IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 23 June 2023, accepted 6 August 2023, date of publication 9 August 2023, date of current version 15 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3303479

RESEARCH ARTICLE

AI-News Personalization System Combining Complete Content Characterization and Full Term Interest Portrayal in the Big Data Era

WENWEN FU
College of Media and Communications, Weifang University, Weifang, Shandong 261061, China
e-mail: weisixun27745@163.com

ABSTRACT In order to leverage the advantages of the large sample capacity in the era of big data and improve the performance of contemporary news recommendation methods, we have characterized complete information such as news textual details, explicit themes, and implicit themes, and thoroughly portrayed users' mixed full term interest including long-term and short-term interests. As a result, we propose an artificial intelligence (AI)-News Personalization system that combines Complete news Content Characterization and user Full term interest portrayal, i.e., NP-3C-FIP. The NP-3C-FIP system first utilizes Latent Dirichlet Allocation (LDA) to extract the implicit theme distribution from the news textual content. Then, it learns the unified news characterizations based on the headline, summary, category, subcategory, and implicit themes. Using these news characterizations, the proposed method transforms the historical clicked news into the representation vectors. Subsequently, the obtained sequence of news representation vectors is fed into a Gate Recurrent Unit (GRU) network to capture the sequential interest features of the user. Furthermore, this paper introduces a personalized attention mechanism to model the stable tastes. Finally, the system concatenates the portrayals of the full term to obtain a unified user representation vector, and calculates the click probability for candidate articles using vector dot product. Experimental results demonstrate the effectiveness of the proposed method in improving news recommendation performance.

INDEX TERMS AI, news personalization system, content characterization, full term interest portrayal, big data.

I. INTRODUCTION

In the era of information explosion, both online news platforms and readers face tremendous challenges. For readers, it is difficult to find interested content from a large amount of news information. For news platforms, it is not easy to make their produced content stand out and attract a wide range of readers. News recommendation systems exist to address this contradiction, with the key being how to estimate the likelihood of users adopting a certain news based on their historical interaction behavior. It is of great significance for alleviating information overload and facilitating a win-win situation for platforms and users [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina.

News recommendation is an application of recommendation technology in the field of personalized news reading. Among all recommendation techniques, Collaborative Filtering (CF) is one of the earliest and fundamental techniques [2], [3]. Traditional CF algorithms typically calculate the similarity of user behavior using metrics such as cosine similarity and Jaccard coefficient [4], [5], [6]. With the development of deep learning, collaborative recommendation has recently shifted towards learning implicit space representations of users and items [7], [8], [9]. When applied to the field of news recommendation, how to learn representations of news and users' vectors has become a key problem to solve [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22]. According to the [23], at the model representation level, CF has gone through the stages of modeling individual users or items (i.e., using their own IDs to represent users or items) [24], modeling their own interaction history (i.e., representing users using items they have directly interacted with) [7], and modeling the entire user-item interaction graph (i.e., representing users or items using higher-order interaction relationships) [8], [9]. This idea of transitioning from self to interaction history and interaction graph is worth learning from. However, the performance of pure CF algorithms is unsatisfactory when directly applied to news recommendation [10]. A news article contains rich textual information such as headline, summary, body, as well as additional information like theme. How to utilize advanced Natural Language Processing (NLP) techniques to learn high-quality text representations and how to integrate these heterogeneous news information to obtain richer semantic news representations are key to improving recommendation quality [25].

Currently, most neural network-based news recommendation methods adopt a two-stage approach: extracting news features first and then exploring personalized interests based on the user's historical interactions with news [26]. For example, the Neural News Recommendation with Attentive Multi-View Learning (NAML) model [11] first learns unified news representation vectors based on headlines, bodies, and categories, and then integrates all the news clicked by user using an attention mechanism to obtain a unified user representation vector. From the perspective of user interest modeling, this paper summarizes three categories of methods from the surveyed literature. The first category focuses on overall interests [10], [11], [12], [14], [15], [16], [17], [18], [19], [20] generally integrating all the news interacted by the user using attention aggregation, such as NAML. The second category focuses on sequential interests [27], [28], [29], using recurrent neural networks (RNNs) to model the user's evolving dynamic interests over time. For example, literature [28] uses Long Short-Term Memory (LSTM) to capture the user's sequential interests, also known as short-term interests. The third category focuses on modeling mixed interests [13], [15], [16], [21], [22], which combines overall and sequential interests. For example, the Neural News Recommendation with Long- and Short-Term User Representation (LSTUR) model [13] uses the Gate Recurrent Unit (GRU) to capture the user's short-term interests from the clicked news and represents the user's long-term interests using their user ID.

Compared to CF algorithms, many news recommendation works combine news content with user interaction history using a two-stage hybrid recommendation method [27], [28], [29], [30]. Existing methods mostly rely on a single information source, such as the headline, which may not achieve satisfactory results [11]. The NAML model verifies the effectiveness of combining different types of news information, but it only uses simple explicit themes when considering theme information. An article may belong to multiple themes, and a simple category label, such as sports, may not adequately express the news [27]. In terms of user representation, existing methods usually consider overall, sequential, or mixed interests, but few methods explore mixed user interests on the basis of integrating different types of news information. Relying solely on sequential behavior may not be accurate enough when users accidentally click on incorrect news or are attracted by unrelated news out of curiosity [31]. Therefore, this paper explores users' mixed interests based on fully utilizing the news textual content and additional information, proposing a News Personalization system combining Complete Content Characterization and Full term Interest Portrayal (NP-3C-FIP). The core of the model is a Complete Content Characterization Module (3CM) and a Full term user Interest Portrayal Module (FIPM). The proposed 3CM first uses Latent Dirichlet Allocation (LDA) to extract the implicit theme distribution from the news body [32], and then learns a unified news characterization based on the headline, summary, category, subcategory, and implicit themes. The FIPM emphasizes the exploration of mixed interests. Considering the evolving nature of user interests over time, this paper first uses GRU to model the user's sequential interest features from the historical interaction sequence, using the last time step's interest feature vector as the user's short-term interest portrayal. Then, an personalized attention mechanism is introduced to adaptively model frequent behaviors in the interaction sequence, obtaining the user's long-term interest portrayal. The model predicts and recommends by taking the dot product of the user portrayal vector and the candidate news characterization vector.

In summary, the main contributions of this paper are as follows:

(1) We propose a News Personalization system combining Complete Content Characterization and Full term Interest Portrayal (NP-3C-FIP), which explores the implicit theme information of news on one hand and comprehensively considers users' full term interest, i.e., long and short-term interests, on the other hand.

(2) We emphasize the fusion of textual content such as headlines and summaries, as well as additional information such as implicit and explicit themes, to obtain more semantically rich news representations, and explore users' mixed interests on this basis.

(3) Experiments on large-scale real-world datasets demonstrate that the proposed model's design can effectively improve the performance of news recommendation.

II. RELATED WORKS

In recent years, news personalization tasks have gained widespread attention in the fields of recommender and NLP. CF, as the most basic and well-known recommendation algorithm, was widely applied in news recommendation in earlier years. For example, the first CF-based news recommendation system, GroupLens, was proposed in [4], and an extensible collaborative recommendation solution for Google News was designed in [6]. With the development of deep learning, the mainstream of collaborative recommendation has shifted towards learning user and item representations in implicit spaces. For instance, literature [24] used Stacked Denoising Auto-Encoders (SDAE) to extract useful low-dimensional features from the original user-news interaction matrix to obtain user representation vectors. Then, cosine similarity was calculated for all user representations to complete Top-N recommendation. According to [23], at the model representation level, CF algorithms have undergone developments such as modeling individual users or items, modeling self-interaction history, and modeling the entire user-item interaction graph. For example, literature [7] designed a general CF recommendation framework called Neural Collaborative Filtering (NCF), which learns user and item representations using multi-layer perceptrons based on neural networks. In [8], Neural Graph Collaborative Filtering (NGCF) was proposed based on NCF, which captured high-order connectivity in the user-item interaction graph using graph neural networks. Subsequently, literature [9] proposed LightGCN, which achieved a more concise and better-performing model by removing the feature transformation and nonlinear activation modules from the embedding propagation mechanism of graph convolutional neural networks (GCN). These methods have achieved good results in general recommendation domains, such as product recommendations, but their effectiveness in direct application to news recommendation has been unsatisfactory. This is possibly due to the highly time-sensitive nature of news articles, with new articles being constantly published and existing ones quickly becoming outdated. The pure CF algorithm encounters severe data sparsity issues in the field of news recommendation. Therefore, how to effectively utilize the rich textual content of news articles is often the key to improving recommendation quality [25].

Compared to CF algorithms, hybrid recommendation methods combining news content and user interaction history are commonly used in news recommendation [33]. In the early stages, feature engineering models such as Term Frequency-Inverse Document Frequency (TF-IDF), Latent Semantic Analysis (LSA), LDA, and Bayesian models were frequently applied for extracting news features [18], [34], [35], [36]. With the development of neural networks, researchers started adopting deep learning techniques such as Convolutional Neural Networks (CNN) [37], Recurrent Neural Networks (RNN) [38], and attention mechanisms [39] in combination, often using a mixture of multiple techniques to improve recommendation effectiveness. For example, the NAML model [11] used a combination of CNN and attention mechanisms to extract news text features and then integrated all historical clicked news with attention weights to model the overall user interest. Literature [14] employed a combination of multi-head self-attention mechanisms and attention mechanisms to learn news representations. Literature [19] utilized Transformer [40] to learn news representations from news headlines and themes and employed one-hop and two-hop graph learning modules to capture high-order interaction relationships in the user-news interaction graph. Literature [10] introduced knowledge graphs into news recommendation to obtain more informative representations. Literature [28] used LSTM (Long Short Term Memory) to infer short-term user interests from their historical interaction sequences. Literature [13] combined CNN with attention mechanisms to extract news textual features and used a GRU network to model user short-term interests based on the sequence of news representations that the user clicked on.

Thanks to various effective representation learning methods in deep learning, the aforementioned works have achieved outstanding results, but there is still room for exploration. For example, at the news representation level, most methods only utilize the combination of headlines or headlines and themes, making it difficult to learn accurate news representations when the headlines are short and concise or ambiguous [11]. Additionally, at the user representation level, few methods explore users' mixed interests based on the fusion of different types of news information. Therefore, this paper emphasizes exploring users' mixed interests based on the comprehensive utilization of news textual content and additional information.

TABLE 1. Symbol definition.

Symbols	Descriptions
u	Target user
n	Candidate news
r_u	Representation vector of user u
r_n	Representation vector of news n
w_i	The i-th word in the word sequence
e_i	Word vector representation of w_i
c_i	Contextual representation of w_i
a_i	Attention weight of w_i
w_c, w_{sc}	Words describing categories and subcategories
e_c, e_{sc}	Word vector representations of w_c and w_{sc}
i_n	Implicit theme distribution vector of news n
$z_{n,i}$	Probability of news n belonging to implicit theme i
$a_H, a_A, a_C, a_{SC}, a_{IT}$	Attention weights of the headline, abstract, category, subcategory, and implicit themes
$r_H, r_A, r_C, r_{SC}, r_{IT}$	Representation vectors of the headline, abstract, category, subcategory, and implicit themes
M	Length of the historical click sequence of user u
N	Length of the word sequence
n_i	The i-th news article in the historical click sequence
r_i	Representation vector of n_i
h_i	Interest feature vector of user u at time i
r_S	Short-term interest representation of user u
r_L	Long-term interest representation of user u
y	The probability of target user u clicking on candidate news n

III. MATERIALS AND METHODS

The mathematical symbols and their descriptions used in this paper are shown in Table 1.

A. PROBLEM FORMULATION

Given a target user u and their historical click sequence $\{n_1, n_2, \ldots, n_M\}$, our objective is to predict whether user u will click on a candidate news article n that they have never seen before. For the candidate news article n, we simultaneously consider its headline, summary, category, subcategory, and implicit themes. Each headline or summary is represented as a word sequence composed of several words, denoted as $\{w_1, w_2, \ldots, w_N\}$. Each category or subcategory is described as a word or phrase, denoted as w_c and w_{sc}, respectively. The distribution of implicit themes $i_n = [z_{n,i}]_{i=1,2,\ldots,K}$, $\sum_{i=1}^{K} z_{n,i} = 1$ is extracted from the news body using Latent Dirichlet Allocation (LDA), where K represents the number of implicit themes, and $z_{n,i}$ represents the probability that news n belongs to implicit theme i.

B. NP-3C-FIP OVERALL ARCHITECTURE

The overall architecture of the proposed NP-C3-FIP model is shown in Figure 1.

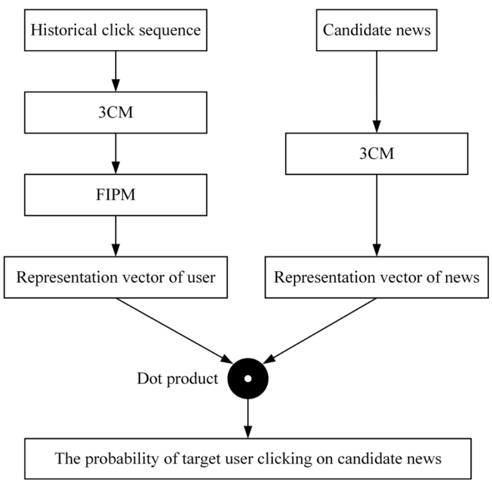

FIGURE 1. The NP-3C-FIP overall architecture.

The input consists of candidate news and the historical click sequence of the target user, and the output is the probability of the target user clicking on the candidate news. The NP-3C-FIP model mainly consists of 3CM and FIPM.

In 3CM, the model learns a unified news characterization vector based on various features of the news, such as the headline, summary, category, subcategory, and implicit themes.

In FIPM, the model first transforms the historical click news into news characterization vectors based 3CM. Then, the obtained news characterization sequence is fed into a GRU network to capture the sequential interest features of the user. The last time step's interest feature vector is used as the short-term interest portrayal of the user. Additionally, the model uses the last time step's interest feature as the query vector for an attention network to adaptively model frequent behaviors in the click sequence, obtaining the long-term interest portrayal of the user. Finally, the model concatenates the long-term and short-term user portrayals to obtain a unified user portrayal vector, and the user's click probability for the candidate news is calculated using vector dot product.

C. 3CM

3CM is used to learn unified news characterization vectors from various features of news, including the headline, summary, category, subcategory, and implicit themes, as shown in Figure 2.

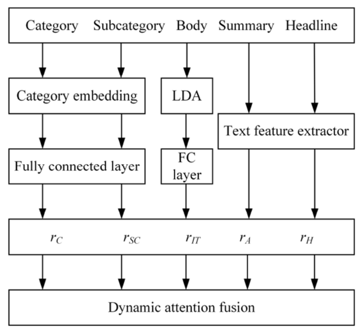

FIGURE 2. The 3CM structure.

It mainly consists of four components.

1) TEXT FEATURE EXTRACTOR

The first component of 3CM is the text feature extractor, which takes word sequences representing the news headline or summary as input and outputs representation vectors for the headline or summary, as shown in Figure 3.

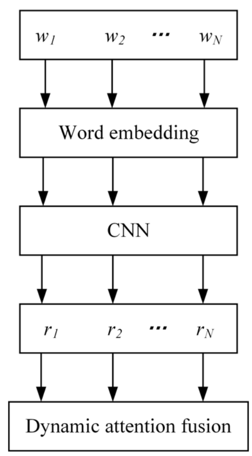

FIGURE 3. The text feature extractor structure.

The text feature extractor consists of a three-layer network structure.

The first layer is word embedding layer, which converts the word sequence into a low-dimensional sequence of word vectors. Let the word sequence be $\{w_1, w_2, \ldots, w_N\}$. By querying the word embedding table $W \in \mathbb{R}^{V \times D}$, the words in the sequence are transformed into word vectors, denoted as $E = [e_1, e_2, \ldots, e_N]^T \in \mathbb{R}^{N\times D}$, where N is the number of words in the sequence, V is the vocabulary size in the embedding table, and D is the embedding dimension.

The second layer is CNN, which captures the local contextual features of words through convolutional operations. According to [37], this paper sets the convolution operation as the inner product between the convolution kernel and the word vector. Let c_i be the contextual representation of w_i, and its calculation process is shown in (1).

$$c_i = \mathrm{ReLU}(M_c\ e_{i:i+l-1} + b_c) \tag{1}$$

In which, ReLU is the non-linear activation function, $e_{i:i+l-1}$ is the concatenation of l word vectors, $M_c \in \mathbb{R}^{N_f\times lD}$ and $b \in \mathbb{R}^{N_f}$ represent the convolution kernel and bias term, which are trainable parameters in the CNN model, N_f is the number of convolution kernels, and l is the size of the convolution kernel window. After this layer, the words in the sequence are transformed from word vectors to contextual characterization vectors that capture contextual features, denoted as $C = [c_1, c_2, \ldots, c_N]^T \in \mathbb{R}^{N\times N_f}$.

The third layer is a word-level attention network, which assigns different importance levels to different words through attention weighting and forms a unified text characterization vector. The proposed model initializes a fixed query vector randomly and uses the fully connected network with a single hidden layer to calculate the attention weights of each contextual characterization vector c_i relative to the query vector. Let the attention weight of w_i be a_i, and its calculation process is shown in (2).

$$a_i = \frac{\exp(q^T \tanh(M_q c_i + b_q))}{\sum_{j=1}^{N} \exp(q^T \tanh(M_q c_j + b_q))} \tag{2}$$

In which, $M_q \in \mathbb{R}^{Q\times N_f}$, $b_q \in \mathbb{R}^{Q}$, $q \in \mathbb{R}^{Q}$ are trainable parameters of the model, and q represents the query vector of the attention network, Q is the dimension of the query vector. The final text characterization is calculated as the weighted sum of the contextual characterizations and their attention weights, as shown in (3).

$$r_H = \sum_{i=1}^{N} a_i c_i \tag{3}$$

In which, r_H is the final characterization vector for the news headline. Similarly, the summary characterization vector r_A can be calculated.

2) IMPLICIT THEME EXTRACTOR

The second component of 3CM is the implicit theme extractor, which uses LDA to extract the distribution of implicit themes from the news body. Given a training corpus consisting of all news bodies, LDA generates the theme distribution for each news and the word distribution for each theme. LDA assumes that the news themes and theme words follow the Dirichlet prior distribution. For any news document n, its theme distribution is shown in (4):

$$i_n = Dirichlet(a) \tag{4}$$

In which,$a \in \mathbb{R}^{K}$ is the hyper-parameter of the distribution, a K-dimensional vector representing the number of implicit themes. For any theme k, its word distribution is shown in (5):

$$\alpha_k = Dirichlet(\beta) \tag{5}$$

In which,$\beta \in \mathbb{R}^{X}$ is also the hyper-parameter about the distribution, and X represents the total number of words in the corpus. To generate the n-th word in document n, the corresponding theme is first selected based on the theme distribution i_n, as shown in (6):

$$z_{\mathrm{nn}} = Multi(i_n) \tag{6}$$

Then, the final probability distribution of the word is obtained based on z_{n_n} and its word distribution $\alpha_{z_{n_\mathrm{n}}}$, as shown in (7):

$$w_{nn} = Multi(\alpha_{z_{nn}}) \tag{7}$$

The above is the LDA theme model, also known as the basic principle of generative probabilistic model. In which, i,α and z are trainable parameters, and K, a,β are the hyper-parameters of the model. We use the variational inference Expectation-Maximization (EM) algorithm in scikit-learn to train this model and obtain the implicit theme distribution $i_n = [z_{n,i}]_{i=1,2,\ldots,K}$, $\sum_{i=1}^{K} z_{n,i} = 1$ for candidate news n, where K is the number of implicit themes, and $z_{n,i}$ is the probability of news n belonging to implicit theme i. Finally, for weighted fusion of news features, the model feeds i_n into a fully connected network. This allows for the extraction of more features and the transformation of the theme distribution into a hidden space with the same dimensionality as the text characterization $r_H, r_A \in \mathbb{R}^{N_f}$. The forward propagation process of i_n in the fully connected network is shown in (8):

$$r_{IT} = \mathrm{ReLU}(M_c\ i_n + b_{IT}) \tag{8}$$

In which, $M_{IT} \in \mathbb{R}^{N_f\times K}$, $b_{IT} \in \mathbb{R}^{N_f}$ are trainable parameters of the model, and r_{IT} is the final characterization vector for the implicit themes.

3) CATEGORY FEATURE EXTRACTION MODULE

The third component of 3CM is the category feature extraction module, which is used to extract explicit thematic features from the news. Online news platforms, such as Microsoft News, often use categories (such as *Politics* and *Sports*) and subcategories (such as *Trump*and *NBA*) to label news articles. These categories indicate the explicit thematic information of the news and directly reflect the user's theme preferences.

The model first converts the words describing categories and subcategories into low-dimensional vectors using a category embedding layer, denoted as e_c and e_{sc}, respectively. Then, similar to the theme distribution, the model utilizes the fully connected network with a single hidden layer to learn the final characterizations of the category and subcategory.

The expressions of this process are shown in (9) and (10).

$$r_C = \mathrm{ReLU}\,(V_c\, e_c + v_c) \tag{9}$$
$$r_{SC} = \mathrm{ReLU}\,(V_{sc}\, e_{sc} + v_{sc}) \tag{10}$$

In which, $V_c, V_{sc} \in \mathbb{R}^{N_f \times D_c}$, and $v_c, v_{sc} \in \mathbb{R}^{N_f}$ represent trainable parameters.

4) ATTENTION NETWORK

The last component of 3CM is an attention network, which is used to model the varying importance levels of different news information and form a unified news characterization vector. Let's denote the attention weights for the headline, summary, category, subcategory, and implicit theme as a_H, a_A, a_C, a_{SC}, a_{IT}, respectively. We first use the neural network to fit the similarity between the query vector and each representation vector. Then, we apply the Softmax function to normalize the similarity results and obtain the weight coefficients. This process is shown in (11) and (12).

$$a_H = p^{\mathrm{T}} \tanh\,(V_H r_H + v_H) \tag{11}$$
$$a_H = \frac{\exp(a_H)}{\exp(a_H) + \exp(a_A) + \exp(a_C) + \exp(a_{SC}) + \exp(a_{IT})} \tag{12}$$

In which, $V_H \in \mathbb{R}^{Q \times N_f}$, $v_H \in \mathbb{R}^Q$, and $p \in \mathbb{R}^Q$ are trainable parameters. Similarly, we can compute a_A, a_C, a_{SC}, a_{IT}. Therefore, the final characterization of the candidate news article n is calculated as the weighted sum of the headline, summary, category, subcategory, and implicit theme characterizations, using their corresponding attention weights. This is represented by (13).

$$r_n = a_H r_H + a_A r_A + a_C r_C + a_{SC} r_{SC} + a_{IT} r_{IT} \tag{13}$$

D. FIPM

The FIPM structure as shown in Figure 4.

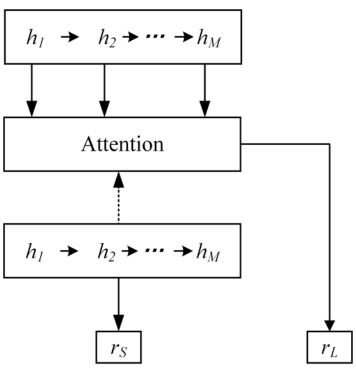

FIGURE 4. The FIPM structure.

Given a target user u and their historical click sequence $\{n_1, n_2, \ldots, n_M\}$, we first convert the historical clicked news into news characterization vectors based on 3CM. Let's denote the obtained news characterization sequence as $\{r_1, r_2, \ldots, r_M\}$. To capture the user's evolving dynamic interests over time, the model utilizes the GRU network to model the user's sequential interests from the historical click behavior. GRU is effective in handling sequential data and combines the current input with the previous hidden state output to compute the current hidden state output. This computation is repeated, and the information flow and amount are controlled by the reset gate x_t and update gate z_t. The specific calculations involved at each time step are shown in (14)-(17).

$$x_t = \sigma(M_1[h_{t-1}, r_t]) \tag{14}$$
$$z_t = \sigma(M_2[h_{t-1}, r_t]) \tag{15}$$
$$\tilde{h}_t = \tanh(M_3[x_t \odot h_{t-1}, r_t]) \tag{16}$$
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{17}$$

In which, r_t represents the vector characterization of news n_t, which serves as the input to the GRU network at time step t. h_{t-1} is the hidden state output of the GRU network at time step *t-1*, representing the user's interest feature vector from the previous time step. $\odot$ denotes element-wise multiplication. σ represents the sigmoid non-linear activation function. M_1, M_2, M_3 are trainable parameters of the model. As the final hidden state output h_M represents the user's interest feature vector at the current time step, we consider h_M as the user's short-term interest portrayal, denoted as $r_S \in \mathbb{R}^H$, where H is the dimension of the hidden vectors.

However, relying solely on sequential behavior may not be accurate enough, as users may accidentally click on wrong news or be attracted by unrelated news out of curiosity. Therefore, we introduce a personalized attention network on $\{h_1, h_2, \ldots, h_M\}$ to assign different importance weights to each click behavior. In 3CM, the query vector is trained along with the other network parameters through random initialization [11]. In this case, the query vector is set as the user's interest feature vector h_M at the last time step. Thus, each interest feature vector at the other time steps is evaluated for similarity with h_M, and the corresponding weight coefficients are obtained by normalizing the similarities using Softmax. This computation process is shown in (18).

$$a_i = \frac{\exp(h_M \tanh(V_i h_i + v_i))}{\sum_{j=1}^{M} \exp(h_M^T \tanh(V_j h_j + v_j))} \tag{18}$$

In which, $V_i \in \mathbb{R}^{Q \times H}$, $v_i \in \mathbb{R}^Q$ represent projection matrices, which are trainable parameters of the model. Finally, we compute the user's long-term interest portrayal as the weighted sum of all interest feature vectors at each time step, multiplied by their attention weights, as shown in (19).

$$r_L = \sum_{i=1}^{M} a_i h_i \tag{19}$$

Finally, the model concatenates the long-term and short-term user representations to obtain a unified user representation vector, i.e., $r_u = [r_S{:}r_L]$.

E. MODEL TRAINING

For online news service platforms, both user and news representations can be pre-computed offline in order to reduce recommendation latency, and the calculation of click-through rates (CTR), i.e., scoring mechanisms, as simple as possible [13]. Therefore, we use a simple vector dot product to calculate the click probability of the target user for candidate news, as shown in (20).

$$y = r_u^T r_n \tag{20}$$

Furthermore, in news recommendation, the ratio of positive to negative samples is highly unbalanced. Therefore, we employ a negative sampling strategy for model training. For each news article clicked by a user, i.e., a positive sample, we randomly sample R articles that appeared in the same session but were not clicked by the user, i.e., negative samples, and reconstruct the CTR prediction problem as an $R + 1$ classification task. Let y^+ and $[y_1^-, y_2^-, \ldots, y_R^-]$ represent the click scores of the positive sample and R negative samples, respectively. According to [41], we optimize this classification problem using cross-entropy loss function. Specifically, first, the click probabilities are softmax-normalized to compute the posterior click probability of positive samples, as shown in (21).

$$p_i = \frac{\exp(y_i^+)}{\exp(y_i^+ + \sum_{j=1}^{R} \exp(y_{i,j}^-))} \tag{21}$$

Then, the negative log-likelihood of all positive samples is used as the final loss function, as shown in (22).

$$\ell = -\sum_{i \in P} \log(p_i) \tag{22}$$

In which, P denotes the set of all positive news samples.

IV. EXPERIMENT

A. DATASETS

We conducted experiments on the Microsoft News Dataset (MIND) and MINDsmall.

1) The MIND dataset [25] is a large-scale news recommendation dataset collected from anonymous behavior logs from October 12, 2019, to November 22, 2019, spanning 6 weeks. The last week's data was used to construct the test set, while the 5th week's data was used to construct the training set, and the data from the last day of the training set was extracted to build the validation set. For the training data, the interaction history was built using the user's click behavior from the previous 4 weeks. For the test data, the interaction history was built using the user's click behavior from the previous 5 weeks.

2) The MINDsmall dataset is a lightweight version of the MIND dataset, constructed by randomly selecting 50,000 training users and their behavior logs from the MIND dataset. Following the format of the MIND dataset, this paper built the user interaction history using the click behavior from the previous 4 weeks. For the last week's data, the data from the first 5 days was used for training, the data from the 6th day was used for validation, and the data from the last day was used for testing.

Detailed statistical information about used datasets is shown in Table 2.

TABLE 2. Details of both data sets.

Statistics	MIND	MINDsmall
Number of users	1,000,000	94,057
Number of news	161,013	65,238
Number of sessions	24,155,470	230,117
News information	Headline, summary, category, subcategory, body	Headline, summary, category, subcategory, body

B. EVALUATION METRICS

To evaluate the performance of the proposed NP-3C-FIP system, we independently repeated each experiment 5 times and reported the average scores for each evaluation metric. The evaluation metrics used are as follows:

1) Area Under the ROC Curve (AUC) is commonly used for classification models, where a higher score indicates better model performance.

2) Mean Reciprocal Rank (MRR) calculates the reciprocal of the rank position where the user clicked on the news and takes the average. This metric reflects how prominent the position of the news is to the user and a higher score indicates a more significant impact on the recommendation system.

3) Normalized Discounted Cumulative Gain (nDCG) considers the ranking order of good recommendation results in the top-k list. This is important for the recommendation system because it is necessary to place good results in higher positions to give them a better chance of being chosen by users. We used nDCG@5 and nDCG@10 as evaluation metrics.

C. EXPERIMENTAL SETUP

Our model was implemented using TensorFlow. The hyper-parameters were determined by optimizing the Area Under Curve (AUC) on the validation set. We initialized the word embedding matrix W with a pre-trained 300-dimensional GloVe (Global Vectors for Word Representation) technique [42]. The word embedding dimension, D, was set to 300, the category embedding dimension, D_c, was set to 100, the number of CNN convolutional filters, N_f, was set to 300, the window size, l, was set to 3. The dimension of the attention query vector, Q and GRU hidden vector, H, was set to 200. The number of words in the headline and summary, n, was set to 20 and 50, respectively. The length of the historical click sequence, m, was set to 50. The number of implicit themes, K, was set to 50, and the negative sampling rate, R, was set to 4. In addition, to prevent over-fitting, the model used a 0.2 dropout strategy after the word embedding and CNN outputs [43]. Adam optimization [44] was used for model optimization with a batch size of 64 and a learning rate of 0.001.

TABLE 3. Contrast experimental results.

Methods	MIND				MINDsmall			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
LightGCN	0.629 1	0.294 7	0.315 9	0.373 5	0.499 7	0.219 3	0.223 8	0.286 8
DKN	0.634 2	0.296 7	0.318 3	0.376 0	0.609 3	0.276 2	0.301 9	0.366 4
NPA	0.635 4	0.298 6	0.320 5	0.378 4	0.583 9	0.2606 9	0.279 1	0.339 9
NRMS	0.646 5	0.307 8	0.332 1	0.389 7	0.616 2	0.273 9	0.298 7	0.364 8
LSTUR	0.656 7	0.312 0	0.337 2	0.394 6	0.615 8	0.281 1	0.303 7	0.366 5
GERL	0.666 1	0.320 9	0.348 9	0.406 2	0.600 7	0.272 3	0.291 4	0.355 9
NAML	0.687 0	0.336 2	0.366 7	0.423 6	0.643 5	0.295 5	0.321 9	0.386 7
NP-3C-FIP	0.695 1	0.340 9	0.372 5	0.430 1	0.660 5	0.309 3	0.343 0	0.404 6

D. BENCHMARKS

To validate the effectiveness of our method, we compared the proposed NP-3C-FIP model with 7 state-of-the-art(SOTA)benchmark models, including 1 CF-based model that does not utilize news content information, and 6 hybrid models that incorporate news content information.

1) LightGCN [9] is an advanced CF recommendation model that uses their own IDs as the original features of the user and the item and employs a simplified GCN (Graph Convolutional Network) embedding propagation mechanism to capture high-order collaborative signals on the user-item interaction graph. To apply the model to the MIND dataset, we randomly sampled 30 neighbors for each node to construct the user-news adjacency matrix. However, for the MINDsmall dataset, we utilized all the interaction data.

2) DKN [10] is a hybrid news recommendation model that integrates knowledge graph information. It learns news representations based on headlines and entities and user representations based on the similarity between candidate news and historical clicked news.

3) NAML [11] is a hybrid news recommendation model that combines headline, theme, and body information. It is also an influential method that inspires the proposed model.

4) NPA [12] is a hybrid news recommendation model that utilizes headline information. It replaces the query vector of the attention network with the user ID and selects important words and news based on user preferences.

5) LSTUR [13] is a hybrid news recommendation method that considers both user short-term and long-term interests. It uses GRU to learn the user's short-term interest representation from historical clicked news and employs the user ID as the representation of long-term interest.

6) NRMS [14] is a hybrid news recommendation method that utilizes headline information. It captures long-term interaction relationships between words based on a multi-head self-attention mechanism.

7) GERL [19] is a hybrid news recommendation method that considers high-order user-news interaction relationships. It uses one-hop and two-hop learning modules to capture first-order and second-order interaction relationships in the user-news interaction graph.

E. CONTRAST EXPERIMENT

The experimental results of the proposed NP-3C-FIP model compared to the benchmark methods are shown in Table 3.

Based on Table 3, the following observations can be made:

1) The hybrid recommendation models that incorporate news content information outperform the CF model, i.e., LightGCN. This may be because pure collaborative filtering algorithms encounter serious sparsity issues in the user-item rating matrix in news recommendation. Therefore, leveraging the rich textual content of news articles could be a key factor in improving recommendation quality.

2) Models that incorporate multiple news information such as headline, body, and category (e.g., LSTUR, GERL, and NAML) perform better than models that only utilize title information (e.g., NRMS and NPA). This is likely because the rich content information helps learn more accurate representations of news. It is worth noting that the NAML model achieves the best performance among all benchmark models, and it is the only benchmark model that combines headline, body, and category information, demonstrating the effectiveness of integrating multiple news information.

3) The NP-3C-FIP model consistently outperforms all benchmark models. This may be because the proposed model fully utilizes the textual content, explicit themes, and implicit themes of news, and considers the mixed interests of users, resulting in informative news and user representations. Additionally, compared to the NAML model, the proposed model uses LDA theme modeling to extract the implicit theme distribution from the news body instead of treating the body text as a fixed length, such as retaining the first 100 words and directly inputting them into the model, which effectively utilizes the body text.

4) It is worth noting that the performance of the six hybrid recommendation models on the MINDsmall dataset is not entirely consistent with the results on the MIND dataset. This may be due to the models' different adaptability to different datasets. Nevertheless, they still outperform the LightGCN model, which is sufficient to demonstrate the effectiveness of utilizing news content information.

F. ABLATION EXPERIMENT

To further investigate the effectiveness of the components in the proposed model, namely 3CM and FIPM, we conducted

TABLE 4. Ablation experimental results.

Methods	MIND				MINDsmall			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
Headline-ONLY	0.635 4	0.298 6	0.320 5	0.378 4	0.629 3	0.284 3	0.310 2	0.375 6
Category-ONLY	0.656 0	0.314 6	0.340 3	0.397 4	0.621 5	0.289 7	0.317 5	0.378 9
Summary-ONLY	0.658 4	0.315 2	0.340 7	0.398 3	0.633 7	0.293 6	0.321 6	0.384 5
Implicit theme-ONLY	0.658 8	0.314 0	0.338 5	0.396 8	0.634 7	0.303 2	0.330 9	0.393 7
Sum average-ONLY	0.673 6	0.326 0	0.353 6	0.411 7	0.641 5	0.290 0	0.323 0	0.385 1
Attention mechanism-ONLY	0.681 8	0.331 6	0.361 4	0.418 8	0.645 6	0.301 2	0.332 4	0.395 5
Short-term interest -ONLY	0.692 6	0.339 5	0.370 5	0.427 7	0.640 9	0.299 0	0.329 5	0.392 2
NP-3C-FIP	0.695 1	0.340 9	0.372 5	0.430 1	0.660 5	0.309 3	0.343 0	0.404 6

experiments on various ablation variants of the NP-3C-FIP model, and the results are shown in Table 4.

1) we compared the impact of different types of news information, such as headline, summary, category, and implicit themes, on the model. From Table 4, it can be observed that implicit themes outperformed headline, summary, and category information. This may be because implicit themes are extracted from the news body, which contains the original text of the news and provides rich training data for theme modeling, resulting in accurate implicit theme distributions. Additionally, category information (including category and subcategory) also achieved good performance as they reveal the explicit themes of news articles, directly reflecting users' theme preferences. Moreover, the experimental performance of the summary was significantly better than that of the news headline, indicating that the summary provides richer and more detailed content information, further confirming that rich textual content leads to better recommendation results. Furthermore, although news headlines are concise, they have a decisive influence on user reading behavior, and thus using only news headlines can achieve decent recommendation results. Finally, compared to using only news headlines, the complete NP-3C-FIP model improved the AUC score by approximately 0.06 on the MIND dataset and by approximately 0.03 on the MINDsmall dataset, demonstrating the effectiveness of integrating different types of news information and the complementary role of different news information in learning news representations.

2) we explored the effectiveness of user long-term and short-term representation models. From Table 4, it can be observed that using attention mechanisms for dynamic fusion or modeling user sequential interests with recurrent neural networks yielded better experimental performance compared to simply summing and averaging historical interacted news. Furthermore, from the last two experiments, it can be seen that removing the user long-term interest representation module resulted in a decrease in all evaluation metric scores, demonstrating the effectiveness of exploring the mixture of user long-term and short-term interests.

G. THE STUDY ABOUT HYPER-PARAMETERS

To explore the impact of various hyper-parameters on the performance of the NP-3C-FIP model, relevant experiments were conducted on the MINDsmall dataset.

1) we examined the number of convolutional kernels, i.e., N_f, and the window size, i.e., l in the CNN, which determine the model's ability to capture local contextual features of words.

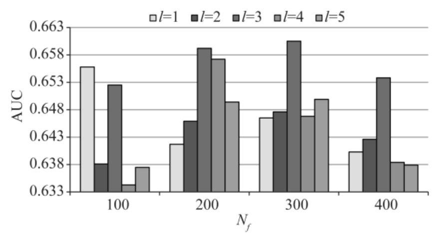

FIGURE 5. Effect of N_f and l on model.

As shown in Figure 5, overall, a window size of $l = 3$ and N_f =300 yielded the best model performance. Additionally, given a window size, the AUC score initially increased with an increase in the number of convolutional kernels, as more kernels imply capturing more local contextual features. However, when N_f becomes too large, such as 400, the trend may change due to over-fitting. Similar patterns can be observed for window size, where too small window fails to capture long-term contextual features, and excessively large windows are susceptible to over-fitting due to noise.

2) we investigated the length of the user's historical click sequence, i.e., M. Generally, more click records provide more clues for modeling personalized interests. However, M cannot be infinitely large as it is limited by the dataset.

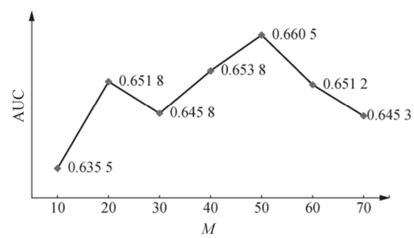

FIGURE 6. Effect of M on model.

As shown in Figure 6, on the MINDsmall dataset, a length of $M = 50$ yielded the best model performance. This result suggests that inferring personalized interests for users with extremely scarce behaviors on the platform is quite challenging.

3) In the category feature extractor, the model first converts category and subcategory into low-dimensional vectors using a category embedding layer. In general, encoding more bits can capture more useful information about category descriptions and aid in modeling user theme preferences.

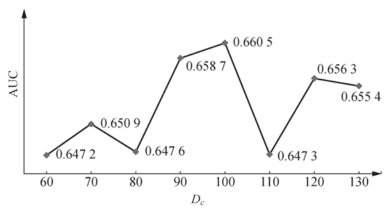

FIGURE 7. Effect of ***Dc*** on model.

As shown in Figure 7, when the dimensionality of the category embedding, i.e., D_c, is set to 100, the proposed model achieved optimal performance.

4) In the text feature extractor, the model takes word sequences representing news headlines or summaries as input and generates the corresponding representation vectors. Therefore, experiments were conducted to explore the lengths of headlines or summaries.

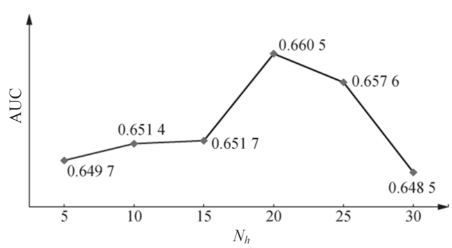

FIGURE 8. Effect of N_h on model.

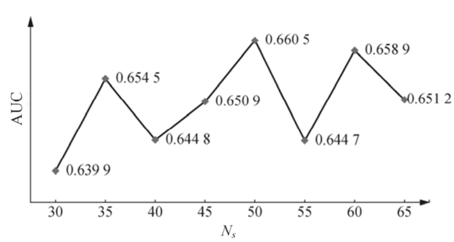

FIGURE 9. Effect of N_s on model.

As shown in Figures 8 and 9, the model achieved the best performance when the length of headline, i.e., N_h or the length of summary, i.e., N_s were set to 20 and 50, respectively. This result is intuitive, as news summaries are usually slightly longer than headlines. Furthermore, Figure 8 demonstrates a clear trend of initially increasing and then decreasing performance. This can be explained by the fact that longer word sequences provide more information, allowing for more accurate modeling of news semantics. However, excessively long word sequences may introduce noise and degrade model performance.

V. CONCLUSION

We propose a news recommendation method called NP-3C-FIP in this paper. In general, the proposed model consists of 3CM and FIPM. The former can fully characterize the complete news information, including text details, explicit themes, and implicit themes. The latter thoroughly captures the user's mixed long-term and short-term interests to explore their immediate needs and future favors. Experiments demonstrate that the design framework of proposed model effectively improves the performance of news recommendation. In future work, we will conduct in-depth research on the cold-start user problem in news recommendation.

REFERENCES

[1] J. Hendrickx, A. Smets, and P. Ballon, "News recommender systems and news diversity, two of a kind? A case study from a small media market," *Journalism Media*, vol. 2, no. 3, pp. 515–528, Sep. 2021, doi: 10.3390/journalmedia2030031.

[2] H. Abdollahpouri, E. Malthouse, J. A. Konstan, B. Mobasher, and J. Gilbert, "Toward the next generation of news recommender systems," in *Proc. Companion Web Conf.*, Apr. 2021, pp. 402–406, doi: 10.1145/3442442.3452327.

[3] M. Wieland, G. V. Nordheim, and K. K.-V. Königslöw, "One recommender fits all? An exploration of user satisfaction with text-based news recommender systems," *Media Commun.*, vol. 9, no. 4, pp. 208–221, Nov. 2021, doi: 10.17645/mac.v9i4.4241.

[4] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, "Applying collaborative filtering to Usenet news," *Commun. ACM*, vol. 40, no. 3, pp. 77–87, 1997.

[5] J. Mulder and P. D. Hoff, "A latent variable model for relational events with multiple receivers," 2021, *arXiv:2101.05135*.

[6] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: Scalable online collaborative filtering," in *Proc. 16th Int. Conf. World Wide Web*, May 2007, pp. 1–22.

[7] R. Raziperchikolaei and Y.-J. Chung, "Simultaneous learning of the inputs and parameters in neural collaborative filtering," 2022, *arXiv:2203.07463*.

[8] J. Mizem-Pietraszko and L. K. Wroclawia, "Neural graph collaborative filtering," *Comput. Rev.*, vol. 2, p. 62, 2021.

[9] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "LightGCN: Simplifying and powering graph convolution network for recommendation," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 639–648, doi: 10.1145/3397271.3401063.

[10] Y. Du, X. Zhu, L. Chen, B. Zheng, and Y. Gao, "HAKG: Hierarchy-aware knowledge gated network for recommendation," 2022, *arXiv:2204.04959*.

[11] G. Palumbo, D. Carneiro, M. Guimares, V. Alves, and P. Novais, "Algorithm recommendation and performance prediction using meta-learning," *Int. J. Neural Syst.*, vol. 33, no. 3, Mar. 2023, Art. no. 2350011, doi: 10.1142/S0129065723500119.

[12] C. Wu, F. Wu, M. An, J. Huang, Y. Huang, and X. Xie, "NPA: Neural news recommendation with personalized attention," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 2576–2584.

[13] Z. Qiu, Y. Hu, and X. Wu, "Graph neural news recommendation with user existing and potential interest modeling," *ACM Trans. Knowl. Discovery Data (TKDD)*, vol. 16, no. 5, pp. 1–17, Oct. 2022.

[14] Z. Mao, X. Zeng, and K.-F. Wong, "Neural news recommendation with collaborative news encoding and structural user encoding," 2021, *arXiv:2109.00750*.

[15] Z. Ji, M. Wu, H. Yang, and J. E. A. Ìñigo, "Temporal sensitive heterogeneous graph neural network for news recommendation," *Future Gener. Comput. Syst.*, vol. 125, pp. 324–333, Dec. 2021, doi: 10.1016/j.future.2021.06.007.

[16] Y. Pang, Y. Zhang, J. Tong, and Z. Wei, "Time-aware attentive neural network for news recommendation with long- and short-term user representation," in *Knowledge Science, Engineering and Management*. Cham, Switzerland: Springer, 2020, doi: 10.1007/978-3-030-55393-7_7.
[17] Y. Hu, Z. Qiu, and X. Wu, "Denoising neural network for news recommendation with positive and negative implicit feedback," 2022, *arXiv:2204.04397*.
[18] J. Yi, F. Wu, C. Wu, Q. Li, G. Sun, and X. Xie, "DebiasedRec: Bias-aware user modeling and click prediction for personalized news recommendation," 2021, *arXiv:2104.07360*.
[19] C. Huang, J. Chen, L. Xia, Y. Xu, P. Dai, Y. Chen, L. Bo, J. Zhao, and J. X. Huang, "Graph-enhanced multi-task learning of multi-level transition dynamics for session-based recommendation," 2021, *arXiv:2110.03996*.
[20] C. Wu, S. Liu, Z. Zeng, M. Chen, A. Alhudhaif, X. Tang, F. Alenezi, N. Alnaim, and X. Peng, "Knowledge graph-based multi-context-aware recommendation algorithm," *Inf. Sci.*, vol. 595, pp. 179–194, May 2022, doi: 10.1016/j.ins.2022.02.054.
[21] R. Xie, Q. Liu, S. Liu, Z. Zhang, P. Cui, B. Zhang, and L. Lin, "Improving accuracy and diversity in matching of recommendation with diversified preference network," *IEEE Trans. Big Data*, vol. 8, no. 4, pp. 955–967, Aug. 2022.
[22] S. Raza and C. Ding, "A regularized model to trade-off between accuracy and diversity in a news recommender system," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 551–560, doi: 10.1109/BigData50022.2020.9378340.
[23] L. Wu, X. He, X. Wang, K. Zhang, and M. Wang, "A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation," 2021, *arXiv:2104.13030*.
[24] S. Cao, N. Yang, and Z. Liu, "Online news recommender based on stacked auto-encoder," in *Proc. IEEE/ACIS 16th Int. Conf. Comput. Inf. Sci. (ICIS)*, May 2017, pp. 721–726, doi: 10.1109/ICIS.2017.7960088.
[25] G. Hu and Q. Yang, "TrNews: Heterogeneous user-interest transfer learning for news recommendation," 2021, *arXiv:2101.05611*.
[26] F. Yang, H. Wang, and J. Fu, "Improvement of recommendation algorithm based on collaborative deep learning and its parallelization on spark," *J. Parallel Distrib. Comput.*, vol. 148, pp. 58–68, Feb. 2021, doi: 10.1016/j.jpdc.2020.09.014.
[27] H.-S. Sheu and S. Li, "Context-aware graph embedding for session-based news recommendation," in *Proc. 14th ACM Conf. Recommender Syst.*, Sep. 2020, pp. 657–662, doi: 10.1145/3383313.3418477.
[28] J. Ni, Z. Huang, J. Cheng, and S. Gao, "An effective recommendation model based on deep representation learning," *Inf. Sci.*, vol. 542, pp. 324–342, Jan. 2021.
[29] C. Feng, C. Shi, C. Liu, Q. Zhang, S. Hao, and X. Jiang, "Context-aware item attraction model for session-based recommendation," *Expert Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114834, doi: 10.1016/j.eswa.2021.114834.
[30] F. Liu, W. Guo, and H. Wang, "Framework model of personalized learning recommendation system based on deep learning under the background of big data," in *Proc. Int. Conf. Frontier Comput., Theory, Technol. Appl.*, Feb. 2020, pp. 1141–1148.
[31] A. Hanselowski, H. Zhang, Z. Li, D. Sorokin, B. Schiller, C. Schulz, and I. Gurevych, "UKP-Athene: Multi-sentence textual entailment for claim verification," in *Proc. 1st Workshop Fact Extraction Verification (FEVER)*, 2018, pp. 103–108, doi: 10.18653/v1/W18-5516.
[32] S. Zhou, P. Kan, Q. Huang, and J. Silbernagel, "A guided latent Dirichlet allocation approach to investigate real-time latent topics of Twitter data during Hurricane Laura," *J. Inf. Sci.*, vol. 49, no. 2, pp. 465–479, Apr. 2023, doi: 10.1177/01655515211007724.
[33] J. Feng, Z. Xia, X. Feng, and J. Peng, "RBPR: A hybrid model for the new user cold start problem in recommender systems," *Knowl.-Based Syst.*, vol. 214, Feb. 2021, Art. no. 106732, doi: 10.1016/j.knosys.2020.106732.
[34] T. Qi, F. Wu, C. Wu, and Y. Huang, "PP-Rec: News recommendation with personalized user interest and time-aware news popularity," 2021, *arXiv:2106.01300*.
[35] H. Hara, Y. Fujita, and K. Tsuda, "Population estimation using Twitter for a specific space," *Data Technol. Appl.*, vol. 55, no. 3, pp. 430–445, Jun. 2021, doi: 10.1108/DTA-03-2020-0065.
[36] Z. Peng, D. Wan, A. Wang, X. Lu, and P. M. Pardalos, "Deep learning-based recommendation method for top-K tasks in software crowdsourcing systems," *J. Ind. Manage. Optim.*, vol. 19, no. 9, pp. 6478–6499, 2023, doi: 10.3934/jimo.2022223.
[37] M. Zhao, J. Yang, J. Zhang, and S. Wang, "Aggregated graph convolutional networks for aspect-based sentiment classification," *Inf. Sci.*, vol. 600, pp. 73–93, Jul. 2022.
[38] B. Ahmadnia and B. J. Dorr, "Enhancing phrase-based statistical machine translation by learning phrase representations using long short-term memory network," in *Proc. Recent Adv. Natural Language Process. (RANLP)*, Sep. 2019, pp. 25–32, doi: 10.26615/978-954-452-056-4_004.
[39] J. Song, S. Kim, and S. Yoon, "AligNART: Non-autoregressive neural machine translation by jointly learning to estimate alignment and translate," 2021, *arXiv:2109.06481*.
[40] C. Chen, Y. Zha, D. Zhu, K. Ning, and X. Cui, "Attention is all you need for general-purpose protein structure embedding," Cold Spring Harbor Lab., Laurel Hollow, NY, USA, Tech. Rep., 2021, doi: 10.1101/2021.01.31.428935.
[41] J. Ider, A. Oliveira, R. Rubinger, A. K. Silva, A. Assini, G. Tiago-Filho, and M. Baldissera, "Concentrated solar power with thermoelectric generator—An approach using the cross-entropy optimization method," *Energies*, vol. 15, no. 13, p. 4774, Jun. 2022, doi: 10.3390/en15134774.
[42] M. Sharma, I. Kandasamy, and W. B. Vasantha, "Comparison of neutrosophic approach to various deep learning models for sentiment analysis," *Knowl.-Based Syst.*, vol. 223, Jul. 2021, Art. no. 107058, doi: 10.1016/j.knosys.2021.107058.
[43] J. Shunk, "Neuron-specific dropout: A deterministic regularization technique to prevent neural networks from overfitting & reduce dependence on large training samples," 2022, *arXiv:2201.06938*.
[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

WENWEN FU was born in Weifang, Shandong, China, in 1983. She received the Ph.D. degree from Beijing Sport University, China. She is currently with the College of Media and Communications, Weifang University. Her research interests include new media dissemination, computational journalism, and cultural dissemination.