JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 1437-1457 (2023)
DOI: 10.6688/JISE.202311_39(6).0012

Hybrid Generative Adversarial Networks for News Recommendation*

DUEN-REN LIU[1,+], YANG HUANG[1], CHING-YI TSENG[1] AND SHIN-JYE LEE[2]
[1]*Institute of Information Management*
[2]*Institute of Management of Technology*
National Yang Ming Chiao Tung University
Hsinchu, 300 Taiwan
E-mail: dliu@nycu.edu.tw[+]

Personalized recommendation systems offer rapid information access, especially for online news platforms. Generative Adversarial Network (GAN) models have been successfully adopted for recommendation. However, very few GAN-based recommendation systems consider the content of news articles, despite its essential latent features, which may affect a recommendation system's effectiveness. This study proposes a novel hybrid GAN recommendation model, which combines Long Short-Term Memory (LSTM), Matrix Factorization (MF) and latent feature extraction of textual content in the GAN structure. Existing GAN-recommendation combines MF to analyze users' global preferences and LSTM to capture users' dynamic preferences over time, but it does not consider the relative importance of MF and LSTM. To effectively combine LSTM and MF, this study proposes a novel attention mechanism for adjusting MF and LSTM weights by learning their importance. In addition, existing GAN-based recommendation does not consider item text content. Our proposed model improves the GAN model by combining the Collaborative Topic Modeling (CTM) and Convolutional Neural Network-Matrix Factorization (CNN-MF) in the Generator to enhance content feature extraction when deriving user and item latent vectors. Specifically, CTM is adopted to obtain the initial latent user/item features of the GAN model, and CNN-MF is used to enhance the extraction of potential text content features in the Generator. The proposed model can enhance existing GAN-recommendation, and increase the performance of preference predictions on textual content such as news articles. This study conducted experimental evaluations using the dataset collected from a news website. The experiment result shows that the proposed approach outperforms several baseline methods on real-world news recommendation.

Keywords: recommendation systems, collaborative topic modeling, latent Dirichlet allocation, convolutional neural network, matrix factorization, generative adversarial networks

1. INTRODUCTION

The sheer volume of news articles published daily on the internet, and their vast range of topics and content, has seen users come to rely on personalized news recommendations in order to access articles relevant to their interests [4, 12]. As increasingly more users rely on recommendation systems on news platforms to reduce their time spent browsing for articles they are interested in, the quality and effectiveness of such recommendation systems may increase their loyalty and willingness to utilize particular platforms.

Recommendation systems include content-based filtering [28, 33, 44] and collaborative filtering [5, 15, 29, 31]. The former mainly generates user and item feature vectors

Received October 16, 2022; revised November 9, 2022; accepted November 11, 2022.
Communicated by Churn-Jung Liau.
[+] Corresponding author.
* This research was supported by the Ministry of Science and Technology of Taiwan under Grant No. MOST 108-2410-H-009-046-MY2.

based on the content of an article, and calculates their similarity to predict items in which a user may be interested. The latter builds models based on historical interaction records of users and items, and includes matrix factorization (MF) [23]. However, due to data sparsity and the cold start problem, this method may be prone to unsatisfactory recommendations if a user has no, or very few browsing records. The Collaborative Topic Model (CTM) [35] addresses the cold start issue by integrating the MF and Latent Dirchlet Allocation (LDA) topic model [3], in which the latent topic vectors of text content derived from LDA are used as the initial latent factors of the MF.

In recent years, deep learning methods have been widely used in different fields. Deep learning methods in recommendations [2, 7, 11, 42] are mainly based on matrix decomposition methods, which use neural networks developed to learn hidden factors of users and items by calculating preference scores. Moreover, Generative Adversarial Networks (GANs) have achieved effective performance in image recognition [8, 13, 26] and natural language processing [39]. A GAN consists of two neural networks, a Generator and Discriminator. The two neural networks compete in a Minimax game. The Generator generates candidates, while the Discriminator judges them. Both neural networks optimize in an adversarial way until convergence. However, despite its good performance, relatively few papers have been published on GAN-based recommendation methods [2, 7, 36, 41, 42]. Of these proposed methods, IRGAN [36] uses the GAN Minimax game structure to train preference learning models, in which the kernel rating (preference) prediction model is based on traditional matrix factorization. Through the adversarial learning of Generator and Discriminator, the preference score of data close to the real value is gradually generated. However, the model does not consider the potential impact of users' dynamic preferences over time. RecGAN [2] further expands and modifies the GAN architecture, incorporates time series into the system, and uses Gated Recurrent Units (GRU) to capture hidden features of long short-term time profiles between users and items. PLASTIC [42] combines MF and Long Short-Term Memory (LSTM) in its GAN architecture to improve recommendation. It uses LSTM to trace user and item preferences over time, and uses MF to analyze users' global preferences. However, it does not take the text content of recommended items into account. Items' latent features are thus overlooked, which may lead to poor recommendations.

Other deep learning recommendation approaches extract latent features from the recommended items using recurrent neural networks (RNN) or convolutional neural networks (CNN). Still other models use deep learning methods to extract document feature vectors, and then combine them with MF to optimize the learning of the latent user and item factors. For example, ConvMF [21] combines convolutional neural networks with probability matrix factorization (PMF) for preference prediction.

At present, recommendation systems rarely use GAN as their core architecture. Although IRGAN was the first to propose training information retrieval models through a Minimax game architecture, it did not consider temporal dynamic changes of users and items. PLASTIC, although it is based on IRGAN, and includes LSTM to capture users' preferences over time, ignores the importance of item text content. Thus, considering the latent features of news content in order to find related news articles that users may like may further improve recommendation models.

This paper proposes a novel hybrid GAN recommendation model, which combines LSTM, MF and latent feature extraction of textual content in the GAN structure. To effecttively combine LSTM and MF, this study proposes an innovative deep learning attention mechanism in the Discriminator model. In addition, this study also combines the advantages of CTM, CNN, and MF for content feature extraction, then iteratively updates the latent vectors of users and items, and further updates the Generator according to the updated latent vectors. The proposed model can enhance existing GAN-recommendation, and increase the performance of preference predictions on textual content such as news articles.

This study uses a real-world news dataset taken from NiusNews to evaluate the experimental performance of the proposed model. Experiment results show that the proposed model is superior to traditional methods, including IRGAN, PLASTIC, CTM and ConvMF.

The main contributions of this article consist in integrating several methods, and proposing an innovative hybrid GAN-recommendation method combining CTM, CNN, MF and LSTM in the GAN architecture. Moreover, a novel attention mechanism is proposed to effectively combine LSTM and MF in the Discriminator, and consider the latent features of each item by applying CNN-MF in the Generator. The experiment result shows that the proposed approach outperforms several baseline methods on real-world news recommendation.

The remainder of this paper is organized as follows. Section 2 provides a brief overview of related work. Section 3 describes the proposed recommendation approach. Section 4 describes and analyzes the experiment and evaluation results. Finally, Section 5 offers conclusions and suggestions for future work.

2. RELATED WORK

This section introduces related work and the methods adopted in the proposed model.

2.1 Recommendation Systems

Recommender systems have been applied in various areas, such as products [22, 24] and news [6, 18, 25, 37]. Collaborative filtering (CF) methods use historical records to measure the similarity between users and items, which can improve the recommendation of unknown items to users according to their preferences [15, 29, 31, 35]. Content-based filtering (CBF) methods recommend similar items by creating user profiles, which convert users' personal data into feature files, and also take text content of items into account [28, 33]. Hybrid filtering methods apply a weighted combination of CF and CBF for recommendation [20, 32]. Latent factor models use matrix factorization (MF) techniques to discover the user and item latent factors [23], and then recommend items to a user if the correspondence between item and user factors is high.

In recent years, many related studies on recommendation systems based on deep learning methods have been conducted [10, 11, 15, 21, 33, 34]. Among them, Kim *et al.* [21] proposed Convolutional Matrix Factorization (ConvMF). ConvMF mainly uses different filtering layers of CNN (including an embedding layer, a convolutional layer, a merge layer and an output layer) to generate hidden article feature vectors. It then combines a PMF scoring matrix to learn the latent factors of users and items.

2.2 Latent Dirichlet Allocation and Collaborative Topic Modeling

The latent topic model explores the relationship between the words in the article content, and mines hidden topics in the article collection. A common hidden topic model is Latent Dirichlet Allocation (LDA) [3]. LDA is a probability generation model used to calculate the probability distribution of topics in an article in order to analyze the latent variables. In an article, some specific words can correspond to a certain topic, so the topic distribution of each article is a multinomial distribution. And in each topic, the word distribution is also a multinomial distribution, which can then be solved by expectation maximization (EM) algorithm.

Collaborative topic modeling (CTM) [35] combines LDA and MF, using LDA latent topic vectors as the initial latent factors of articles (items), and then MF to learn the latent factors for users and articles (items). Each article j will create its latent topic vector $\theta(j)$, and create latent factor vectors to users and articles, u_j and o_j' respectively, according to the rating information. The CTM method assumes the latent factor of an article to be $v_j = \theta_j + o_j'$. Therefore, the preference of user i for article j can be obtained by $u_i^T v_j$.

The traditional MF method is prone to cold start problems when encountering new items (no item rating information) or items with very few records, but when the CTM encounters cold start items, it can still make recommendations based on the latent topic vector θ_j of the cold start item.

2.3 Recurrent Neural Network and Long Short-Term Memory

Recurrent neural networks (RNN) [27] are a kind of neural network. However, a simple RNN has the problem of exponential weight explosion or the vanishing gradient problem, and struggles to capture long-term time correlation [30]. Long short-term memory (LSTM) can solve this problem, meaning RNNs can learn dynamic time behavior by cycling through states in their own networks. Long short-term memory (LSTM), a special architecture developed by Hochreiter and Schmidhuber [17], is an RNN method that is suitable for processing and predicting important events with very long intervals and delays in time series. LSTM can avoid the vanishing gradient problem due to its three-gate structure: input gate, forget gate, and output gate. These gates can determine whether to remove or add information to a cell, and thus can memorize previous events to make better use of its input.

2.4 Attention Mechanism

Vision researchers developed the concept of Visual Attention by investigating how human beings analyze images through their visual systems [16, 19]. It is a process that determines which regions of an observed thing should be selected for more detailed analysis. This is similar to the basic idea of attention mechanism [1, 34]. Attention mechanism allows a model to distinguish the importance of every input by allocating different weights, giving more weight to features that more significantly influence the output. As one of the most influential methods of deep learning, it is widely used in sequence-to-sequence models, such as speech recognition [9], image captioning [38, 40], and many other fields [43].

2.5 Generative Adversarial Networks for Recommender Systems

Generative adversarial networks (GANs) [14] consist of two models: a generative model G, and a discriminative model D. The two models can gradually be improved under iteratively adversarial competition: the Generator simulates the probability distribution of the real world, and the Discriminator distinguishes real and fake samples. This moves the Generator closer to the true target probability distribution, and produces realistic samples.

Wang *et al.* [36] proposed the IRGAN recommendation system for information retrieval applications. In [50], they define the relevant text that has been browsed as a positive sample, and the text that has not been browsed as a negative sample. The conditional probability $P_{true}(d|q_n, r)$ defines the underlying true relevance distribution, where q_n is a given query from a user (*e.g.*, keywords, questions, …, *etc.*), d is its corresponding document, and r denotes relevance. The Generator retrieval model $P_\theta(d|q_n, r)$ tries to approximate the true relevance distribution $P_{true}(d|q_n, r)$, while the Discriminator retrieval model $f_\varnothing(q, d)$ tries to distinguish between relevant documents (labeled 1) and non-relevant documents (labeled 0). The overall objective function is shown as Eq. (1), where $D(d|q_n)$ is a sigmoid function of the Discriminator score:

$$J^{G^*,D^*} = \min_{\theta} \max_{\varnothing} \sum_{n=1}^{N} (E_{d\sim p_{true}(d|q_n,r)}[\log D(d \mid q_n)] + (E_{d\sim p_\theta(d|q_n,r)}[\log(1 - D(d \mid q_n))]). \quad (1)$$

Zhao *et al.* [42] proposed the PLASTIC GAN model structure combining MF and LSTM, and considered users' long short-term preferences to improve recommendations. They used MF to analyze a user's global preferences, and adopted RNN to learn the temporal changes in the preferences of users and items. They proposed four methods for dealing with the temporal dynamics of users' preferences for items. The best of the four uses an attention mechanism in LSTM to compute a weight for each hidden state by exploiting global factors. The rating prediction function is defined as Eqs. (2) and (3),

$$r_{ij,t} = g(e_i^u, e_j^m, h_{i,t-1}^u, h_{j,t-1}^m, c_{i,t}^u, c_{j,t}^m) = \frac{1}{1+\exp(-s)}, \quad (2)$$

$$s = e_i^u \cdot e_j^m + h_{i,t-q}^u \cdot h_{j,t-1}^m + b_i + b_j, \quad (3)$$

where $g(\cdot)$ is a score function, and its inputs e_i^u and e_j^m are the global latent factors of user i and item j; $c_{i,t}^u$ and $c_{j,t}^m$ are the context vectors at time step t for user i and item j; $h_{i,t}^u$ and $h_{j,t}^m$ represent the hidden states at time step t, which are computed by LSTM in Eqs. (4) and (5):

$$h_{i,t}^u = LSTM(h_{i,t-1}^u, z_{i,t}^u, c_{i,t}^u), \quad (4)$$

$$h_{j,t}^m = LSTM(h_{j,t-1}^m, z_{j,t}^m, c_{j,t}^m). \quad (5)$$

In the equations, $z_{i,t}^u \in \mathbb{R}^U$ and $z_{j,t}^m \in \mathbb{R}^M$ represent the rating vector of user i and item j given time t, respectively. U is the total number of users, and M is the total number of items. The context vectors $c_{i,t}^u$ and $c_{j,t}^m$ act as extra inputs of the hidden states of LSTMs to provide long-term information, as shown in Eq. (6),

$$c_{i,t}^u = \sum_{k=1}^{U} \alpha_{k,t}^i e_k^u; c_{j,t}^m = \sum_{p=1}^{M} \beta_{p,t}^j e_p^m \quad (6)$$

where $\alpha_{k,t}^i$ and $\beta_{p,t}^j$ are attention weights for user u and item j at time step t, computed by Eqs. (7) and (8),

$$\alpha_{k,t}^{i} = \frac{\exp(\sigma(h_{i,t-1}^{u}, e_{k}^{u}))}{\sum_{k'=1}^{U} \exp(\sigma(h_{i,t-1}^{u}, e_{k'}^{u}))}, \tag{7}$$

$$\beta_{p,t}^{j} = \frac{\exp(\sigma(h_{j,t-1}^{m}, e_{p}^{m}))}{\sum_{p,t}^{M} \exp(\sigma(h_{j,t-1}^{m}, e_{p'}^{m}))}. \tag{8}$$

The feed-forward neural network σ shown in the equations can produce a real-valued score. The attention weights together determine which user and item factors should be selected to generate $r_{ij,t}$.

Furthermore, Bharadhwaj *et al.* [2] proposed RecGAN, which improved IRGAN by combining it with RNN. RecGAN uses GRU to effectively extract a user's preference model related to time series, thereby improving the recommendation effect. Yu *et al.* [41] proposed the SeqGAN model, which considers the sequence generation procedure as a sequential decision-making process, providing the possibility of text generation. Chen *et al.* [7] added reinforcement learning (RL) to GAN, and proposed a GAN model to imitate the dynamic changes of users' behavior and learn their reward function in order to predict better user preference models. They also proposed a new cascading Q-network to deal with a large number of candidate items by combining recommendation strategies.

3. PROPOSED METHOD

3.1 Overview

This paper proposes a hybrid recommendation method based on GAN, which takes into accounts the user's preference changes in time series and the text characteristics of news articles. An overview of the proposed recommendation method is shown in Fig. 1. It can be divided into three parts: pre-processing, model training, and top-*N* recommendation. The input raw data includes a user's browsing history and the contents of news articles.

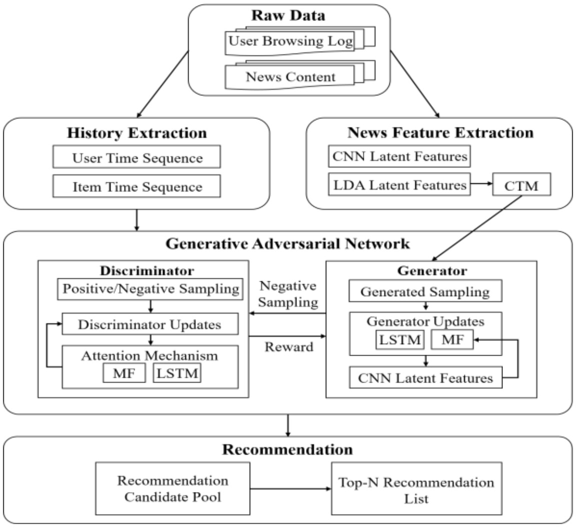

Fig. 1. Overview of the proposed recommendation method.

The proposed method first processes the raw data, including organizing the two into sequential form, so that there will be time information when running the LSTM. It also uses CNN, LDA and CTM to extract the latent features of news articles, which will be used in the subsequent GAN model.

Next, the GAN architecture is used to train the model, and LSTM and MF are used in both Generator and Discriminator. This study proposes an attention framework in the Discriminator, which effectively adjusts the weights of the two (LSTM and MF) according to their importance. In the Generator, a CNN-MF sub-model is added, so that the content of the article can be taken into accounts, and the embedding of users and items can be updated by MF. Once the Generator and the Discriminator have iteratively trained and learned, a top-*N* recommendation list is generated based on the prediction of the Generator.

3.2 Data Preprocessing and Extraction

This stage pre-processes the news articles, and extracts the news features. CNN is adopted to extract the latent features of the text. Moreover, the CTM model is adopted using LDA to derive the latent topic vector of each article, and MF is used to derive the latent factors of users and news articles.

This work considers the dynamic changes of time factors by recording the articles browsed by each user on a weekly basis. Similarly, users' browsing history for each week is recorded. The user/item time sequence can then be used by LSTM to push back the sequence of users and items within a month (four weeks), and analyze the patterns of users and items.

3.2.1 CTM latent factors

The latent factors of users and items derived from the CTM model are taken as the initial item and user vectors for the proposed GAN model. In the pre-processing step, the proposed method first extracts the latent topic vectors of news articles by LDA. The latent topic vectors of the articles are used as the initial latent factors of items in the MF. To optimize CTM latent factors, a rating matrix is first generated based on users' click count on each article. Next, the difference between the predicted ratings and the real ratings in the rating matrix are minimized through MF.

Since CTM uses LDA latent topic vectors as the initial latent factors of articles (items), and then uses MF to learn the latent factors for users and articles (items), it can then generate the latent factors of articles according to the latent topic vectors from LDA and latent factors from MF. Thus, cold start items can still be recommended to users according to the latent topic vectors derived from LDA.

3.2.2 CNN preprocessing

To convert news articles into trainable latent feature vectors, the proposed method enhances the Generator by employing the CNN-MF model to improve the learning of latent feature vectors of news articles. For the preprocessing of news articles, it applies the word2vec pretrained with 354,158 Chinese wiki documents to generate word embedding vectors representing each word. If words share similar context meanings, they will be located close to one another in the vector space, and then mapped to word embedding.

3.3 Generative Adversarial Networks

The proposed model enhances the PLASTIC GAN model [42] by integrating CNN-MF in the Generator to improve the learning of the latent features. Note that CNN is adopted to extract latent features of articles with text content. The proposed PLASTIC-CNN-MF-Attention framework integrates the advantages of existing models with the enhancement of CNN-MF for learning latent features of text content, and attention mechanism for combining the MF and LSTM in the Discriminator. MF is used to analyze users' global preferences, while LSTM is used to trace users' preferences on items over time. During the GAN iterative learning process, the Generator ϑ continuously selects samples from the candidate pool according to the predicted probability distribution of samples, while the Discriminator Φ distinguishes whether the samples are real or fake. Under the adversarial cycle, the model can gradually grasp the true distribution of the data. The overall objective function is shown as Eq. (9),

$$\begin{aligned}\vartheta^{*}, \Phi^{*} = \arg\min_{\vartheta}\max_{\Phi}\sum_{u}\sum_{d}(E_{r\sim D(r|u,d)_{real}}[\log Dis(r\,|\,u,d)])\\ + E_{r\sim D(r|u,d)_{\vartheta}}[\log(1-Dis(r\,|\,u,d))].\end{aligned} \tag{9}$$

where $Dis(r|u, d)$ estimates the Discriminator's prediction of the probability that news article d came from the true distribution. In Eq. (9), for the Generator, it is expected that $\log Dis(r|u, d)$ will be minimized if d is sampled from the true distribution, and $\log(1 - Dis(r|u, d))$ will be minimized, which is equal to maximizing $\log Dis(r|u, d)$, if d is sampled from the Generator. For the Discriminator, it is expected that $\log Dis(r|u, d)$ will be maximized if d is sampled from the true distribution, and $\log(1 - Dis(r|u, d))$ will be maximized, which is equal to minimizing $\log Dis(r|u, d)$, if d is sampled from the Generator.

3.3.1 Generator

Unlike traditional GAN, the Generator in the proposed model uses the latent factors derived from the CTM as the initial latent feature vectors of users and items. At each epoch, the Generator selects samples to update the latent features of users and items by minimizing the joint loss of MF and LSTM. The selection is based on the probability value of news article d's probability distribution $D(r|u, d)_{\vartheta}$, which is the conditional probability of relevance r given u and d. The larger the value, the higher the probability d will be selected as important examples for the Generator, with the reward obtained from the Discriminator. Moreover, if news article d is in the training set, its probability value is also increased. Then, *softmax* is performed for all articles, yielding the probability distribution of all articles, as shown in Eq. (10),

$$D(r\,|\,u,d)_{\vartheta} = \frac{\exp(D(r\,|\,u,d)_{\vartheta})}{\sum_{1}^{k}\exp(D(r\,|\,u,d_k)_{\vartheta})}. \tag{10}$$

The higher the $D(r|u, d)_{\vartheta}$ value of news article d, the more likely d is to be selected. In addition, if news article d is in the training set, $D(r|u, d)_{\vartheta}$ is increased, giving news article d a higher chance to be selected as an important example for the Generator.

This study uses reinforcement learning policy gradient to optimize the Generator model, which can solve the problem of scattered news article items and the inability to use gradient descent directly. We adopt the approach [41] to pass the reward from the Discriminator to the Generator. The generative model's loss function is optimized according to $D(r|u, d)_\vartheta$ and the reward from the Discriminator. The reward of the Discriminator is defined as Eq. (11),

$$R = 2 \cdot D(r|u, d) - 1. \tag{11}$$

The higher the $D(r|u, d)_\vartheta$ value of news article d, the more likely d is to be selected. Similarly, the higher the reward from the Discriminator of news article d, the higher the chance news article d has to be selected. By contrast, the generative model aims to minimize the objective function, however, it is worth mentioning that while minimizing the objective of the generative model, this study also optimizes the parameters of CNN-MF model. The objective function of the Generator is shown as Eq. (12),

$$\vartheta^* = \arg\min_{\vartheta} \sum_u \sum_d (E_{r\sim D(r|u,d)_\vartheta}[\log(1-Dis(r\,|\,u,d))]). \tag{12}$$

For the Generator, if article d is selected from the Generator, it is expected that log-$Dis(r|u, d)$ will be maximized, which is equivalent to minimizing $\log(1 - Dis(r|u, d))$.

In addition, this study refers to PLASTIC [42], and adds LSTM and MF to the Generator, so that the Generator can simultaneously use LSTM to learn users' dynamic preferences, and MF to learn users' long-term global preferences. First, the samples are preprocessed into data sequences: the articles that the user has seen each week, and users who have viewed the article each week. Then the LSTM is performed on the data sequences to predict the LSTM-based preferences. Meanwhile, the samples are also used to obtain the corresponding long-term global preferences by using MF. Finally, the overall joint logits combing LSTM and MF predicted preferences are shown in Eq. (13),

$$joint\ \mathrm{log}its = \mathrm{log}its_{MF} + \mathrm{log}its_{LSTM}. \tag{13}$$

In the Generator model, both LSTM and MF are considered. During learning, MF and LSTM are trained together, and then the user and item latent factors derived by the MF of the Generator are fed into the CNN-MF model to further update the latent factors.

3.3.2 CNN-MF in the generator

This study refers to ConvMF's approach [20], and uses CNN's four-layer architecture to convolve news article content into latent features. These latent features are extracted from the text information of news articles through shared weights and sliding windows. In the convolutional layer, the text feature c_d^j of news article d is obtained by Eqs. (14) and (15),

$$c_d^j = f(W_c^j * D_{(:,i:(i+ws-1))} + b_c^j), \tag{14}$$

$$c^j = [c_1^j, c_2^j, \ldots, c_{i-ws+1}^j]. \tag{15}$$

In the formula, W_c^j determines the jth shared weight, the sliding window ws represents the number of words related to the sample word, and b_c^j is the bias of W_c^j. Then, the pooling layer extracts representative features from the output of the convolutional layer, and constructs vector features of fixed length. The pooling layer uses maximum pooling to obtain the maximum of text features from each text feature vector. This extracted latent feature of d, denoted by cnn_d, can be expressed by Eq. (16),

$$cnn_d = [\max(c^1), \max(c^2), \max(c^j), \ldots, \max(c^{n_c})]. \tag{16}$$

In this study, each news article d is defined as an input file X_d, in which the weight of CNN is W, and the latent feature vector of each article is learned through the CNN convolution method by Eq. (17),

$$cnnlf_d = CNN(W, X_d). \tag{17}$$

CNN-MF iteratively optimizes the objective function to update the model parameters, which is shown in Eq. (18),

$$\begin{aligned} L(U,D,W) = {} & \sum_u \sum_d \frac{c_{ud}}{2}(p_{ud} - x_u^T y_d)^2 + \frac{\lambda_U}{2}\sum_u \| x_u \|^2 \\ & + \frac{\lambda_D}{2}\sum_d \| y_d - CNN(W, X_d) \|^2 + \frac{\lambda_W}{2}\sum_k^{|w_k|} \| w_k \|^2, \end{aligned} \tag{18}$$

where p_{ud} is a binary variable, and if the user u has read the document d, it is set to 1, and if it is not, it is set to 0; c_{ud} stands for observing the confidence level of p_{ud}; x_u stands for the user latent factor, y_d denotes the document latent factor.

3.3.3 Discriminator

This study defines news articles that the user has actually seen as positive samples, labeled 1, and the items selected according to probability distribution $D(r|u, d)_\vartheta$ calculated by the Generator are defined as negative samples, labeled 0. The greater the value of $D(r|u, d)_\vartheta$, the higher its probability of being selected as a negative sample. The training process of the Discriminator uses Sigmoid Cross Entropy as the loss function, and the objective function is defined as Eq. (12):

$$\begin{aligned} \Phi^* = {} & \arg\min_\Phi \sum_u \sum_d (E_{r\sim D(r|u,d)_{real}}[\log Dis(r \mid u, d)]) + E_{r\sim D(r|u,d)_{\vartheta^*}} \\ & [\log(1 - Dis(r \mid u, d))]. \end{aligned} \tag{12}$$

In the objective equation, if the item d is a training article that the user has actually seen, it is expected that $\log D(r|u, d)_\vartheta$ will be maximized; if the item d is selected from the Generator, it is expected that $\log(1 - Dis(r|u, d))$ will be maximized.

3.3.4 Attention mechanism in the discriminator

In the Discriminator, this study also considers LSTM and MF jointly, and proposes an attention mechanism to effectively combine the two. Although PLASTIC uses an attention mechanism to compute a weight for each hidden state in the LSTM by exploiting the global factors, it merely combines the LSTM and MF results, *i.e.*, the predicted logits of LSTM and MF, by adding them together.

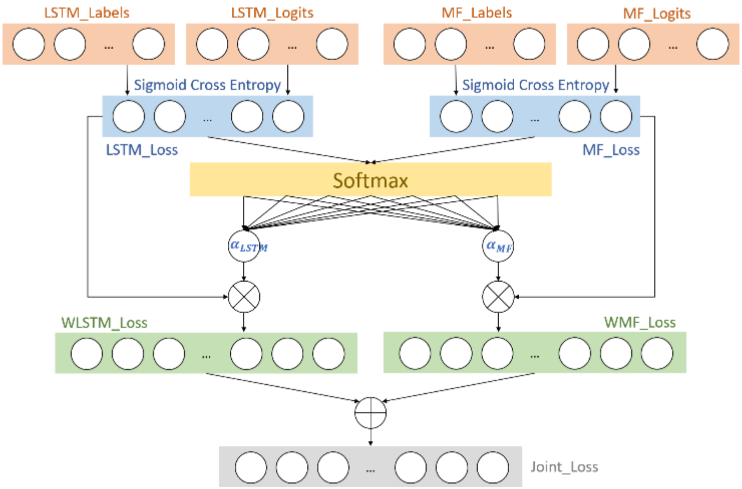

Fig. 2. Attention-based LSTM-MF loss method in discriminator.

The proposed attention structure is shown in Fig. 2 and as Eqs. (20)-(24). The weekly data obtained after the pre-processing are run down through LSTM and MF, respectively. After calculating *LSTM_Logits*/*MF_Logits*, the Sigmoid Cross Entropy is applied with the corresponding labels to get the *LSTM_Loss*/*MF_Loss*. The two are then combined through a dense layer with a two-dimensional output by softmax to derive the attention weight $\alpha_{LSTM}/\alpha_{MF}$.

$$\begin{aligned}\alpha_{LSTM} &= soft\max(LSTM_{Loss}) \\ &= \frac{\exp(LSTM\_Loss \cdot \omega_{hLSTM} + b_{hLSTM})}{\exp(LSTM\_Loss \cdot \omega_{hLSTM} + b_{hLSTM}) + \exp(MF\_Loss \cdot \omega_{hMF} + b_{hMF})}\end{aligned} \tag{20}$$

$$\begin{aligned}\alpha_{MF} &= soft\max(MF\_Loss) \\ &= \frac{\exp(MF\_Loss \cdot \omega_{hMF} + b_{hMF})}{\exp(LSTM\_Loss \cdot \omega_{hLSTM} + b_{hLSTM}) + \exp(MF\_Loss \cdot \omega_{hMF} + b_{hMF})}\end{aligned} \tag{21}$$

where $\omega_{hLSTM}/\omega_{hMF}$ denotes the weight, and b_{hLSTM}/b_{hMF} denotes bias variables of the softmax activation function. Moreover, the softmax method is applied, which results in the sum of α_{LSTM} and α_{MF} always being 1, and both denote the importance of LSTM latent features and MF latent features, respectively.

Accordingly, the weight adjustment can be obtained through Eqs. (22) and (23) by multiplying the original loss *LSTM_Loss*/*MF_Loss* with the attention weights $\alpha_{LSTM}/\alpha_{MF}$. to generate the attentional LSTM loss (*WLSTM_Loss*) and attentional MF loss (*WMF_Loss*). The sum of the two is the integrated loss value (joint loss) of the objective function, which is shown in Eq. (24).

$$WLSTM\_Loss = \alpha_{LSTM} * LSTM\_Loss \tag{22}$$

$$WMF\_Loss = \alpha_{MF} * MF\_Loss \tag{23}$$

$$Joint\_Loss = WLSTM\_Loss + WMF\_Loss \tag{24}$$

The Discriminator is thus enhanced through the attention mechanism, in which LSTM and MF can be effectively fused together, and the Discriminator model can be updated through the integrated loss value adjusted by the attention mechanism.

3.4 Top-*N* Recommendation

When the Generator and the Discriminator iteratively optimize the model to achieve convergence, the Discriminator is no longer able to discriminate whether the sample is real or fake. The probability model of the Generator $D(r|u, d)_{\vartheta}$ approaches the true probability distribution $D(r|u, d)_{real}$, and the model reaches equilibriums. During recommendation, the proposed model selects top-N articles with the highest probability according to the probability distribution $D(r|u, d)_{\vartheta}$ derived from the Generator to recommend to the user. The higher the $D(r|u, d)_{\vartheta}$ value of news article d for user u, the higher news article d's chance to be selected for user u, and thus the top-N highest articles are recommended for each user.

Algorithm 1: The Proposed Recommendation Method

Input: Generator ϑ, Discriminator Φ, training data S.
Initialize model Φ with random weights, model ϑ with CTM.
Pre-train ϑ, Φ using S
repeat
for g-steps **do**
  Generator ϑ selects important samples for each user u
  Get each sample's time and append them into user/item sequence forms
  Apply LSTM and MF to user/item sequences and compute joint logits by Eq. (13)
  Update Generator ϑ by Eq. (12) and Discriminator's reward by Eq. (11)
  Use current Generator ϑ's user/document latent vectors to initialize CNN-MF model
  for CNN-MF steps **do**
    Optimize and update CNN latent feature by Eq. (17) based on document latent vectors
    Optimize and update MF user preference model by Eq. (18)
  end for
    Use user/document latent vectors of CNN-MF model as the Generator ϑ's user/document latent vectors
end for
for d-steps **do**
  Use current Generator ϑ to select negative examples and combine with given positive examples S
  Get each sample's time and append them into user/item sequence forms
    Apply LSTM and MF to user/item sequences and compute joint loss by Eq. (24)

```
            using the proposed attention mechanism
        Update and optimize Discriminator Φ by Eq. (19)
    end for
        until convergence
        Recommend top-N articles for each user u according to Generator ϑ
```

Fig. 3. Algorithm of the proposed method.

The overall logic of the proposed model is summarized in Fig. 3. Before the adversarial training, the Discriminator can be initialized by its conventional models, and the Generator uses the pre-processed CTM vectors as the initial latent feature vectors of users and items. This study integrates CNN-MF into the Generator in order to learn the latent features of text content, and uses the attention mechanism in the Discriminator to combine MF and LSTM. Then, during the adversarial training stage, the Generator and Discriminator are trained alternatively via Eqs. (12) and (19).

4. EXPERIMENT AND EVALUATION

4.1 Experimental Setting

This research utilized the NiusNews (https://www.niusnews.com/) dataset, an online news site, to evaluate the proposed method. Niusnews media platform provides multi-lateral services, including news, makeup tips, e-commerce, and online activities. There are about 50 to 60 daily news posts for Niusnews. The experiments use the Nius-News February and March, 2019 dataset since March contains the largest amount of data for that year. The data for each user and item was backtracked over four weeks. Thus, the March data was mainly used to split training data and test data, and the February and March data was used for backtracking. There are 2,629 users, 2,368 news articles and 29,209 browsing records in the dataset.

In the experiments, five baseline approaches, MF [23], CTM [35], CNN-MF [21], IRGAN [36] and PLASTIC [42] were compared with the four proposed approaches, IRGAN-MF, PLASTIC-MF, PLASTIC-CNN-MF and PLASTIC-CNN-MF-Att.

- **MF:** Matrix Factorization is designed by rating prediction matrix, which is multiplied by user and item vectors. Both of the initial vectors of MF are randomly generated.
- **CTM:** Collaborative Topic Modeling is a state-of-the-art recommendation model, which uses LDA as the initial latent factors of news articles for MF.
- **CNN-MF:** Recommendation is designed by Convolutional Matrix Factorization (Conv-MF) approach, which integrates CNN into MF.
- **IRGAN:** Recommendation is designed by IRGAN approach using the GAN framework for top-*N* item recommendation for each user. The approach is described as Eq. (1) in Section 2.5.
- **PLASTIC:** Prioritize long and short-term information in top-n recommendation is designed by combining MF and LSTM through an adversarial training framework.
- **IRGAN-MF:** Recommendation is designed by using MF to update user and item vectors in the Generator of IRGAN after the Generator is updated.

- **PLASTIC-MF:** Recommendation is designed by using MF to update user and item vector in the Generator of PLASTIC after the Generator is updated.
- **PLASTIC-CNN-MF:** Recommendation is designed by using CNN to extract items' latent feature vectors, and then using MF to update user and item vectors in the Generator of PLASTIC after the Generator is updated.
- **PLASTIC-CNN-MF-Att:** Recommendation is designed by using CNN to extract items' latent feature vectors, and then using MF to update user and item vectors in the Generator of PLASTIC after the Generator is updated. There is also an attention mechanism in the Discriminator to combine LSTM and MF.

Note that the four proposed approaches, IRGAN-MF, PLASTIC-MF, PLASTIC-CNN-MF and PLASTIC-CNN-MF-Att use CTM as their initial input vectors. The proposed method was implemented using the Python, Tensorflow 2.0 library with an NVIDIA 1080Ti graphics card. Various parameter settings were tested for model training. After numerous tests, this study adopted 1) number of topics: 80 for LDA and CTM. 2) CNN parameters: word latent vectors with dimension size of 200, window size (3, 4, 5), and dropout rate of 0.2 to prevent CNN from over-fitting. 3) learning rate: 0.001 for both Generator and Discriminator in the IRGAN series model; 0.000015 for the Generator, 0.000001 for the Discriminator in the PLASTIC series model.

To test the performance of each experiment on the real world dataset, this study used the precision, recall, F1 scores and Normalized Discounted Cumulative Gain (NDCG) for the various models to evaluate their recommendation results. F1 score can be used to observe the balance between Precision and Recall.

4.2 Model Evaluation

This section examines the effect of different combinations of MF, CNN, attention mechanism, and different initial input vectors separately. This study determines whether using MF to update the user and item vectors can improve performance during the training process of the Generator. In addition, CNN is added after MF in each Generator epoch to examine the improvement of extracting latent vectors. Finally, a novel attention mechanism is applied to combine LSTM and MF jointly in the Discriminator. The results demonstrate that the proposed approach, PLASTIC-CNN-MF-Attention, can outperform the other compared methods.

4.2.1 Evaluation of MF and CNN updating

This section compares the effect of using MF to update the Generator's user and item vectors during each Generator iteration. Unlike PLASTIC, which only uses MF to observe global records, the proposed model further uses MF to update the user/item latent factors derived by the Generator in each iteration. This allows the model to generate a more accurate latent model. The experiment result of precision, recall and F1 scores for top-10 recommendation is shown in Table 1. This study modifies two baseline GAN models, IRGAN and PLASTIC, to examine the results of IRGAN(CTM) and PLASTIC(CTM) with the initial input vectors produced by CTM. Note that the original baseline IRGAN(MF) and PLASTIC(MF) models only use MF as their initial input vectors. The result shows that IRGAN(CTM) and PLASTIC(CTM) can get better performance than the baseline models

IRGAN(MF) and PLASTIC(MF). The models IRGAN-MF and PLASTIC-MF further enhance IRGAN (CTM) and PLASTIC (CTM) by using MF to update the user/item latent factors derived by the Generator in each iteration. The result shows that PLASTIC performs better than IRGAN, and each of them combining with MF update – IRGAN-MF and PLASTIC-MF can get better performance than the models IRGAN (CTM) and PLASTIC (CTM) without MF update.

This section also examines whether CNN can improve recommendation results by including it in the Generator before using MF to update. This study applied Google's Word2Vec as pre-trained model, and then utilized the CNN with PLASTIC-MF to design PLASTIC-CNN-MF. Table 1 demonstrates that PLASTIC-CNN-MF outperforms other competitors, where all the PLASTIC methods adopt CTM as initial input vectors. The result shows that PLASTIC-CNN-MF's performance was superior to all the compared methods. Thus, it can be concluded that using CNN to extract contextual latent features can help enhance recommendation results.

Table 1. Evaluation of MF and CNN updating for top-10 recommendation.

	Precision	Recall	F1-score
IRGAN(MF)	0.101767	0.122000	0.110969
PLASTIC(MF)	0.102473	0.121727	0.111273
IRGAN(CTM)	0.109187	0.130702	0.118980
PLASTIC(CTM)	0.110954	0.133551	0.121208
IRGAN-MF	0.116608	0.138923	0.126791
PLASTIC-MF	0.116961	0.139276	0.127147
PLASTIC-CNN-MF	0.118375	0.140081	0.128316

4.2.2 Evaluation of attention mechanism

PLASTIC simply combines LSTM and MF by adding their result logits. However, sometimes the MF value may be comparatively larger than that of LSTM, resulting in the sum being influenced by MF. The proposed method, however, applies an attention mechanism in the Discriminator which enables the model to learn the importance of LSTM and MF through a neural network, and therefore adjust their weights. Table 2 shows that the proposed method, PLASTIC-CNN-MF-Att, performs better than the model PLASTIC-CNN-MF without attention. This implies that the proposed attention-based model can prevent the sum of LSTM and MF being influenced by only MF or LSTM, and thus provide more effective recommendations.

Table 2. Evaluation of attention mechanism for top-10 recommendation.

	Precision	Recall	F1-score
PLASTIC-CNN-MF	0.118375	0.140081	0.128316
PLASTIC-CNN-MF-Att	0.120848	0.143734	0.131301

4.3 Experiment Results

The proposed approach was evaluated using a dataset taken from an online news media platform, NiusNews (https://www.niusnews.com/). The overall performances (F1 and

NDCG) of the compared approaches for top-10 and top-15 recommendations are shown in Figs. 4 and 5, respectively. From the experiment results, PLASTIC-CNN-MF-Att achieves better improvements than all other compared approaches. Other conclusions can also be drawn from these results.

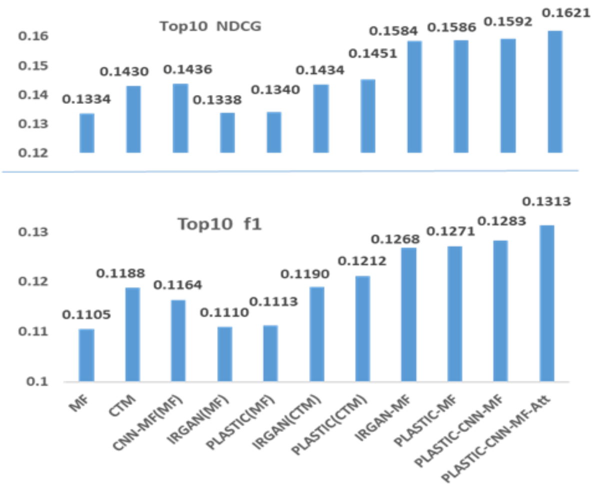

Fig. 4. Comparison of F1-scores and NDCG of various methods in Top-10 recommendation.

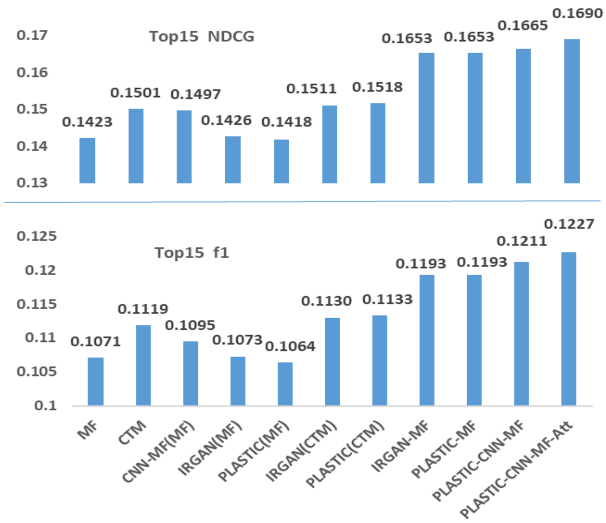

Fig. 5. Comparison of F1-scores and NDCG of various methods in Top-15 recommendation.

First, the four proposed approaches, IRGAN-MF, PLASTIC-MF, PLASTIC-CNN-MF and PLASTIC-CNN-MF-Att perform better than conventional methods like MF, CTM or CNN-MF. By virtue of the GAN adversarial structure, both Generator and Discriminator models allow more flexibility on training models. Next, PLASTIC with LSTM can generally explore the dynamic changes of users' preference more precisely over time series than can IRGAN.

There is a performance gap between MF and CTM when selecting GAN's initial input vector, since CTM has the ability to extract the latent features of news articles. The approaches use CTM as their initial input vectors, including IRGAN(CTM), PLASTIC (CTM), IRGAN-MF, PLASTIC-MF, PLASTIC-CNN-MF and PLASTIC-CNN-MF-Att, perform better than IRGAN(MF) and PLASTIC(MF) which only use MF as their input vectors.

IRGAN-MF, PLASTIC-MF, PLASTIC-CNN-MF and PLASTIC-CNN-MF-Att perform better than IRGAN(CTM) and PLASTIC(CTM). The models using MF to update the Generator's user and item vectors helps in learning latent user and item models. Moreover, PLASTIC-CNN-MF performs better than PLASTIC-MF. Incorporating CNN into the PLASTIC-MF model can capture documents' contextual information, which enhances the proposed model by extracting latent vectors from news content. Furthermore, the PLASTIC architecture was improved in this study using a novel attention mechanism to effectively combine MF and LSTM. All of these factors were compared and verified gradually, and the proposed PLASTIC-CNN-MF-Att approach achieves the best performance among all the compared methods.

5. DISCUSSIONS AND CONCLUSIONS

This study proposes a novel hybrid GAN-based news recommendation model. The proposed method applies CTM to extract initial user and item latent factors. The Generator then applies CNN-MF to these latent factors to improve content extraction. In addition, in order to improve recommendation results based on a user's dynamic preferences over time, this study considers time an important factor as well. LSTM and MF are considered in the proposed GAN model simultaneously: LSTM can track the dynamic behaviors and trends of users and items, while MF can observe their global records. Moreover, inspired by PLASTIC's limitation, an innovative attention mechanism structure is employed, whereby the model can learn and adjust the MF and LSTM weights in order to more effectively combine the two.

Several experiments were conducted to evaluate the performance of the proposed model. The experiment results show that the proposed PLASTIC-CNN-MF-Att model can improve recommendation results and achieve better performance than the MF, CTM, CNN-MF, IRGAN and PLASTIC baselines. The result implies that PLASTIC, considering time series, can perform better than IRGAN. PLASTIC-CNN-MF can further enhance contextual extraction of news articles and latent factor learning over PLASTIC. And finally, PLASTIC-CNN-MF-Att modifies the Discriminator with an innovative attention mechanism to combine the LSTM and MF. Accordingly, the proposed PLASTIC-CNN-MF-Att model achieves the best performance among all compared methods. The model has great potential in other practical applications, which can increase the effectiveness of recommending textual content.

Planned future work will focus on the following. The NiusNews website contains other activities such as shopping, video channels, and event bookings. Further work will integrate various activities, and recommend similar items and articles to users. Users may have similar preferences in different activities, and a complete user experience on the NiusNews website can improve customers' loyalty. Almost every recommendation system encounters problems with new item recommendation. Since new items lack rating information, recommendation models with simple approaches perform poorly in this area. Future work will focus on new items, and improve recommendation results by applying other deep learning methods.

REFERENCES

1. D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv Preprint*, 2014, arXiv:1409.0473.
2. H. Bharadhwaj, H. Park, and B. Y. Lim, "Recgan: recurrent generative adversarial networks for recommendation systems," in *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018, pp. 372-376.
3. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, Vol. 3, 2003, pp. 993-1022.
4. J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-based Systems*, Vol. 46, 2013, pp. 109-132.
5. T.-M. Chang, W.-F. Hsiao, and W.-L. Chang, "An ordinal regression model with SVD Hebbian learning for collaborative recommendation," *Journal of Information Science and Engineering*, Vol. 30, 2014, pp. 387-401.
6. H.-Y. Chen, Y.-C. Lin, D.-R. Liu, and T.-F. Liu, "Online product recommendations based on diversity and latent association analysis on news and products," *Journal of Information Science and Engineering*, Vol. 38, 2022, pp. 1065-1085.
7. X. Chen, S. Li, H. Li, S. Jiang, Y. Qi, and L. Song, "Generative adversarial user model for reinforcement learning based recommendation system," *arXiv Preprint*, 2018, arXiv: 1812.10613.
8. Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789-8797.
9. J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *Advances in Neural Information Processing Systems*, 2015, pp. 577-585.
10. Y.-C. Chou, H.-Y. Chen, D.-R. Liu, and D.-S. Chang, "Rating prediction based on merge-CNN and concise attention review mining," *IEEE Access*, Vol. 8, 2020, pp. 190934-190945.
11. P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 191-198.
12. J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, and B. Livingston, "The YouTube video recommendation system," in *Proceedings of the 4th ACM Conference on Recommender Systems*, 2010, pp. 293-296.
13. E. L. Denton, S. Chintala, and R. Fergus, "Deep generative image models using a lapacian pyramid of adversarial networks," *Advances in Neural Information Processing*

Systems, 2015, pp. 1486-1494.
14. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, 2014, pp. 2672-2680.
15. X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 173-182.
16. C. Healey and J. Enns, "Attention and visual memory in visualization and computer graphics," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, 2011, pp. 1170-1188.
17. S. Hochreiter and J. J. N. c. Schmidhuber, "Long short-term memory," *Neural Computation*, Vol. 9, 1997, pp. 1735-1780.
18. L. Hu, C. Li, C. Shi, C. Yang, and C. Shao, "Graph neural news recommendation with long-term and short-term interest modeling," *Information Processing & Management*, Vol. 57, 2020, p. 102142.
19. L. Itti and C. Koch, "Computational modelling of visual attention," *Nature Reviews Neuroscience*, Vol. 2, 2001, pp. 194-203.
20. K.-Y. Jung and J.-H. Lee, "User preference mining through hybrid collaborative filtering and content-based filtering in recommendation system," *IEICE Transactions on Information and Systems*, Vol. 87, 2004, pp. 2781-2790.
21. D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional matrix factorization for document context-aware recommendation," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 233-240.
22. Y. S. Kim, "Recommender system based on product taxonomy in e-commerce sites," *Journal of Information Science and Engineering*, Vol. 29, 2013, pp. 63-78.
23. Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, Vol. 42, 2009, pp. 30-37.
24. P. D. Lakshmi, K. Ramani, and B. E. Reddy, "Neighborhood algorithm for product recommendation," in *Computational Intelligence and Big Data Analytics*, Springer, 2019, pp. 37-53.
25. D.-R. Liu, Y.-S. Liao, and J.-Y. Lu, "Online news recommendations based on topic modeling and online interest adjustment," *Industrial Management & Data Systems*, Vol. 119, 2019, pp. 1802-1818.
26. Q.-M. Liu, R.-S. Jia, Y.-B. Liu, H.-B. Sun, J.-Z. Yu, and H.-M. Sun, "Infrared image super-resolution reconstruction by using generative adversarial network with an attention mechanism," *Applied Intelligence*, 2020, pp. 1-13.
27. T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proceedings of the 11th Annual Conference of International Speech Communication Association*, 2010, pp. 1045-1048.
28. M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*, Springer, 2007, pp. 325-341.
29. J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*, Springer, 2007, pp. 291-324.
30. L. Shi, J.-P. Du, M.-Y. Liang, and F.-F. Kou, "SRTM: A sparse RNN-topic model for discovering bursty topics in big data of social networks," *Journal of Information Science and Engineering*, Vol. 35, 2019, pp. 749-767.

31. X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, Vol. 2009, 2009, pp. 1-19.
32. P. B. Thorat, R. Goudar, and S. Barve, "Survey on collaborative filtering, content-based filtering and hybrid recommendation system," *International Journal of Computer Applications*, Vol. 110, 2015, pp. 31-36.
33. A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," *Advances in Neural Information Processing Systems*, 2013, pp. 2643-2651.
34. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017, pp. 5998-6008.
35. C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011, pp. 448-456.
36. J. Wang, L. Yu, W. Zhang, Y. Gong, Y. Xu, B. Wang, P. Zhang, and D. Zhang, "Irgan: A minimax game for unifying generative and discriminative information retrieval models," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 515-524.
37. C. Wu, F. Wu, M. An, Y. Huang, and X. Xie, "Neural news recommendation with topic-aware news representation," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1154-1159.
38. K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of International Conference on Machine Learning*, 2015, pp. 2048-2057.
39. T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1316-1324.
40. Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4651-4659.
41. L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 2017, pp. 2852-2858
42. W. Zhao, B. Wang, J. Ye, Y. Gao, M. Yang, and X. Chen, "PLASTIC: Prioritize long and short-term information in top-n recommendation using adversarial training," in *Proceedings of International Joint Conferences on Artificial Intelligence*, 2018, pp. 3676-3682.
43. H. Zhu and T. Kaneko, "Residual network for deep reinforcement learning with attention mechanism," *Journal of Information Science and Engineering*, Vol. 37, 2021, pp. 517-533.
44. L. Zhu, L. Yu, Z. Cai, and J. Zhang, "Toward pattern and preference-aware travel route recommendation over location-based social networks," *Journal of Information Science and Engineering*, Vol. 35, 2019, pp. 959-975.

Duen-Ren Liu is a Professor of Institute of Information Management at the National Yang Ming Chiao Tung University of Taiwan. He received the BS and MS degrees in Computer Science from the National Taiwan University, Taiwan, in 1985 and 1987, respectively. He received the Ph.D. degree in Computer Science from the University of Minnesota, USA, in 1995. His research interests include data mining, knowledge engineering, e-commerce and recommender systems.

Yang Huang received the BS degree in Power Mechanical Engineering from National Tsing Hua University, Taiwan, in 2016. He received the master's degree in Institute of Management of Technology from the National Yang Ming Chiao Tung University in 2018. He is currently pursuing the Ph.D. degree with the Institute of Information Management, National Yang Ming Chiao Tung University, Taiwan. His research interests include machine learning, recommender systems, and data mining.

Ching-Yi Tseng received the BS degree in Department of Information Management and Finance from the National Yang Ming Chiao Tung University, Taiwan. She received the MS degree in Institute of Information Management from the National Yang Ming Chiao Tung University, Taiwan. Her research interests include recommender systems, machine learning, and deep learning.

Shin-Jye Lee received the M.Sc. (Eng.) degree from the Department of Computer Science, The University of Sheffield, U.K., in 2001, the M.Phil. degree from the Judge Business School, University of Cambridge, U.K., in 2012, and the Ph.D. degree from the School of Computer Science, The University of Manchester, U.K., in 2011. He was with the National Pilot School of Software, Yunnan University, China, and he also made his academic career in Poland, in 2012. In addition, he also had practical experiences at Fujitsu and Microsoft, from 2002 to 2005. He is currently an Associate Professor with the Institute of Management of Technology, National Yang Ming Chiao Tung University, Taiwan. His research interests primarily comprise machine learning, computational intelligence and decision support system, and technology policy.