Information Processing and Management 57 (2020) 102142

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Graph neural news recommendation with long-term and short-term interest modeling

Linmei Hu[a], Chen Li[a], Chuan Shi[*,a], Cheng Yang[a], Chao Shao[b]

[a] Beijing University of Posts and Telecommunications, No 10, Xitucheng Road, Haidian District, Beijing, PRC, 100876 China
[b] Alibaba Group, Hangzhou, China

ARTICLE INFO

Keywords:
News recommendation
Graph neural networks
Long-term interest
Short-term interest

ABSTRACT

With the information explosion of news articles, personalized news recommendation has become important for users to quickly find news that they are interested in. Existing methods on news recommendation mainly include collaborative filtering methods which rely on direct user-item interactions and content based methods which characterize the content of user reading history. Although these methods have achieved good performances, they still suffer from data sparse problem, since most of them fail to extensively exploit high-order structure information (similar users tend to read similar news articles) in news recommendation systems. In this paper, we propose to build a heterogeneous graph to explicitly model the interactions among users, news and latent topics. The incorporated topic information would help indicate a user's interest and alleviate the sparsity of user-item interactions. Then we take advantage of graph neural networks to learn user and news representations that encode high-order structure information by propagating embeddings over the graph. The learned user embeddings with complete historic user clicks capture the users' long-term interests. We also consider a user's short-term interest using the recent reading history with an attention based LSTM model. Experimental results on real-world datasets show that our proposed model significantly outperforms state-of-the-art methods on news recommendation.

1. Introduction

As the amount of online news platforms such as Yahoo! news[1] and Google news[2] increases, users are overwhelmed with a large volume of news from the worldwide covering various topics. To alleviate the information overloading, it is critical to help users target their reading interests and make personalized recommendations Bansal, Das, and Bhattacharyya (2015); Li, Chu, Langford, and Schapire (2010); Liu, Dolan, and Pedersen (2010); Phelan, McCarthy, and Smyth (2009). Therefore, news recommender systems that automatically recommend a small set of news articles for satisfying users preferences, have growingly attracted attentions in both industry and academic Das, Datar, Garg, and Rajaram (2007); Wang, Zhang, Xie, and Guo (2018); Wang et al. (2017).

There is a wide variety of typical methods to make personalized news recommendations, including collaborative filtering (CF) methods Das et al. (2007); Wang and Blei (2011) and content based methods Huang et al. (2013); IJntema, Goossen, Frasincar, and

* Corresponding author.
E-mail addresses: hulinmei@bupt.edu.cn (L. Hu), leechen@bupt.edu.cn (C. Li), shichuan@bupt.edu.cn (C. Shi), albertyang33@gmail.com (C. Yang), shaochao.sc@alibabainc.com (C. Shao).

[1] https://news.yahoo.com/
[2] https://news.google.com/

https://doi.org/10.1016/j.ipm.2019.102142
Received 29 July 2019; Received in revised form 20 September 2019; Accepted 7 October 2019
Available online 15 November 2019
0306-4573/

Hogenboom (2010); Wang et al. (2018); Zhu, Zhou, Song, Tan, and Guo (2019). CF methods based on IDs always suffer from the cold start problem since out-of-date news are substituted by newer ones frequently. While content based methods completely ignore the collaborative signal. Hybrid methods combing CF and content for news recommendation have been proposed to address the problems De Francisci Morales, Gionis, and Lucchese (2012); Li, Wang, Li, Knox, and Padmanabhan (2011). However, all these methods still suffer from the data sparsity problem, since they fail to extensively exploit high-order structure information (e.g., the $u_1 - d_1 - u_2$ relationship indicates the behavior similarity between the users u_1 and u_2). In addition, most of them ignore the latent topic information which would help indicate a users interest and alleviate the sparse user-item interactions. The intuition is that news items with few user clicks can aggregate more information with the bridge of topics. What's more, the existing methods on news recommendation do not consider the user's long-term and short-term interests. A user usually has relatively stable long-term interests and may also be temporally attracted to certain things, i.e., short-term interests, which should be considered in news recommendation. For example, a user may continuously concern about political events, which is a long-term interest. In contrast, certain breaking news events such as attacks usually attract temporary interests.

To address the above issues, in this paper, we propose a novel ***G***raph Neural ***News Rec***ommendation model (***GNewsRec***) with long-term and short-term user interest modeling. We first construct a heterogeneous user-news-topic graph as shown in Fig. 2 to explicitly model the interactions among users, news and topics with complete historic user clicks. The topic information can help better reflect a user's interest and alleviate the sparsity issue of user-item interactions. To encode the high-order relationships among users, news items and topics, we take advantage of graph neural networks (GNN) to learn user and news representations by propagating embeddings over the graph. The learned user embeddings with complete historic user clicks are supposed to encode a user's long-term interest. We also model a user's short-term interest using recent user reading history with an attention based LSTM Hochreiter and Schmidhuber (1997); Liu et al. (2018) model. We combine both long-term and short-term interests for user modeling, which are then compared to the candidate news representation for prediction. Experimental results on real-world datasets show that our model significantly outperforms state-of-the-art methods on news recommendation.

Our main contributions can be summarized as follows:

1) In this work, we propose a novel graph neural news recommendation model GNewsRec with long-term and short-term user interest modeling.
2) Our model constructs a heterogeneous user-news-topic graph to model user-item interactions, which alleviates the sparsity of user-item interactions. Then it applies graph convolutional networks to learn user and news embeddings with high-order information encoded by propagating embeddings over the graph.
3) Experimental results on real-world datasets demonstrate that our proposed model significantly outperforms sate-of-the-art methods on news recommendation.

2. Related work

Personalized news recommendation has been widely studied with the information overloading of news articles. A variety of methods have been proposed including collaborative filtering (CF) based methods Das et al. (2007); Li, Kawale, and Fu (2015); Marlin and Zemel (2004); Rendle (2010); Wu, DuBois, Zheng, and Ester (2016); Xue, Dai, Zhang, Huang, and Chen (2017), and content based methods Huang et al. (2013); IJntema et al. (2010); Kompan and Bieliková (2010).

CF methods assume that behaviorally similar users would exhibit similar preference on items. They parameterize users and items for reconstructing historical interactions, and predict user preferences based on the parameters Cao, Wang, He, Hu, and Chua (2019); He et al. (2017); Wang, He, Wang, Feng, and Chua (2019c). For example, matrix factorization (MF) directly embeds user/item ID as vectors and models user-item interaction with inner product. DMF Xue et al. (2017) is a deep matrix factorization model which uses multiple nonlinear layers to process both explicit ratings and implicit feedback of users and news. However, most existing CF-based methods build the user and item embeddings with the descriptive features only (e.g., ID and attributes), without considering the higher-order information within the user-item interaction graph. Wang et al. (2019c) proposed a neural graph CF method which exploits the user-item graph structure by propagating embeddings over the graph. While CF methods still suffer from the cold-start problem since news items are substituted frequently.

To address this issue, content-based or hybrid methods have been proposed Cheng et al. (2016); Guo, Tang, Ye, Li, and He (2017); Huang et al. (2013); Wang et al. (2018); Zhang, Liu, and Gulla (2018); Zhu et al. (2019). Content based methods consider the actual content or attributes of the items for making recommendations. For example, DeepWide Cheng et al. (2016) combines the linear model(Wide) and feed-forward neural network(Deep) to model feature interactions simultaneously. DeepFM Guo et al. (2017) integrates a component of factorization machines and a component of deep neural networks to model low-level and high-level feature interactions, respectively. DKN Wang et al. (2018) proposed a news recommendation framework that fuse semantic-level and knowledge-level representations of news by a multi-channel CNN, and uses an attention module to dynamically calculate a users aggregated historical representation. DeepJONN Zhang et al. (2018) is a session-based model which uses a tensor based CNN to model the session representation and an RNN to capture sequential patterns in streams of clicks and associated features. DAN Zhu et al. (2019) improves DKN by designing an attention-based RNN to capture sequential information of clicked news, which achieves the state-of-the-art performance on news recommendation. Hybrid methods Cantador, Castells, and Bellogín (2011); Li et al. (2011); Liu et al. (2010) such as SCENE Li et al. (2011) usually combine several different recommender algorithms to recommend items.

Different from the above works, in this paper, we propose a novel graph neural news recommendation model with long-term and

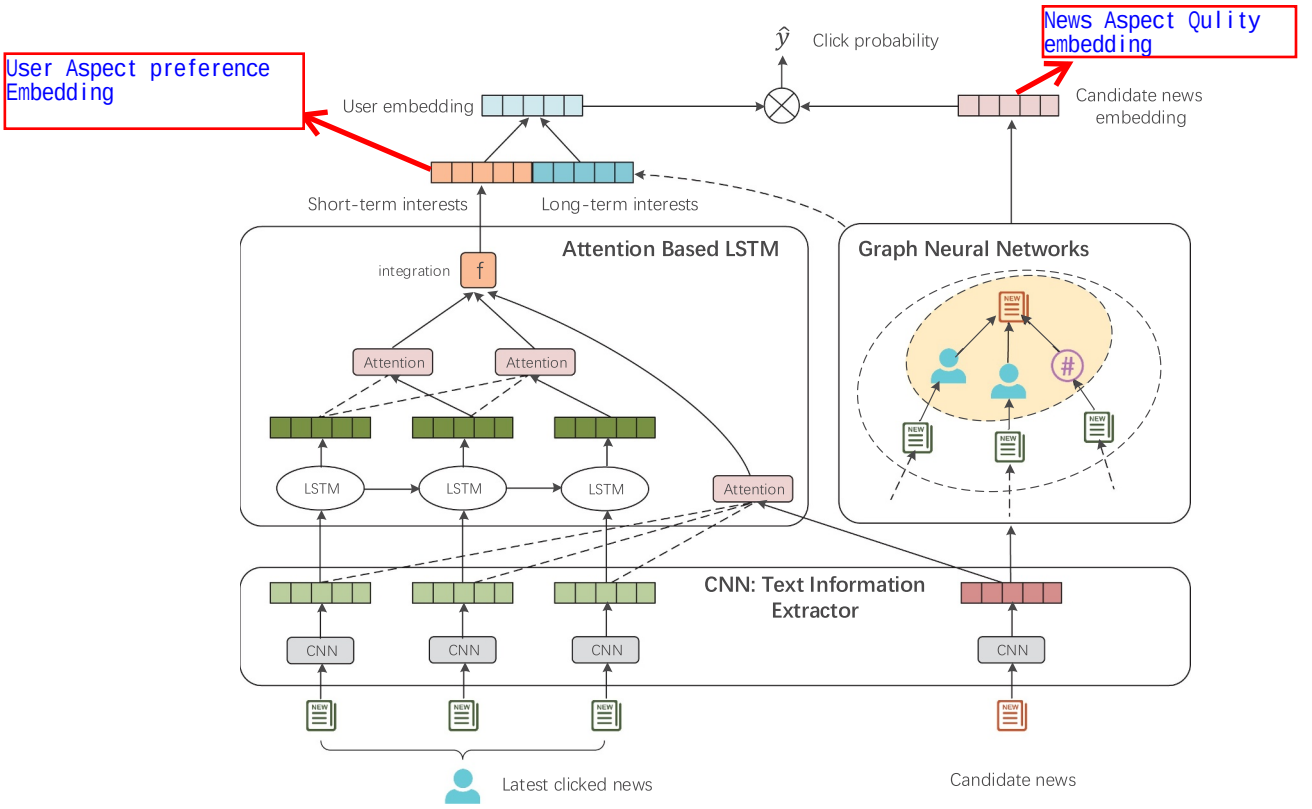

Fig. 1. The framework of GNewsRec.

short-term interest modeling. It is a hybrid method utilizing both user-item interactions and the contents of news articles. Our method extensively exploits the high-order structure information between users and items by constructing a heterogeneous graph and applying graph convolutional networks to propagate the embeddings.

3. Problem formulation

The news recommendation problem in our paper can be illustrated as follows. We have the click histories for K users $U = \{u_1, u_2, \cdots, u_K\}$ over M news items $I = \{d_1, d_2, \cdots, d_M\}$. The user-item interaction matrix $Y \in R^{K \times M}$ is defined according to users' implicit feedback, where $y_{u,d} = 1$ indicate the user u clicked the news d, otherwise $y_{u,d} = 0$. Additionally, from the click history with timestamps, we can obtain the recent click sequence $s_u = \{d_{u,1}, d_{u,2}, \cdots, d_{u,n}\}$ for a specific user u, where $d_{u,j} \in I$ is the j-th news the user u clicked.

Given the user-item interaction matrix Y as well as the users' recent click sequences S, we aim to predict whether a user u has potential interest in a news item d which he/she has not seen before. This paper considers the title and profile (a given set of entities E and their entity types C from the news page content) of news as features. Each news title T contains a sequence of words $T = \{w_1, w_2, \cdots, w_m\}$. The profile contains a sequence of entities $E = \{e_1, e_2, \cdots, e_n\}$ as well as its type set $C = \{c_1, c_2, \cdots, c_n\}$, where c_j is the type of the j-th entity e_j.

4. The proposed method

In this section, we present our graph neural news recommendation model GNewsRec with long-term and short-term interest modeling. Our model takes full advantage of the high-order structure information between users and news items by first constructing a heterogeneous graph modeling the interactions and then applying GNN to propagate the embeddings. As illustrated in Fig. 1, GNewsRec contains three main parts: CNN for text information extraction, GNN for long-term user interest modeling and news modeling, and attention based LSTM model for short-term user interest modeling. The first part extracts the news feature from the news title and profile through CNN. The second part constructs a heterogeneous user-news-topic graph with complete historic user clicks and applies GNN to encode high-order structure information for recommendation. The incorporated latent topic information can alleviate the user-item sparsity since news items with few user clicks can aggregate more information with the bridge of topics. The learned user embeddings with complete historic user clicks are supposed to encode the relatively stable long-term user interest. We also model the user's short-term interest with recent reading history through an attention based LSTM in the third part. Finally,

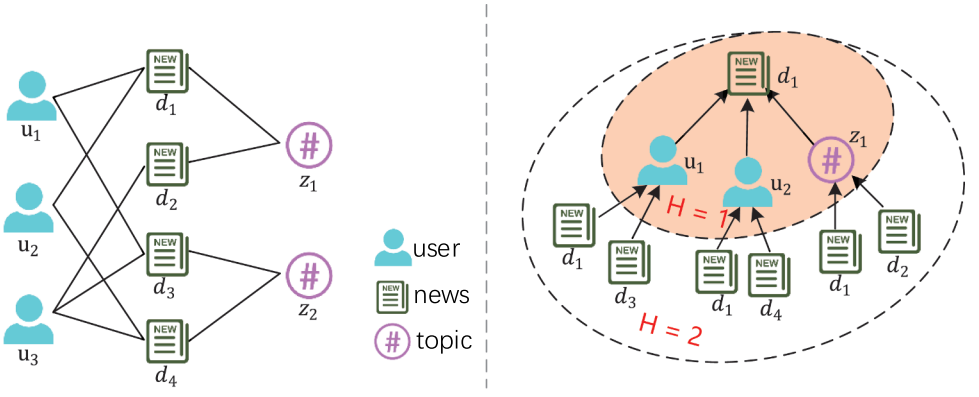

Fig. 2. Heterogeneous user-news-topic graph(left) and two-layers GNN(right).

we combine a user's long-term and short-term interests for user representation, which is then compared to candidate news representation for recommendation. We will detail the three parts as follows.

4.1. Text information extractor

We use two parallel CNNs as the news text information extractor, which respectively take the title and profile of news as inputs and learn the title-level and profile-level representations of news. The concatenation of such two representations is regarded as the final text feature representation of news.

Specifically, we represent the title as $\mathbf{T} = [\mathbf{w}_1, \cdots, \mathbf{w}_m]^T$ and the profile as $\mathbf{P} = [\mathbf{e}_1, f(\mathbf{c}_1), \mathbf{e}_2, f(\mathbf{c}_2), \cdots, \mathbf{e}_n, f(\mathbf{c}_n)]^T$, where $\mathbf{P} \in R^{2n \times k_1}$ and k_1 is the dimension of entity embedding. $f(\mathbf{c}) = \mathbf{W}_c\,\mathbf{c}$ is the transformation function. $\mathbf{W}_c \in R^{k_1 \times k_2}$ (k_2 is the dimension of entity type embedding) is the trainable transformation matrix.

The title $\mathbf{T}$ and profile $\mathbf{P}$ are respectively fed into two parallel CNNs that have separate weight parameters. Hence we separately obtain their feature representations as $\widetilde{\mathbf{T}}$ and $\widetilde{\mathbf{P}}$ through two parallel CNNs. Finally we concatenate $\widetilde{\mathbf{T}}$ and $\widetilde{\mathbf{P}}$ as the final news text feature representation:

$$\mathbf{d} = f_c([\widetilde{\mathbf{T}}; \widetilde{\mathbf{P}}]), \tag{1}$$

where $\mathbf{d} \in R^D$ and f_c is a densely connected layer.

4.2. Long-term user interest modeling and news modeling

To model long-term user interest and news, we first construct a heterogeneous user-news-topic graph with users' complete historic clicks. The incorporated topic information can help better indicate a user's interest and alleviate the sparsity of user-item interactions. Then we apply graph convolutional networks for learning embeddings of users and news items, which encodes the high-order information between users and items through propagating embeddings over the graph.

LDA：aspect：提取模型

user-news之间是1，0的关系，即是否点击，确定边的联系；
news-topics之间是相似度最大的概率关系。

4.2.1. Heterogeneous user-News-Topic graph

We incorporate the latent topic information in news articles to better indicate the user's interest and alleviate the user-item sparsity issue. Hence, we construct a heterogeneous undirected graph $G = (V, R)$ as illustrated in the left part of Fig. 2, where V and R are respectively the sets of nodes and edges. Our graph contains three types of nodes: users U, news items I and topics Z. The topics Z can be mined through the topic model LDA Blei, Ng, and Jordan (2003).

We build the user-item edges if the user u clicked a news item d, i.e., $y_{u,d} = 1$. For each news document d, we can obtain its topic distribution $\theta_d = \{\theta_{d,i}\}_{i=1,\cdots,\mathcal{K}}$, $\sum_{i=1}^{\mathcal{K}} \theta_i = 1$ through LDA. We build the connection of the news document d and the topic z with the largest probability.

Note that for testing, we can infer the topics of new documents based on the estimated LDA model Newman, Smyth, Welling, and Asuncion (2008). In this way, the new documents that do not existed in the graph can be connected with the constructed graph and update their embeddings through graph convolution. Hence, the topic information can alleviate the cold start problem as well as the sparsity issue of user-item interactions.

牛逼的方法即解决了用户稀疏问题又解决了冷启动问题！！！

4.2.2. GNN for heterogeneous user-News-Topic graph

With the constructed heterogeneous user-news-topic graph, we then apply GNN Hamilton, Ying, and Leskovec (2017); Wang, Zhao, Xie, Li, and Guo (2019a); Wang, He, Cao, Liu, and Chua (2019b) to capture high-order relationships between users and news by propagating the embeddings through it. Following are the general form of computing a certain node embedding of a single GNN layer:

计算表示 node embedding

$$\mathbf{h}_{\mathcal{N}_v} = \text{AGGREGATE}(\{\mathbf{W}^t\mathbf{h}_u^t, \forall\, u \in \mathcal{N}_v\}), \tag{2}$$

$$\mathbf{h}_v = \sigma(\mathbf{W}\cdot\mathbf{h}_{\mathcal{N}_v} + \mathbf{b}), \tag{3}$$

where AGGREGATE is the aggregator function, which aggregates information from neighboring nodes, in our paper, we use the mean aggregator which simply takes the elementwise mean of the vectors of the neighbors. $\mathcal{N}_v$ denotes the neighborhood of a certain node v and $\mathbf{W}^t$ is trainable transformation matrix for transforming different types of nodes h_u^t into the same space. $\mathbf{W}$ and $\mathbf{b}$ are the weight matrices and bias of one GNN layer to update the center node embedding $\mathbf{h}_v$.

In particular, consider the candidate pair of user u and news d. We use $U(d)$ and $Z(d)$[3] to respectively denote the set of users and topics directly connected to the news document d. In real applications, the size of $U(d)$ may vary significantly over all news documents. To keep the computational pattern of each batch fixed and more efficient, we uniformly sample a set of neighbors $S(d)$ with fixed size for each news d instead of using its full neighbors, where the size $|S(d)| = L_u$.[4]

Following Eq. (2) and (3), to characterize the topological proximity structure of news d, firstly, we compute the linear average combination of all its sampled neighbors:

建立新闻向量的拓扑结构：先计算所有邻居结点的线性组合的平均值；然后更新结点

$$\mathbf{d}_{\mathcal{N}} = \frac{1}{|S(d)|}\sum_{u\in S(d)}\mathbf{W}_u\mathbf{u} + \frac{1}{|Z(d)|}\sum_{z\in Z(d)}\mathbf{W}_z\mathbf{z}, \tag{4}$$

where $\mathbf{u} \in R^D$ and $\mathbf{z} \in R^D$ are the representations of the neighboring user and topic of news d. $\mathbf{u}$ and $\mathbf{z}$ are initialized randomly, while $\mathbf{d}$ are initialized with the text feature embeddings obtained from text information extractor (Sectioin 4.1). $\mathbf{W}_u \in R^{D\times D}$ and $\mathbf{W}_z \in R^{D\times D}$ are respectively the trainable transformation matrix for users and topics, which map them from the different spaces to the same space of news embeddings.

Then we update the candidate news embedding with the neighborhood representation $\mathbf{d}_{\mathcal{N}}$ by:

$$\widetilde{\mathbf{d}} = \sigma(\mathbf{W}^1\cdot\mathbf{d}_{\mathcal{N}} + \mathbf{b}^1), \tag{5}$$

where σ is the nonlinear function $ReLU$, and $\mathbf{W}^1 \in R^{D\times D}$ and $\mathbf{b}^1 \in R^D$ are transformation weight and bias of the first layer of GNN, respectively.

This is a single layer GNN, where the final embedding of the candidate news is only dependent on its immediate neighbors. In order to capture high-order relationships between users and news, we can extend the GNN from one layer to multiple layers, propagating the embeddings in a broader and deeper way. As shown in Fig. 2, 2-order news embeddings can be obtained as follows. We first get its 1-hop neighboring user embeddings $\mathbf{u}_l$ and topic embeddings $\mathbf{z}$ by aggregating their neighboring news embeddings using Eq. (2) and (3). Then we aggregate their embeddings $\mathbf{u}_l$ and $\mathbf{z}$ to get 2-order news embeddings $\widetilde{\mathbf{d}}$. Generally speaking, the H-order representation of an news is a mixture of initial representations of its neighbors up to H hops away.

Through the GNN, we can get the final user and news embeddings $\mathbf{u}_l$ and $\widetilde{\mathbf{d}}$ with high-order information encoded. The user embeddings learned with complete user click history are supposed to capture the relatively stable long-term user interests. However, we argue that a user could be temporally attracted to certain things, namely, a user has short-term interest, which should also be considered in personalized news recommendation.

4.3. Short-term user interest modeling

In this subsection, we present how to model a user's short-term interest using her recent click history through an attention based LSTM model. We pay attention to not only the news contents but also the sequential information.

Attention over Contents. Given a user u with her latest l clicked news $\{\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_l\}$[5], we use an attention mechanism to model the different impacts of the user's recent clicked news on the candidate news d:

$$\mathbf{u}_j = tanh(\mathbf{W}'\mathbf{d}_j + \mathbf{b}'), \tag{6}$$

$$\mathbf{u} = tanh(\mathbf{W}\mathbf{d} + \mathbf{b}), \tag{7}$$

$$\alpha_j = \frac{exp(\mathbf{v}^T(\mathbf{u} + \mathbf{u}_j))}{\sum_j exp(\mathbf{v}^T(\mathbf{u} + \mathbf{u}_j))}, \tag{8}$$

$$\mathbf{u}_c = \sum_j \alpha_j\mathbf{d}_j, \tag{9}$$

where $\mathbf{u}_c$ is the user's current content-level interest embedding, α_j is the impact weight of clicked news $d_j (j = 1, \cdots, l)$ on candidate news d, $\mathbf{W}'$, $\mathbf{W} \in R^{D\times D}$, $\mathbf{d}_j$, $\mathbf{b}_w$, $\mathbf{b}_t$, $\mathbf{v}^T \in R^D$, D is the dimension of news embedding.

Attention over Sequential Information. Besides applying attention mechanism to model user current content-level interest, we also take attention of the sequential information of the latest clicked news, thus we use an attention based LSTM Hochreiter and

[3] In this paper, we assume each news has only one topic, i.e., $|Z(d)| = 1$

[4] $S(d)$ may contain duplicates if $|U(d)| < L_u$. If $U(d) = \emptyset$, then $S(d) = \emptyset$.

[5] If the click history sequence length is less than l, we pad it with zero embeddings.

Schmidhuber (1997) to capture the sequential features.

As is shown in Fig. 1, LSTM takes user's clicked news embeddings as input, and output the user's sequential feature representation. Since each user's current click is affected by previous clicked news, the attention mechanism described above(for content-level interest modeling) is applied on each hidden state $\mathbf{h}_j$ and their previous hidden states $\{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_{j-1}\}$ ($\mathbf{h}_j$= LSTM($\mathbf{h}_{(j-1)}$, $\mathbf{d}_j$)) of the LSTM to obtain richer sequential feature representation $\mathbf{s}_j$, ($j = 1, \cdots, l$) at different click times. These features ($\mathbf{s}_1, \cdots, \mathbf{s}_l$) are integrated by a CNN to get the final sequential feature representation $\tilde{\mathbf{s}}$ of user's latest l clicked history.

We feed the concatenation of current content-level interest embedding and the sequence-level embedding into a fully connected network and get the final user's short-term interest embedding:

$$\mathbf{u}_s = \mathbf{W}_s[\mathbf{u}_c; \tilde{s}], \tag{10}$$

where $\mathbf{W}_s \in R^{D \times 2D}$.

4.4. Prediction and training

Finally, the user embedding $\mathbf{u}$ is computed by taking linear transformation over the concatenation of the long-term and short-term embedding vectors:

$$\mathbf{u} = \mathbf{W}[\mathbf{u}_l; \mathbf{u}_s], \tag{11}$$

where $\mathbf{W} \in R^{d \times 2d}$.

Then we compare the final user embedding $\mathbf{u}$ to the candidate news embedding $\tilde{\mathbf{d}}$, the probability of user u clicking news d is predicted by a DNN:

$$\hat{y} = DNN(\mathbf{u}, \tilde{\mathbf{d}}). \tag{12}$$

To train our proposed model GNewsRec, we select positive samples from the existing observed clicked reading history and equal amount of negative samples from unobserved reading. A training sample is denoted as $X = (u, x, y)$, where x is the candidate news to predict whether click or not. For each positive input sample, $y = 1$, otherwise $y = 0$. After our model, each input sample has a respective estimated probability $\hat{y} \in [0, 1]$ of the user whether will click the candidate news x. We use the cross-entropy loss as our lost function:

$$\mathcal{L} = -\Big\{ \sum_{X \in \Delta^+} y \log \hat{y} + \sum_{X \in \Delta^-} (1 - y) \log(1 - \hat{y}) \Big\} + \lambda \|W\|_2, \tag{13}$$

where Δ^+ is the positive sample set and Δ^- is the negative sample set, $\|W\|_2$ is the L2 regularization to all the trainable parameters and λ is the penalty weight. Besides, we also apply dropout and early stopping to avoiding over-fitting.

5. Experiments

5.1. Datasets

We conduct experiments on a real-world online news dataset Adressa Gulla, Zhang, Liu, Özgöbek, and Su (2017)[6], which is a click log data set with approximately 20 million page visits from a Norwegian news portal as well as a sub-sample with 2.7 million clicks. Adressa is published with the collaboration of Norwegian University of Science and Technology (NTNU) and Adressavisen (local newspaper in Trondheim, Norway) as a part of RecTech project on recommendation technology, it is one of the most comprehensive open datasets for training and evaluating news recommender systems.

The datasets are event-based including anonymized users with their clicked news logs. In addition to the click logs, the data set contains some contextual information about the users such as geographical location, active time (time spent reading an article), and session boundaries etc. We use the two light versions, named Adressa-1week, which collects news click logs as long as 1 week (from 1 January to 7 January 2017), and Adressa-10week, which collects 10 weeks (from 1 January to 31 March 2017) dataset. Following DAN Zhu et al. (2019), for each event, we just select the (sessionStart, sessionStop)[7], user id, news id, time-stamp, the title and profile of news for building our datasets.

Specifically, we first sort the news in chronological order. For the Adressa-1week dataset, we split the data as: the first 5 days' history data for graph construction and the latest l news clicked in the 5 days for short-term interest modeling, the 6-th day's for generating training pairs $< u, d >$, 20% of the last day's for validation and the left 80% for testing. Note that during testing, we reconstruct the graph with the previous 6 days' history data and use the latest l news clicked in the 6 days to model short-term user interest. Similarly, for the Adressa-10week dataset, in training period, we use the previous 50 days' data for graph construction, the following 10 days' for generating training pairs, 20% of the left 10 days' for validation and 80% for testing.

To reduce the noise of textual data, we preprocess the data as follows. We remove the stopwords of the titles and filter out meaningless entities and entity types[8] in the news profile. The statistics of our final datasets are shown in Table 1.

详细的数据集预处理过程，Nice！！！

[6] http://reclab.idi.ntnu.no/dataset/

[7] sessionStart and sessionStop determine the session boundaries.

[8] 8 types of entities including site, author, language, adressa-importance, kundeservice-access, kundeservice-importance, adressa-access and

Table 1
Statistics of the dataset.

Number	Adressa-1week	Adressa-10week
#users	537,627	590,673
#news	14,732	49,994
#events	2,527,571	23,168,411
#vocabulary	116,603	279,214
#entity-type	11	11
#average words per title	4.03	4.10
#average entity per news	22.11	21.29

5.2. Parameter setting

Topic约等于Aspect

We implement our model based on Tensorflow. The hyper-parameter settings are determined by optimizing *AUC* on validation set. They are set as follows. The dimension of both word embeddings and entity type embeddings is set as $k_1 = k_2 = 50$, and the dimension of news embeddings, user embeddings and topic embeddings are set as $D = 128$. The parameter configurations of parallel CNNs are set followed by DAN Zhu et al. (2019). The selected number of latest clicked news is set as the same ($l = 10$) as DAN Zhu et al. (2019). For LDA, the number of topics is set as $\mathcal{K} = 20$. In GNN, the fixed number of sampled neighboring users, neighboring news documents are set as $L_u = 10$ and $L_d = 30$.

The embeddings are randomly initialized using a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. And the parameters are optimized by Adam Kingma and Ba (2014) algorithm with learning rate 0.0003. L2 penalty is set to 0.005 and the dropout rate is set to 0.5. We follow previous works Wang et al. (2018); Zhu et al. (2019), and use the same parameter settings for the baseline models.

5.3. Baselines

We use the following state-of-the-art methods as baselines in our experiments:

- DMF Xue et al. (2017) is a deep matrix factorization model which uses multiple non-linear layers to process raw rating vectors of users and items. They ignore the news contents and take the implicit feedback as its input.
- DeepWide Cheng et al. (2016) is a deep learning based model that combines the linear model (Wide) and feed-forward neural network(Deep) to model low- and high-level feature interactions simultaneously. In this paper, we use the concatenation of news title and profile embeddings as features.
- DeepFM Guo et al. (2017) is also a general deep model for recommendation, which combines a component of factorization machines and a component of deep neural networks that share the input to model low- and high-level feature interactions. We use the same input as in DeepWide for DeepFM.
- DKN Wang et al. (2018) is a deep content based recommendation framework, which fuses semantic-level and knowledge-level representations of news by a multi-channel CNN. In this paper, following DAN Zhu et al. (2019), we model news title as semantic-level representations and profile as knowledge-level representations.
- DAN Zhu et al. (2019) is a deep attention based neural network for news recommendation, which improves DKN Wang et al. (2018) by considering the users click sequence information.

All the baseline models are based on deep neural networks. DMF is a collaborative filtering based model, while the others are all content based.

5.4. Experimental results

5.4.1. Comparisons of different models

In this subsection, we conduct experiments to compare our model with the state-of-the-art baseline models on two datasets, and report the results in Table 2 in terms of *AUC* and *F*1 metrics.

As we can see from Table 2, our model consistently improves all the baselines on both datasets by more than 10.67% on F1 and 2.37% on AUC. We attribute the significant superiority of our model to its three advantages: (1) Our model constructs a heterogeneous user-news-topic graph and learns better user and news embeddings with high-order information encoded by GNN. (2) Our model considers not only the long-term user interest but also the short-term interest. (3) The topic information incorporated in the heterogeneous graph can help better reflect a user's interest and alleviate the sparsity issue of user-item interactions. The news items

(*footnote continued*)
pageclass are filtered out. And the remain 11 types of entities are: concept, sentiment, entity, classification, category, adressa-tag, person, location, company, taxonomy and acronym.

Table 2
Comparison of Different Models.

Model	Adressa-1week		Adressa-10week	
	AUC(%)	F1(%)	AUC(%)	F1(%)
DMF	55.66	56.46	53.20	54.15
DeepWide	68.25	69.32	73.28	69.52
DeepFM	69.09	61.48	74.04	65.82
DKN	75.57	76.11	74.32	72.29
DAN	75.93	74.01	76.76	71.65
GNewsRec	**81.16**	**82.85**	**78.62**	**81.01**

with few user clicks can still aggregate neighboring information through the topics.

We also find that all content-based models achieve better performance than the CF-based model DMF. This is because CF-based methods cannot work well in news recommendation due to cold-start problem. Our model as a hybrid model can combine the advantages of content-based models and CF-based model. In addition, new arriving documents without user clicks can also be connected to the existing graph via topics, and update their embeddings through GNN. Thus, our model can achieve better performance.

5.4.2. Comparisons of GNewsRec Variants

Further, we compare among the variants of GNewsRec to demonstrate the efficacy of the design of our model with respect to the following aspects: GNN for learning user and news embeddings with high-order structure information encoded, combining of long-term and short-term user interests, and the incorporation of topic information. The results are shown in Table 3.

As we can see from Table 3, there is a great decline in performance when we remove the GNN module for modeling long-term user interest and news, which encodes high-order relationships on the graph. This demonstrates the superiority of our model by constructing a heterogeneous graph and applying GNN to propagate the embeddings over the graph.

Removing short-term interest modeling module will decrease the performance by around 2% in terms of both AUC and F1. It demonstrates that considering both long-term and short-term user interests is necessary.

Compared to the variant model without topic information, GNewsRec achieves significant improvements on both metrics. This is because that the topic information can alleviate the user-item sparsity issue as well as the cold-start problem. New documents with few user clicks can still aggregate neighboring information through topics. GNewsRec without topic performs slightly better than GNewsRec without short-term interest modeling, which shows that considering short-term interest is important.

5.4.3. Parameter sensitivity

In this section, we mainly explore the impact of different parameters of GNewsRec. We study the impact of different number of GNN layers, and the effect of different dimension of news, user and topic embedding D (which are set as the same).

We vary the number of GNN layers from 1 to 3. From Table 4, we can find that GNewsRec with 2-layer GNN performs best. This is because 1-layer GNN can't capture the higher-order relationships between users and news. Nevertheless, 3-layer GNN may bring massive noise to the model. Higher layers with too long relation-chains make little sense when inferring inter-node similarities Wang et al. (2018). Thus, we choose 2-layer GNN in our model GNewsRec.

We select the dimensions of D in set {32, 64, 128, 256}. Fig. 3 gives the convinced results, which are (1) Our model achieves the best performance at $D = 128$ setting, indicating than such dimension setting best express the semantic information of news, user and topic space. (2) The performance of our model first increases with the growth of D and then drops as d further increases. This is because that too low dimension has insufficient capability of capturing the necessary information, and too large dimension introduces unnecessary noise and reduces generalization ability.

Table 3
Comparison of GNewsRec variants.

Model	Adressa-1week		Adressa-10week	
	AUC(%)	F1(%)	AUC(%)	F1(%)
GNewsRec without GNN	75.93	74.01	76.76	71.65
GNewsRec without short-term interest	79.00	80.53	77.03	80.21
GNewsRec without topic	79.27	80.73	77.21	80.32
GNewsRec	**81.16**	**82.85**	**78.62**	**81.01**

Table 4
Impact of different GNN layers of GNewsRec.

Model	Adressa-1week		Adressa-10week	
	AUC(%)	F1(%)	AUC(%)	F1(%)
GNewsRec-1 layer	75.24	72.17	76.17	71.92
GNewsRec-2 layers	**81.16**	**82.85**	**78.62**	**81.01**
GNewsRec-3 layers	78.94	80.36	77.92	80.11

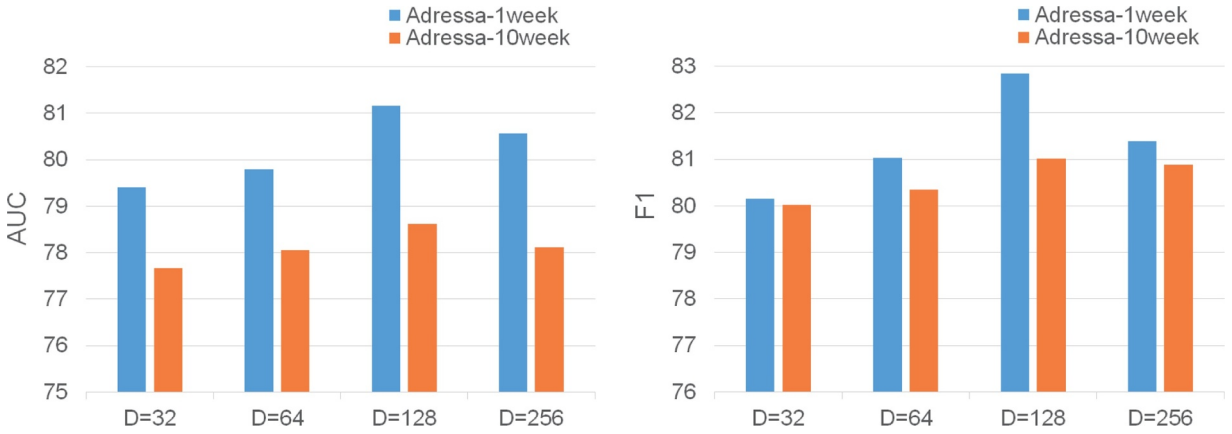

Fig. 3. Dimension sensitivity of news embedding *D*.

6. Conclusion

In this paper, we propose a novel graph neural news recommendation model GNewsRec with long-term and short-term interest modeling. Our model constructs a heterogeneous user-news-topic graph to model user-item interactions, which alleviate the sparsity of user-item interactions. Then it applies graph convolutional networks to learn user and news embeddings with high-order information encoded by propagating embeddings over the graph. The learned user embeddings with complete historic user clicks are supposed to encode a user's long-term interest. We also model a user's short-term interest using recent user reading history with an attention based LSTM model. We combine both long-term and short-term interests for user modeling, which are then compared to the candidate news representation for prediction. Experimental results on a real-world dataset show that our model significantly outperforms state-of-the-art methods on news recommendation.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 61806020, 61772082, 61972047, 61702296), the National Key Research and Development Program of China (2017YFB0803304), the Beijing Municipal Natural Science Foundation (4182043), the CCF-Tencent Open Fund, and the Fundamental Research Funds for the Central Universities.

References

Bansal, T., Das, M., & Bhattacharyya, C. (2015). *Content driven user profiling for comment-worthy recommendations of news and blog articles. Proceedings of the 9th acm conference on recommender systems*. ACM195–202.
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research, 3*(Jan), 993–1022.
Cantador, I., Castells, P., & Bellogín, A. (2011). An enhanced semantic layer for hybrid recommender systems: Application to news recommendation. *International Journal on Semantic Web and Information Systems (IJSWIS), 7*(1), 44–78.
Cao, Y., Wang, X., He, X., Hu, Z., & Chua, T.-S. (2019). *Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. The world wide web conference*. ACM151–161.
Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., ... Ispir, M., et al. (2016). *Wide & deep learning for recommender systems. Proceedings of the 1st workshop on deep learning for recommender systems*. ACM7–10.
Das, A. S., Datar, M., Garg, A., & Rajaram, S. (2007). *Google news personalization: scalable online collaborative filtering. Proceedings of the 16th international conference on world wide web*. ACM271–280.
De Francisci Morales, G., Gionis, A., & Lucchese, C. (2012). *From chatter to headlines: harnessing the real-time web for personalized news recommendation. Proceedings of the fifth acm international conference on web search and data mining*. ACM153–162.
Gulla, J. A., Zhang, L., Liu, P., Özgöbek, Ö., & Su, X. (2017). *The adressa dataset for news recommendation. Proceedings of the international conference on web intelligence*. ACM1042–1048.
Guo, H., Tang, R., Ye, Y., Li, Z., & He, X. (2017). *Deepfm: a factorization-machine based neural network for ctr prediction. Proceedings of the 26th international joint conference on artificial intelligence*1725–1731.
Hamilton, W., Ying, Z., & Leskovec, J. (2017). *Inductive representation learning on large graphs. Advances in neural information processing systems*1024–1034.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T.-S. (2017). *Neural collaborative filtering. Proceedings of the 26th international conference on world wide web*. International World Wide Web Conferences Steering Committee173–182.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735–1780.
Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., & Heck, L. (2013). *Learning deep structured semantic models for web search using clickthrough data. Proceedings of the 22nd acm international conference on information & knowledge management*. ACM2333–2338.
IJntema, W., Goossen, F., Frasincar, F., & Hogenboom, F. (2010). *Ontology-based news recommendation. Proceedings of the 2010 edbt/icdt workshops*. ACM16.
Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
Kompan, M., & Bieliková, M. (2010). *Content-based news recommendation. International conference on electronic commerce and web technologies*. Springer61–72.
Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). *A contextual-bandit approach to personalized news article recommendation. Proceedings of the 19th international conference on world wide web*. ACM661–670.
Li, L., Wang, D., Li, T., Knox, D., & Padmanabhan, B. (2011). *Scene: a scalable two-stage personalized news recommendation system. Proceedings of the 34th international acm sigir conference on research and development in information retrieval*. ACM125–134.
Li, S., Kawale, J., & Fu, Y. (2015). *Deep collaborative filtering via marginalized denoising auto-encoder. Proceedings of the 24th acm international on conference on information and knowledge management*. ACM811–820.
Liu, J., Dolan, P., & Pedersen, E. R. (2010). *Personalized news recommendation based on click behavior. Proceedings of the 15th international conference on intelligent user interfaces*. ACM31–40.
Liu, M., Wang, X., Nie, L., Tian, Q., Chen, B., & Chua, T.-S. (2018). *Cross-modal moment localization in videos. 2018 acm multimedia conference on multimedia conference*. ACM843–851.
Marlin, B., & Zemel, R. S. (2004). *The multiple multiplicative factor model for collaborative filtering. Proceedings of the twenty-first international conference on machine learning*. ACM73.
Newman, D., Smyth, P., Welling, M., & Asuncion, A. U. (2008). *Distributed inference for latent dirichlet allocation. Advances in neural information processing systems*1081–1088.
Phelan, O., McCarthy, K., & Smyth, B. (2009). *Using twitter to recommend real-time topical news. Proceedings of the third acm conference on recommender systems*. ACM385–388.
Rendle, S. (2010). *Factorization machines. 2010 ieee international conference on data mining*. IEEE995–1000.
Wang, C., & Blei, D. M. (2011). *Collaborative topic modeling for recommending scientific articles. Proceedings of the 17th acm sigkdd international conference on knowledge discovery and data mining*. ACM448–456.
Wang, H., Zhang, F., Xie, X., & Guo, M. (2018). *Dkn: Deep knowledge-aware network for news recommendation. Proceedings of the 2018 world wide web conference*. International World Wide Web Conferences Steering Committee1835–1844.
Wang, H., Zhao, M., Xie, X., Li, W., & Guo, M. (Zhao, Xie, Li, Guo, 2019a). *Knowledge graph convolutional networks for recommender systems. The world wide web conference*. ACM3307–3313.
Wang, X., He, X., Cao, Y., Liu, M., & Chua, T.-S. (He, Cao, Liu, Chua, 2019b). *Kgat: Knowledge graph attention network for recommendation. Proceedings of the 25th acm sigkdd international conference on knowledge discovery and data mining*. ACM950–958.
Wang, X., He, X., Wang, M., Feng, F., & Chua, T. (He, Wang, Feng, Chua, 2019c). *Neural graph collaborative filtering. Proceedings of the 42nd international acm sigir conference on research and development in information retrieval*165–174.
Wang, X., Yu, L., Ren, K., Tao, G., Zhang, W., Yu, Y., & Wang, J. (2017). *Dynamic attention deep model for article recommendation by learning human editors' demonstration. Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*. ACM2051–2059.
Wu, Y., DuBois, C., Zheng, A. X., & Ester, M. (2016). *Collaborative denoising auto-encoders for top-n recommender systems. Proceedings of the ninth acm international conference on web search and data mining*. ACM153–162.
Xue, H.-J., Dai, X., Zhang, J., Huang, S., & Chen, J. (2017). *Deep matrix factorization models for recommender systems. Proceedings of the 26th international joint conference on artificial intelligence*3203–3209.
Zhang, L., Liu, P., & Gulla, J. A. (2018). *A deep joint network for session-based news recommendations with contextual augmentation. Proceedings of the 29th on hypertext and social media*. ACM201–209.
Zhu, Q., Zhou, X., Song, Z., Tan, J., & Guo, L. (2019). *Dan : Deep attention neural network for news recommendation. Aaai 2019*5973–5980.