DIGAT: Modeling News Recommendation with Dual-Graph Interaction

Zhiming Mao\textsuperscript{1,2}, Jian Li\textsuperscript{3}, Hongru Wang\textsuperscript{1,2}, Xingshan Zeng\textsuperscript{4}, Kam-Fai Wong\textsuperscript{1,2}

\textsuperscript{1}The Chinese University of Hong Kong, Hong Kong, China
\textsuperscript{2}MoE Key Laboratory of High Confidence Software Technologies, China
\textsuperscript{3}Tencent, Shenzhen, China

\textsuperscript{1,2}\{zmmao,hrwang,kfwong\}@se.cuhk.edu.hk
\textsuperscript{3}lijianjack@gmail.com
\textsuperscript{4}zxshamson@gmail.com

Abstract

News recommendation (NR) is essential for online news services. Existing NR methods typically adopt a news-user representation learning framework, facing two potential limitations. First, in news encoder, single candidate news encoding suffers from an insufficient semantic information problem. Second, existing graph-based NR methods are promising but lack effective news-user feature interaction, rendering the graph-based recommendation suboptimal. To overcome these limitations, we propose dual-interactive graph attention networks (DIGAT) consisting of news- and user-graph channels. In the news-graph channel, we enrich the semantics of single candidate news by incorporating the semantically relevant news information with a semantic-augmented graph (SAG). In the user-graph channel, multi-level user interests are represented with a news-topic graph. Most notably, we design a dual-graph interaction process to perform effective feature interaction between the news and user graphs, which facilitates accurate news-user representation matching. Experiment results on the benchmark dataset MIND show that DIGAT outperforms existing news recommendation methods\textsuperscript{1}. Further ablation studies and analyses validate the effectiveness of (i) semantic-augmented news graph modeling and (ii) dual-graph interaction.

1 Introduction

News recommendation is an important technique to provide people with the news which satisfies their personalized reading interests (Okura et al., 2017; Wu et al., 2020). Effective news recommender systems require both accurate textual modeling on news content (Wang et al., 2018; Wu et al., 2019d; Wang et al., 2020) and personal-interest modeling on user behavior (Hu et al., 2020b; Qi et al., 2021c). Hence, most news recommendation methods (An et al., 2019; Wu et al., 2019a,b,c,d; Ge et al., 2020;\textsuperscript{1}) adopt a news-user representation learning framework to learn discriminative news and user representations, as illustrated in Figure 1.

Though promising, there are still two potential limitations in the existing news recommendation framework. First, in news encoder, single candidate news encoding suffers from an insufficient semantic information problem. Unlike long-term items in common recommendation (e.g., E-commerce product recommendation), the candidate news items are short-term and suffer from the cold start problem.

In the real-world setting, news recommender systems usually handle the latest news, where existing user-click interactions are always not available\textsuperscript{2}. Hence, it is intractable to use existing user-click records to enrich the information of candidate news. On the other hand, compared to abundant historical clicked news in user encoder, the single candidate news may not contain sufficient semantic information for accurate news-user representation matching in the click prediction stage. Prior studies (Wu et al., 2019a,c; Qi et al., 2021c) pointed out that users were usually interested in specific news topics (e.g., Sports). Empirically, the text of single candidate news does not contain enough syntactic and semantic information to accurately represent a genre of news topic and match user interests.

\textsuperscript{1}Our code is available at https://github.com/Veason-silverbullet/DIGAT. Jian Li is the corresponding author.

\textsuperscript{2}From the viewpoint of experimental dataset, most candidate news in test data does not appear in training user history.
Second, previous studies generally follow two research directions to model user history, i.e., sequence and graph modeling. Formulating user history as a sequence of user’s clicked news is a more prevalent direction, based on which time-sequential models (Okura et al., 2017; An et al., 2019; Qi et al., 2021b) and attentive models (Zhu et al., 2019; Wu et al., 2019a,b,d; Qi et al., 2021a,c) are proposed. Besides, graph modeling is proved effective for recommender systems (Chen et al., 2020). Ge et al. (2020) and Hu et al. (2020b) formulate news and users jointly in a bipartite graph to model news-user interaction. However, since most candidate news in test data has no existing interaction with users (i.e., cold-news), the isolated cold-news nodes cause this bipartite graph modeling degenerate. Recent works formulate user history as heterogeneous graphs and employ advanced graph learning methods to extract the user-graph representations (Hu et al., 2020a; Mao et al., 2021; Wu et al., 2021). These works focus on how to extract fine-grained representations from the user-graph side but neglect necessary feature interaction between the candidate news and user-graphs.

In this work, we propose Dual-Interactive Graph ATtenion networks (DIGAT) to address the aforementioned limitations. DIGAT consists of news- and user-graph channels to encode the candidate news and user history, respectively. In the news-graph channel, we introduce semantic-augmented graph (SAG) modeling to enrich the semantic representation of the single candidate news. In SAG, the original candidate news is regarded as the root node, while the semantic-relevant news documents are represented as the extended nodes to augment the semantics of the candidate news. We integrate the local and global contexts of SAG as the semantic-augmented candidate news representations.

In the user-graph channel, motivated by Mao et al. (2021) and Wu et al. (2021), we model user history with a news-topic graph to represent multi-levels of user interests. Most notably, we design a dual-graph interaction process to learn news- and user-graph representations with effective feature interaction. Different from the individual graph attention network (Veličković et al., 2018), DIGAT updates news and user graph embeddings with the interactive attention mechanism. Particularly, in each layer of the dual-graph, the user (news) graph context is incorporated into its dual news (user) node embedding learning iteratively.

Extensive experiments on the benchmark dataset MIND (Wu et al., 2020) show that DIGAT significantly outperforms the existing news recommendation methods. Further ablation studies and analyses confirm that semantic-augmented news graph modeling and dual-graph interaction can substantially improve news recommendation performance.

2 Related Work

Personalized news recommendation is important to online news services (Okura et al., 2017; Yi et al., 2021). Existing neural news recommendation methods typically aim to learn informative news and user representations (Wang et al., 2018; Zhu et al., 2019; An et al., 2019; Wu et al., 2019a,b,d; Liu et al., 2020; Wang et al., 2020; Qi et al., 2021a,b,c; Wu et al., 2021; Li et al., 2022). For example, An et al. (2019) used a CNN network to extract textual representations from news titles and used a GRU network to learn short-term user interests combined with long-term user embeddings. The matching probabilities between candidate news and users are computed over the learned news and user representations. Wu et al. (2019d) utilized multi-head self-attention networks to learn informative news and user representations from news titles and user clicked history. These methods regarded the single candidate news as the input to news encoder, which may not contain sufficient semantics to represent a user-interested news topic. Different from these methods, we encode the candidate news with semantic-augmented graphs to enrich its semantic representations. More recently, graph-based methods were proposed for news recommendation (Ge et al., 2020; Hu et al., 2020a,b; Mao et al., 2021; Wu et al., 2021). For example, Wu et al. (2021) proposed a heterogeneous graph pooling method to learn fine-grained user representations. However, feature interaction between candidate news and users is inadequate or neglected in these methods. In contrast, our approach models effective feature interaction between news and user graphs for accurate news-user representation matching.

3 Approach

Problem Formulation. Denote the clicked-news history of a user $u$ as $H_u = [n_1, n_2, \ldots, n_H]$, containing $|H|$ clicked news items. For the news $n$, its textual content consists of a sequence of $|T|$ words as $T_n = [w_1, w_2, \ldots, w_{|T|}]$. Based on $H_u$ and $T_n$, the goal of news recommendation is to predict the
score \( \delta_{n,u} \), which indicates the probability of the user \( u \) clicking the candidate news \( n_{can} \). The recommendation result is generated by ranking the user-click scores of multiple candidate news items.

### 3.1 News Semantic Representation

We introduce how to extract semantic representation from news content text \( T_n = [w_1, w_2, ..., w_T] \). Our news encoder first maps the news word tokens into word embeddings \( E_n = [e_1, e_2, ..., e_T] \). Then, we use the multthead self-attention network MSA(\( Q, K, V \)) of Transformer encoder (Vaswani et al., 2017) to learn the contextual representations \( H_n \in \mathbb{R}^{T \times d} \) (where \( d \) is the feature dimension). Finally, we employ an attention network \( f_{\text{att}}(\cdot) \) to aggregate the news semantic representation \( h \in \mathbb{R}^d \):

\[
H_n = \text{MSA}(Q = E_n, K = E_n, V = E_n)
\]

\[
h = f_{\text{att}}(\text{ReLU}(H_n))
\]

The attentive aggregation function \( f_{\text{att}}(\cdot) \) is implemented by a feed-forward network in our experiments. It is worth noting that the semantic news encoder in our framework is plug-and-play, which can be easily replaced by any other textual encoders or pretrained language models, e.g., BERT (Devlin et al., 2019) or DeBERTa (He et al., 2021).

### 3.2 News Graph Encoding Channel

In this section, we will explain the news semantic-augmented graph (SAG) construction and graph context learning. Our motivation is to retrieve semantic-relevant news from training corpus and construct a semantic-augmented graph to enrich the semantics of the original single candidate news.

#### 3.2.1 News Graph Construction

**Semantic-relevant News Retrieval.** Pretrained language models (PLM) have achieved remarkable performance (Reimers and Gurevych, 2019, 2020) on semantic textual similarity (STS) benchmarks. Motivated by Lewis et al. (2020), we utilize a PLM \( \phi(\cdot) \) to retrieve semantic-relevant news from training news corpus\(^3\) to augment the semantic information of the original single candidate news. In the retrieval process, the semantic similarity score \( s_{i,j} \) of news \( n_i \) and \( n_j \) (corresponding texts \( T_i \) and \( T_j \)) is computed by the similarity function \( \text{sim}(\cdot, \cdot) \):

\[
s_{i,j} = \text{sim}(n_i, n_j) = \cosine(\phi(T_i), \phi(T_j)) \tag{2}
\]

**Semantic-augmented Graph (SAG).** For the original candidate news \( n_{can} \), we initialize it as the root node \( v_0 \) of the semantic-augmented news graph \( G_n \). We build \( G_n \) by repeatedly extending semantic-relevant neighboring nodes to existing nodes of \( G_n \). In each graph construction step, for an existing node \( v_i \) (corresponding news \( N_i \) of \( G_n \)), \( M \) news documents \( \{N_j\}_{j=1}^M \) are retrieved from the news corpus \( \{N_G\} \) with the highest semantic similarity scores \( s_{i,j} \). We extend the nodes \( \{v_j\}_{j=1}^M \) as neighboring nodes to the node \( v_i \) by adding bidirectional edge \( \{e_{i,j}\}_{j=1}^M \) between them. To heuristically discover semantic-relevant news in higher-order relations, we repeatedly extend the semantic-relevant news nodes within \( K \) hops from the root node. The scale of news graph \( G_n \) is approximated to be \( O(M^K) \). Detailed SAG construction and qualitative analysis are provided in Appendix A.

#### 3.2.2 News Graph Context Extraction

Given an SAG \( G_n \) generated from the candidate news node \( v_0 \) with \( N \) semantic-relevant news nodes \( \{v_j\}_{j=1}^N \), we use the semantic news encoder (introduced in Section 3.1) to extract their semantic representations as \( h_{n,0} \in \mathbb{R}^d \) and \( \{h_{n,i}\}_{i=1}^N \in \mathbb{R}^{N \times d} \).

We aim to extract the graph context \( c_n \in \mathbb{R}^d \) which augments the semantics of the candidate news \( n_{can} \) by aggregating the information of \( G_n \). We consider the original semantics of the candidate news preserved in the root node \( v_0 \) and regard the local graph context as \( h_{n,0} = h_{n,0} \in \mathbb{R}^d \). Besides, we employ an attention module to aggregate the global graph context \( h^G_n \in \mathbb{R}^d \) from the semantic-relevant news nodes to encode the overall semantic information of \( G_n \). In the attention module, we regard the root node embedding \( h_{n,0} \) as the query and the semantic-relevant news node embeddings \( \{h_{n,i}\}_{i=1}^N \) as the key-value pairs:

\[
e_i = \frac{(h_{n,0}W^Q_n)(h_{n,i}W^K_n)^T}{\sqrt{d}}
\]

\[
\alpha_i = \text{softmax}(e_i) = \frac{\exp(e_i)}{\sum_{j=1}^N \exp(e_j)}
\]

\[
h^G_n = \sum_{i=1}^N \alpha_i h_{n,i} \tag{3}
\]

, where \( W^Q_n \in \mathbb{R}^{d \times d} \) and \( W^K_n \in \mathbb{R}^{d \times d} \) are parameter matrices. We integrate the local and global graph contexts by a simple feed-forward gating network \( \text{FFN}_g(\cdot) \) to derive the news graph context \( c_n \):

\[
c_n = \text{FFN}_g([h^L_n; h^G_n]) \tag{4}
\]

---

\(^3\)Specifically, we use pretrained mpnet-base-v2 (Song et al., 2020) in the Sentence Transformers library https://www.sbert.net/docs/pretrained_models.html as the news retrieval PLM.
The parameters of news graph context extractor are shared among different graph layers of DIGAT (the user graph context extractor in Section 3.3.2 also shares parameters likewise).

3.3 User Graph Encoding Channel

3.3.1 User Graph Encoding Construction

Motivated by Mao et al. (2021) and Wu et al. (2021), we model user history with graph structure to encode multi-levels of user interests. We build a user graph \( G_u \) containing news nodes and topic nodes: (1) For a user’s clicked news \( H_u = [n_1, n_2, ..., n|H_u|] \), we treat it as a set of news nodes for news-level user interest representation. (2) For the clicked news \( n_j \), it is pertaining to a specific news topic \( t(i) \). We treat the clicked news topics as topic nodes for topic-level user interest representation. To capture the interaction among news and topics, we introduce three types of edges:

- **News-News Edge.** News nodes with the same topic category (e.g., *Sports*) are fully connected. In this way, we can capture the relatedness among clicked news with news-level interaction.
- **News-Topic Edge.** We model the interaction between clicked news and topics by connecting news nodes to their pertaining topic nodes.
- **Topic-Topic Edge.** Topic nodes are fully connected. In this way, we can capture the overall user interests with topic-level interaction.

3.3.2 User Graph Context Extraction

Given the user history \( H_u = [n_1, n_2, ..., n|H_u|] \), we employ the semantic news encoder (introduced in Section 3.1) to learn the historical news embeddings \( h_u = [h_{n,1}^u, h_{n,2}^u, ..., h_{n,|H_u|}^u] \in \mathbb{R}^{H_u \times d} \). Given \( |t(.)| \) topics indicated by the clicked news, the topic nodes are embedded into learnable embeddings \( h_t = [h_{t,1}^u, h_{t,2}^u, ..., h_{t,|t(.)|}] \in \mathbb{R}^{t(.) \times d} \).

The user node embeddings are \( h_u = [h_{n,1}^u, h_{t,1}^u] \).

Motivated by Qi et al. (2021c), we extract the graph context \( c_u \in \mathbb{R}^d \) in a hierarchical way. First, we employ an attention module to learn the topic representation \( \tilde{h}_{t(i)} \in \mathbb{R}^d \) of the topic \( t(i) \). The topic-attention module regards the news graph context \( c_u \) as the query and the news embeddings \( \{h_{n,j}\}_{n \in t(i)} \) of topic \( t(i) \) as the key-value pairs:

\[
\tilde{h}_{t(i)} = \text{Attn}(Q=c_u, K=\{h_{n,j}\}_{n \in t(i)}, V=\{h_{n,j}\}) \tag{5}
\]

Then, we employ another attention module to extract the user graph context \( c_u \in \mathbb{R}^d \). The user-attention module regards the news graph context \( c_n \) as the query and the learned topic representations \( \{\tilde{h}_{t(i)}\}_{i=1}^{|t(.)|} \) as the key-value pairs:

\[
c_u = \text{Attn}(Q=c_n, K=\{\tilde{h}_{t(i)}\}, V=\{\tilde{h}_{t(i)}\}) \tag{6}
\]

\( \text{Attn}(Q, K, V) \) in Eq. (5) and (6) denotes the standard attention module with Query/Key/Value. We implement \( \text{Attn}(Q, K, V) \) as scaled dot-product attention (Vaswani et al., 2017) in our experiments.

3.4 Dual-Graph Interaction

In news graph \( G_n \), node embeddings \( \{h_{n,i}\}_{i=0}^{|G_n|} \) contain the information of augmented candidate news semantics. In user graph \( G_u \), node embeddings \( \{h_{u,i}\}_{i=0}^{|G_u|} \) contain the information of user history. We learn informative news and user graph embeddings by aggregating neighboring node information with stacked graph attention layers (Veličković et al., 2018). Most notably, our dual-graph interaction model aims at facilitating effective feature interaction between the news and user graphs. By effective dual-graph feature interaction, accurate news-user representation matching can be achieved. In the dual-graph interaction, the \((l+1)\)-layer news node embeddings \( h_{n,(l+1)} \) is updated based on the \(l\)-layer news node embeddings \( h_{n,l} \) and user graph context \( c_u^{(l)} \), jointly (vice versa to update the user node embeddings \( h_{u,(l+1)} \), as illustrated in Figure 2.

We illustrate the news node embedding update process for example. We first perform a linear transformation on the \(l\)-layer news node embedding \( h_{n,l}^{(l)} \).
to derive higher-level graph features $\hat{h}_{n,i}$:

$$\hat{h}_{n,i} = \mathbf{W}_n^l \hat{h}_{n,i}^{(l)} + \mathbf{b}_n^l,$$

(7)

where $\mathbf{W}_n^l \in \mathbb{R}^{d \times d}$ and $\mathbf{b}_n^l \in \mathbb{R}^d$ are learnable.

In order to learn news node embeddings interacting with user graph, we incorporate the user graph context $c_u^{(l)}$ into news graph attention computation. For news node $i$ and node $j \in \mathcal{N}_i^n$ (where $\mathcal{N}_i^n$ is the neighborhood of node $i$), we incorporate user graph context $c_u^{(l)}$ into computing the attention key vector $K_{i,j}$. We use a feed-forward network $\text{FFN}_n^{(l)}$ to compute $K_{i,j}$ based on the fused information of $c_u^{(l)}$, $h_{n,i}^{(l)}$ and $h_{n,j}^{(l)}$. The news graph attention coefficient $\alpha_{i,j}$ is computed aware of user graph context:

$$K_{i,j} = \text{FFN}_n^{(l)} \left( \left[ c_u^{(l)}; h_{n,i}^{(l)}; h_{n,j}^{(l)} \right] \right)$$

(8)

$$\alpha_{i,j} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}_n^T K_{i,j}))}{\sum_{k \in \mathcal{N}_i^n} \exp(\text{LeakyReLU}(\mathbf{a}_n^T K_{i,k}))}$$

(9)

where $\mathbf{a}_n^T$ is a learnable attention weight vector. Finally, we aggregate the neighboring node embeddings with attention coefficient $\alpha_{i,j}$, followed by ReLU activation. Residual connection is applied to mitigate gradient vanishing in deep graph layers:

$$h_{n,i}^{(l+1)} = \text{ReLU} \left( \sum_{j \in \mathcal{N}_i^n} \alpha_{i,j} \hat{h}_{n,j}^{(l)} \right) + h_{n,i}^{(l)}$$

(10)

The news and user graph contexts $c_u^{(l)}$ and $c_u^{(l)}$ are extracted from the $l$-layer graph node embeddings as described in Section 3.2.2 and 3.3.2. We summarize Eq. (7) to (10) as the news node embedding update function $\Phi_n^{(l)}$:

$$h_{n,i}^{(l+1)} = \Phi_n^{(l)} \left( c_u^{(l)}; h_{n,i}^{(l)}; \{ h_{n,j}^{(l)} \}_{j \in \mathcal{N}_i^n} \right)$$

(11)

Similarly, the update function of user node embeddings is formulated as $\Phi_u^{(l)}$:

$$h_{u,i}^{(l+1)} = \Phi_u^{(l)} \left( c_u^{(l)}; h_{u,i}^{(l)}; \{ h_{u,j}^{(l)} \}_{j \in \mathcal{N}_i^n} \right)$$

(12)

The dual-graph interaction can be viewed as an iterative process that performs (1) user graph context-aware attention to update news node embeddings and (2) news graph context-aware attention to update user node embeddings. We model the dual interaction with $L$ stacked layers. The final layers of news and user graph contexts $c_u^{(l)}$ and $c_u^{(l)}$ are adopted as news and user graph representations $r_n$ and $r_u$ which refine the news and user graph information with deep feature interaction. Algorithm 1 illustrates the dual-graph interaction process.

### 3.5 Click Prediction and Model Training

With the news and user graph representations $r_n$ and $r_u$, our model aims to predict the matching score $\hat{s}_{n,u}$ which signals how likely user $u$ will click news $n$. The matching score between news and user representations is simply computed by dot product as $\hat{s}_{n,u} = r_n^T r_u$.

Following Wu et al. (2019a,b,d), we adopt negative sampling strategy to train our model. For the user behavior that user $u$ had clicked news $n_i$, we compute the click matching score as $\hat{s}_{i,u}$ for $n_i$ and $u$. Besides, we randomly sample $S$ non-clicked news $[n_1, n_2, ..., n_S]$ from the user’s behavior log and compute the negative matching scores as $[\hat{s}_{1,u}, \hat{s}_{2,u}, ..., \hat{s}_{S,u}]$. We optimize the NCE loss $\mathcal{L}$ over the training dataset $\mathcal{D}$ in model training:

$$\mathcal{L} = - \sum_{i=1}^{|\mathcal{D}|} \log \frac{\exp(\hat{s}_{i,u})}{\exp(\hat{s}_{i,u}) + \sum_{j=1}^{S} \exp(\hat{s}_{j,u})}$$

(13)

### 4 Experiments

#### 4.1 Dataset and Experiment Settings

We conduct experiments on the real-world benchmark dataset MIND (Wu et al., 2020). MIND is constructed from anonymized user behavior logs of Microsoft News with two versions of MIND-large and MIND-small. MIND-large contains 1 million anonymized users with user-click impression logs of 6 weeks from October 12 to November 22, 2019. The training and dev sets contain the impression logs of the first 5 weeks, and the last week’s impression logs are reserved for test. MIND-small
consists of 50000 users, which are randomly sampled from MIND-large with the impression logs.

Following previous works (Wang et al., 2020; Qi et al., 2021c), we use news titles with the maximum length of 32 words for news textual encoding. The user history includes 50 news items they have recently clicked. The news word embeddings are 300-dimensional and initialized from the pretrained Glove embeddings (Pennington et al., 2014). Following An et al. (2019), we set the number of negative news samples $S$ to be 4. For our model parameters, the news representation dimension $d$ is set as 400 for fair comparison to baselines. The number of neighboring nodes $M$ and hops $K$ are 5 and 2, respectively. We set the number of dual-graph interaction layers as $L = 3$. We use Adam optimizer (Kingma and Ba, 2015) with the learning rate of 1e-4 to train our model. Following Wu et al. (2020), we employ the recommendation ranking metrics AUC, MRR, nDCG@5, and nDCG@10 to evaluate model performance.

### 4.2 Compared Methods

We compare our model with the state-of-the-art news recommendation methods: (1) **GRU** (Okura et al., 2017), learning user representations from a sequence of clicked news with a GRU network; (2) **DKN** (Wang et al., 2018), using a knowledge-aware CNN to learn news representations from both news texts and knowledge entities; (3) **NPA** (Wu et al., 2019b), encoding news and user representations with personalized attention networks; (4) **NAML** (Wu et al., 2019a), learning news representations from news titles, bodies, categories and subcategories with multi-view attention networks; (5) **LSTUR** (An et al., 2019), jointly modeling long-term user embeddings and short-term user interests learned by a GRU network; (6) **NRMS** (Wu et al., 2019d), encoding informative news and user representations with multihead self-attention networks; (7) **FIM** (Wang et al., 2020), encoding news content with dilated convolutional networks and modeling user interest matching with 3D convolutional networks; (8) **HierRec** (Qi et al., 2021c), modeling user interests in a three-level hierarchy and performing multi-grained matching between candidate news and hierarchical user interest representations.

We also compare our model with competitive graph-based methods: (9) **GERL** (Ge et al., 2020), modeling the news-user relatedness with a bipartite graph, which enhances news and user representations by aggregating neighboring node information; (10) **GNewsRec** (Hu et al., 2020a), using graph neural networks (GNN) (Hamilton et al., 2017) and attentive LSTMs to jointly model users’ long-term and short-term interests; (11) **User-as-Graph** (Wu et al., 2021), utilizing a heterogeneous graph pooling method to extract user representations from personalized heterogeneous behavior graphs.

### 4.3 Main Experiment Results

Table 1 presents the main experiment results. We can observe that DIGAT significantly outperforms previous SOTA methods (i.e., methods #1 to #8) on the both datasets. This is because even though some baselines use topic categories or knowledge entities to enrich news information (e.g., HierRec learns news representations from both news texts...
and knowledge entities), the information entailed in single candidate news may be still insufficient. In contrast, DIGAT can substantially enrich the semantic information of the single candidate news by SAG modeling, which provides more accurate signals of candidate news to match user interests. Besides, DIGAT consistently outperforms three graph-based baselines. We find that GERL is hard to model news-user interaction in test data, because most candidate news items in test data are fresh and have no click-interaction with users. Differently, DIGAT models news and users with dual graph channels instead of a joint bipartite graph, which circumvents this cold-news issue. Compared to GNewsRec and User-as-Graph, DIGAT performs more effective feature interaction between the news and user graphs, which can enhance more accurate news-user representation matching.

4.4 Ablation Study on SAG Modeling

We examine the effectiveness of SAG modeling with three ablation experiments: (1) w/o SA. To examine the effectiveness of semantic-augmentation (SA) strategy, we remove SAG from DIGAT and learn single candidate news representation instead. (2) TF-IDF SA. To inspect the function of the news retrieval PLM $\phi(\cdot)$ in SAG construction (see Section 3.2.1), we replace $\phi(\cdot)$ with a TF-IDF syntactic feature extractor to retrieve relevant news. (3) Seq SA. To examine the effectiveness of graph-based SA, we conduct controlled experiments by arranging the semantic-relevant news in a sequential form and extracting the news sequence context similar to Eq. (3) and (4). Experiments in this section and the following sections are on MIND-small.

Table 2 shows the experiment results. We can see that abandoning the SA strategy (w/o SA) leads to the largest performance drop, as TF-IDF SA and Seq SA also yield better performance than w/o SA. This validates the effectiveness of SA strategy to enrich candidate news semantics and further enhance news recommendation. TF-IDF SA underperforms DIGAT by a considerable margin. We infer that the TF-IDF features can only measure news similarity at the syntactic level, which may not be able to accurately retrieve semantic-relevant news for SAG construction. In contrast, PLM can accurately measure news similarity at the semantic level and help retrieve more relevant news to enhance SAG modeling. It reveals that accurately retrieving semantically relevant news is the key to candidate news semantic-augmentation. Besides, Seq SA is suboptimal compared to the original graph-based SA. This is because the graph-based SA method can accurately model the relatedness among the candidate news and semantic-relevant news with multi-neighbor and multi-hop graph structure, which further improves the effectiveness of the SA strategy.

4.5 Ablation Study on Graph Interaction

To examine the effectiveness of dual-graph interaction, we design the following ablation experiments: (1) w/o Interaction. We employ the vanilla graph attention networks (GAT) (Veličković et al., 2018) to learn news and user graph embeddings, respectively, without interaction between dual graphs. (2) News Graph w/o Inter. The news graph embedding update layers are replaced with vanilla GAT layers. Concretely, Eq. (11) is modified into $h_{n,i}^{(l+1)} = \tilde{\Phi}_n^{(l)}(h_{n,i}^{(l)}, \{h_{n,j}^{(l)}\}_{j \in \mathcal{N}(n)})$, where $\tilde{\Phi}^{(l)}$ is the standard GAT graph embedding update function without feature interaction with user graph context. (3) User Graph w/o Inter. Similar to (2), we replace the user graph embedding update layers with vanilla GAT layers.

Figure 3 shows the performance of the ablation models. We can see that w/o Interaction underperforms the other three models with graph interaction modeling. It indicates that feature interaction between candidate news and users is necessary to enhance news recommendation. We also observe
that removing user graph interaction (User Graph w/o Inter) leads to more performance drop than News Graph w/o Inter, which implies that user graph interaction may contribute more to our model. Moreover, DIGAT surpasses the two single graph interaction ablations by a significant margin, validating the effectiveness of modeling dual-graph feature interaction in an iterative manner.

4.6 Analysis on SAG Parameters

We investigate two key parameters of SAG, i.e., the number of node neighbors $M$ and hops $K$. Figure 4 shows the effect of different $M$ and $K$ settings.

As shown in Figure 4(a), DIGAT performance continues rising as $M$ increases from 1 to 5. This indicates that with more semantic-relevant news incorporated, SAG can leverage more sufficient semantic information to augment the candidate news representations. On the other hand, the model performance slightly declines as $M > 5$. The reason could be twofold. First, as the scale of SAG grows larger, it becomes more challenging for the model to distill the global graph context of SAG (see Section 3.2.2). Second, as $M$ becomes too large, it is inevitable to retrieve more noisy news in the SAG construction process, which may adversely affect SAG modeling. From Figure 4(b), we observe that $K = 2$ is the optimal hop setting. This may be because two hops of SAG can heuristically capture more useful semantic-relevant news information than simple one-hop modeling, while higher-hop extension may introduce too much irrelevant news and interfere with accurate semantic augmentation for candidate news. In general, we select $M = 5$ and $K = 2$ for SAG construction\(^5\).

\(^5\)The SAG construction on MIND-large can be preprocessed in 20 minutes on Intel Xeon(R) Gold 6226R CPU @ 2.90 GHz with Nvidia RTX 3090 GPU before model training.

4.7 The Number of Dual-Graph Layers

We study the effect of the number of dual-graph layers $L$ in DIGAT. The results are presented in Figure 5. We can see that the model performance first keeps rising when $L$ increases from 1 to 3. It suggests that deep feature interaction between news and user graphs is useful to improve recommendation performance, as it can model the news and user representation matching process in a more fine-grained way. We also observe that further increasing $L$ hurts the model performance. It may be caused by the unstable gradient in training the deep dual-graph architecture, as we empirically find that gradient clipping (Pascanu et al., 2013) is indispensable to avoid loss diverging in DIGAT training, in cases when the dual-graph layers become too deep (i.e., $L \geq 6$).

5 Conclusion

In this work, we present a dual-graph interaction framework for news recommendation. In our approach, a graph enhanced semantic-augmentation strategy is employed to enrich the semantic information of candidate news. Moreover, we design a dual-graph interaction mechanism to achieve ef-
fective feature interaction between news and user graphs, facilitating more accurate news and user representation matching. Our approach advances the state-of-the-art news recommendation methods on the benchmark dataset MIND. Extensive experiments and further analyses validate that SAG modeling and dual-graph interaction can effectively improve news recommendation performance.

6 Limitations

In this section, we discuss the limitations of our approach. First, since DIGAT models dual-interaction between news and user graph features iteratively, the inference efficiency is a concern. We compare the model size and inference run-time of experimental methods in Table 3. The news representations (see Section 3.1) of all methods, except NPA6, are pre-computed and cached for fast inference. As DIGAT is scalable with the dual-graph depth \( L \), we also evaluate DIGAT on \( L = 1 \) and 2.

In terms of model size, DIGAT is larger than the first eight models in Table 3. Compared to DIGAT \((L = 1, 2)\), we can see that the parameter growth comes from the stacked graph layers. We also find that embedding layers contain considerable parameters7, while DIGAT does not need additional news and user ID embedding layers. In terms of inference time, DIGAT runs slower than other models. We find that the computational overhead mostly comes from the iterative graph embedding update process in Eq. (11) and (12). Nonetheless, this efficiency issue can be alleviated. Since DIGAT is scalable with the dual-graph depth \( L \), the trade-off between recommendation accuracy and efficiency can be made. We can scale down the dual-graph layers \( L \) to reduce the model size and inference time with compromising performance. As shown in Table 3, when the dual-graph layers turn down to \( L = 1 \), the performance of DIGAT is also superior to baseline methods, while the parameter size and inference time are comparable to several baselines (e.g., FIM and GNewsRec). In industrial deployment, this trade-off can depend on specific requirements of computational resources.

Second, our approach is evaluated on the offline experimental dataset. For online recommender services, searching and retrieving real-time relev-

---

6This is because the personalized user ID embeddings are premised in computing NPA news representations.

7For example, GERL needs additional embedding matrices to embed news and user IDs, NAML needs larger word embedding matrices to encode news titles and bodies together.

| Method       | Param. (MB) | Run-time (s) | AUC  |
|--------------|-------------|--------------|------|
| GRU          | 19.9        | 92.0         | 61.51|
| DKN          | 23.3        | 80.5         | 62.90|
| NPA          | 30.1        | 376.6        | 64.65|
| NAML         | 39.3        | 80.1         | 66.12|
| LSTUR        | 35.3        | 98.9         | 65.87|
| NRMS         | 22.2        | 89.5         | 65.63|
| FIM          | 22.5        | 495.7        | 65.34|
| HieRec       | 31.7        | 106.9        | 67.83|
| GERL         | 75.1        | 129.1        | 65.27|
| GNewsRec     | 50.6        | 186.8        | 65.54|
| DIGAT \((L = 1)\) | 40.3        | 598.5        | 68.77|
| DIGAT \((L = 2)\) | 35.4        | 426.3        | 68.60|
| DIGAT \((L = 1)\) | 30.5        | 250.7        | 68.23|

Table 3: Comparison of experimental methods’ parameters and inference run-time. The Run-time column denotes the inference time on MIND-small test set, which is averaged by 10 times. All models are tested with the same batch size on Nvidia RTX 3090.

vant news by event-driven news clustering models (Saravanakumar et al., 2021) to construct SAG is a more promising option than the static retrieval method. To this end, we will explore applying our approach to online applications in future work.

Acknowledgements

We appreciate constructive comments from some anonymous reviewers. The research described in this paper is supported by Innovation & Technology Commission HKSAR, under ITF Project No. PRP/054/21FX.

References

M. Tarik Altuncu, Sophia N. Yaliraki, and Mauricio Barahona. 2018. Content-driven, unsupervised clustering of news articles through multiscale graph partitioning. In arXiv preprint arXiv:1808.01175.

Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. 2019. Neural news recommendation with long- and short-term user representations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 336–345, Florence, Italy. Association for Computational Linguistics.

Lei Chen, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2020. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, USA, February 7-12, 2020, pages 27–34. AAAI Press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of
deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pages 4171–4186. Association for Computational Linguistics.

Suyu Ge, Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020. Graph enhanced representation learning for news recommendation. In WWW ’20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, pages 2863–2869. ACM / IW3C2.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In International Conference on Learning Representations.

Linmei Hu, Chen Li, Chuan Shi, Cheng Yang, and Chao Shao. 2020a. Graph neural news recommendation with long-term and short-term interest modeling. Information Processing & Management, 57:102142.

Linmei Hu, Siyong Xu, Chen Li, Cheng Yang, Chuan Shi, Nan Duan, Xing Xie, and Ming Zhou. 2020b. Graph neural news recommendation with unsupervised preference disentanglement. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4235–4264, Online. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In Advances in Neural Information Processing Systems, volume 33, pages 9459–9474. Curran Associates, Inc.

Jian Li, Jieming Zhu, Qiwei Bi, Guohao Cai, Lifeng Shang, Zhenhua Dong, Xin Jiang, and Qun Liu. 2022. MINER: Multi-interest matching network for news recommendation. In Findings of the Association for Computational Linguistics: ACL 2022, pages 343–352. Dublin, Ireland. Association for Computational Linguistics.

Danyang Liu, Jianxun Lian, Shiyin Wang, Ying Qiao, Jien-Hung Chen, Guangzhong Sun, and Xing Xie. 2020. Kred: Knowledge-aware document representation for news recommendations. In Fourteenth ACM Conference on Recommender Systems, RecSys ’20, page 200–209, New York, NY, USA. Association for Computing Machinery.

Zhiming Mao, Xingshan Zeng, and Kam-Fai Wong. 2021. Neural news recommendation with collaborative news encoding and structural user encoding. In Findings of the Association for Computational Linguistics: EMNLP 2021, pages 46–55, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, page 1933–1942, New York, NY, USA. Association for Computing Machinery.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In Proceedings of the 30th International Conference on Machine Learning, volume 28 of Proceedings of Machine Learning Research, pages 1310–1318, Atlanta, Georgia, USA. PMLR.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543.

Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2021a. Personalized news recommendation with knowledge-aware interactive matching. In SIGIR ’21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, pages 61–70. ACM.

Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2021b. PP-rec: News recommendation with personalized user interest and time-aware news popularity. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 5457–5467, Online. Association for Computational Linguistics.

Tao Qi, Fangzhao Wu, Chuhan Wu, Peiru Yang, Yang Yu, Xing Xie, and Yongfeng Huang. 2021c. HieRec: Hierarchical user interest modeling for personalized news recommendation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 5446–5456, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 57–64.
Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4512–4525, Online. Association for Computational Linguistics.

Kailash Karthik Saravanakumar, Miguel Ballesteros, Muthu Kumar Chandrasekaran, and Kathleen McKeeown. 2021. Event-driven news stream clustering using entity-aware contextual embeddings. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 2330–2340, Online. Association for Computational Linguistics.

Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019c. Neural news recommendation with topic-aware news representation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1154–1159, Florence, Italy. Association for Computational Linguistics.

Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019d. Neural news recommendation with multi-head self-attention. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 6389–6394, Hong Kong, China. Association for Computational Linguistics.

Chuhan Wu, Fangzhao Wu, Yongfeng Huang, and Xing Xie. 2021. User-as-graph: User modeling with heterogeneous graph pooling for news recommendation. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pages 1624–1630. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Jingwei Yi, Fangzhao Wu, Chuhan Wu, Huizhao Liu, Guangzhong Sun, and Xing Xie. 2021. EfficientFedRec: Efficient federated learning framework for privacy-preserving news recommendation. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qiannan Zhu, Xiaofei Zhou, Zeliang Song, Jianlong Tan, and Li Guo. 2019. Dan: Deep attention neural network for news recommendation. Proceedings of the AAAI Conference on Artificial Intelligence, 33(01):5973–5980.
Algorithm 2 SAG Construction Procedure

Input: candidate news \( n_0 \), news corpus \( \{ N_C \} \), news node neighbors \( M \) and hops \( K \).
Output: semantic-augmented graph \( G_n \).
1: \( n_0 \) as the root node \( v_0 \) of SAG.
2: Initialize the node set \( V ← \{v_0\} \) and edge set \( E ← \{\} \).
Define parent node set \( P ← \{v_0\} \) and node-hop counter \( hop[v_0] = 0 \).

Graph extension process:
3: while \( P ≠ \emptyset \) do
4: \( v_i \) from \( P \), then \( P = P \setminus \{v_i\} \)
5: \( M \) news \( \{n_j\}_{j=1}^M \) from the news corpus \( \{ N_C \} \) with the \( M \) highest semantic similarity scores \( \{ s_{i,j} \}_{j=1}^M \) as nodes \( \{ v_j \}_{j=1}^M \)
6: for \( j = 1, 2, ..., M \) do
7: if \( v_j \) do not belong to \( V \) then
8: \( V = V \cup \{v_j\} \)
9: \( hop[v_j] = hop[v_i] + 1 \)
10: if \( hop[v_j] \lt K \) then
11: \( P = P \cup \{v_j\} \)
12: end if
13: end if
14: if \( v_i \) is not connected to \( v_j \) then
15: \( E = E \cup \{v_i, v_j\} \)
16: end if
17: end for
18: end while
19: \( G_n = (V, E) \).
20: return \( G_n \).

A Semantic-augmented Graph Construction and Qualitative Analysis

Algorithm 2 illustrates the procedure of semantic-augmented graph (SAG) construction. First of all, the SAG \( G_n \) is initialized from the root node \( v_0 \) which represents the original candidate news \( n_0 \).

The graph construction is performed by repeatedly extending semantic-relevant neighboring news nodes to existing nodes in \( G_n \). In the graph extension process (line 3 to 18 in Algorithm 2), for an existing node \( v_i \) (corresponding to news \( n_i \)) in \( G_n \), we retrieve \( M \) news documents \( \{ n_j \}_{j=1}^M \) from the news corpus \( \{ N_C \} \) with the \( M \) highest similarity scores \( \{ s_{i,j} \}_{j=1}^M \). If the similarity score \( s_{i,j} \) of node \( n_i \) and \( n_j \) is evaluated by a PLM \( \phi(\cdot) \) with Eq. (2), we treat the retrieved news \( \{ n_j \}_{j=1}^M \) as news nodes \( \{ v_j \}_{j=1}^M \). For each node \( v_j \), we extend it to \( G_n \) as a neighboring node of \( v_i \) by adding bidirectional edge \( e_{v_i, v_j} \) between \( v_i \) and \( v_j \). To heuristically explore higher-order semantic-relevant news, news nodes in SAG are extended from the root node \( v_0 \) within \( K \) hops at most.

SAG Example. Figure 6 demonstrates an exam-ple of SAG for the candidate news \( n_0 \) “Should the NFL be able to fine players for criticizing officiating?”. Interestingly, from Figure 6(b), we can see that there are many similar news articles in SAG, which refer to the same specific news event or person (i.e., “NFL” and “fine players”) from different narrative points of view. These semantic-relevant news articles are finely retrieved with the help of PLM retriever, forming explicit multi-neighbor and multi-hop graph structure. With the representation power of SAG, DIGAT can learn more accurate relatedness of the relevant news texts and substantially enrich the semantic information of the original candidate news \( n_0 \).

News Clustering Phenomenon. From the SAG example shown in Figure 6(a), we can observe that there exist many cyclic subgraphs (i.e., news clusters), revealing the news clustering phenomenon in semantic space. These cyclic graph structures depict the similar news clusters in real-world distributions, consistent with the previous research. This news semantic clustering phenomenon also inspires the motivation of our work.

Broader Impact. On online news platforms, the Related News is usually displayed along with the original news to users. It is worth mentioning that such Related News on news platforms is practically retrieved from the news database by retrieval models in industrial practice (Algorithm 2 can be seen as such an analogous retrieval process). As an alternative, we can also use the off-the-shelf real-time Related News on online news platforms to construct SAG. Furthermore, the SAG modeling strategy is also applicable to other text-based recommendation (e.g., Twitter Feed Recommendation). We will explore this direction in future work.

B Supplementary Experiments on Semantic-Augmentation Strategy

We conduct supplementary experiments to investigate whether the semantic-augmentation (SA) strategy can be generalized for news recommendation task. To exclude the influence of DIGAT itself, we choose to reinforce the baseline NRMS (Wu et al., 2019d) with SA strategy, named NRMS-SA. For NRMS-SA, we use the PLM news retriever to retrieve 10 semantic-relevant news articles for each
Table 4: Supplementary experiments on NRMS reinforced with semantic-augmentation (SA) strategy.

| Method            | MIND-small       | MIND-large       |
|-------------------|------------------|------------------|
|                   | AUC   | MRR   | nDCG@5 | nDCG@10 | AUC   | MRR   | nDCG@5 | nDCG@10 |
| NRMS              | 65.63 | 30.96 | 34.13  | 40.52   | 67.66 | 33.25 | 36.28  | 42.13   |
| NRMS-SA           | 67.27 | 32.37 | 35.84  | 42.13   | 69.31 | 34.39 | 37.56  | 43.27   |
| HieRec            | 67.83 | 32.78 | 36.31  | 42.49   | 69.03 | 33.89 | 37.08  | 43.01   |
| User-as-Graph     | 69.23 | 34.14 | 37.21  | 43.04   |        |       |        |         |

Table 4 shows the experiment results, which indicate that semantic-augmentation strategy can also be applied to other news recommendation models and achieve substantial performance improvement. Interestingly, we find that the improvement on MIND-large is more significant than on MIND-small, as NRMS-SA is even on par with the previous SOTA baseline (i.e., User-as-graph). We infer that it may be because the MIND-large news corpus is an order of magnitude larger than MIND-small, and hence it contains more semantic-relevant news for SAG modeling. The experiment results also suggest that augmenting the semantic representation of single candidate news by relevant news information sources is a promising direction to improve news recommendation performance.