Conclusion

In order to improve the accuracy of personalized recommendation function in social networks, this study proposes a personalized recommendation algorithm based on improved GRU network. Considering the complexity and diversity of data samples, this study uses the optimized variable time length mini-batch allocation method to optimize the data processing, so as to adapt to the features of the variable length of user behavior sequence, and further improve the analysis and processing efficiency of the algorithm while ensuring the completeness of data.

Although the proposed model can realize accurate personalized information recommendation, its parameters are fixed values, which is difficult to adjust automatically according to the data characteristic information. The next research work is to introduce the parameter adaptive algorithm into the algorithm to enhance the ability of parameter optimization and further improve the efficiency of the algorithm.

Data Availability

The data used to support the findings of this study are included within the article.
**Conflicts of Interest**

The authors declare that there are no conflicts of interest regarding the publication of this study.

**Acknowledgments**

This work was supported by the Scientific Research Fund of Hunan Education Department (No. 21C0868 and 20C0223).

**References**


