FUM: Fine-grained and Fast User Modeling for News Recommendation

Tao Qi
Department of Electronic Engineering,
Tsinghua University
taoqi.qt@gmail.com

Fangzhao Wu*
Microsoft Research Asia
wufangzhao@gmail.com

Chuhan Wu
Department of Electronic Engineering,
Tsinghua University
wuchuhan15@gmail.com

Yongfeng Huang
Department of Electronic Engineering,
Tsinghua University
yfhuang@tsinghua.edu.cn

ABSTRACT

User modeling is important for news recommendation. Existing methods usually first encode user's clicked news into news embeddings independently and then aggregate them into user embedding. However, the word-level interactions across different clicked news from the same user, which contain rich detailed clues to infer user interest, are ignored by these methods. In this paper, we propose a fine-grained and fast user modeling framework (*FUM*) to model user interest from fine-grained behavior interactions for news recommendation. The core idea of *FUM* is to concatenate the clicked news into a long document and transform user modeling into a document modeling task with both intra-news and inter-news word-level interactions. Since vanilla transformer cannot efficiently handle long document, we apply an efficient transformer named Fastformer to model fine-grained behavior interactions. Extensive experiments on two real-world datasets verify that *FUM* can effectively and efficiently model user interest for news recommendation.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

News Recommendation, Fine-Grained User Modeling, Efficient User Modeling

ACM Reference Format:
Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2022. FUM: Fine-grained and Fast User Modeling for News Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/XXXXXX.XXXXXX

*The corresponding author.

SIGIR '22, July 11–15, 2022, Madrid, Spain.

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/XXXXXX.XXXXXX

	Texts of user's clicked news
1	The challenge in the new story for Iron Man.
2	The upcoming movies of Netflix in 2022.
3	The success of Marvel's Avengers.
4	Adele says if 30 doesn't come now it never will.
5	The most popular songs on YouTube in this week.
6	Vinyl and CD sales both went up in 2021, data says.

Figure 1: The news clicked by a randomly selected user. Word-level relatedness across texts of user's clicked news provide detailed clues to understand user interest.

1 INTRODUCTION

News recommendation methods can alleviate the information overload, which are important for improving user experience and developing smart cities [3, 7, 8, 15]. A critical step of news recommendation is to accurately model the interest of a target user [12, 23, 33]. Existing methods usually first independently encode user's clicked news into news embeddings and then aggregate them to build user embedding [13, 14, 16, 26, 31, 34]. For example, Wu et al. [25] first employ the self-attention mechanism to learn news embeddings for user's clicked news from their titles, independently. Then they attentively aggregate embeddings of clicked news to learn user embedding. An et al. [1] propose to first utilize a CNN network to learn representation of each news from their titles and categories. They further learn user representations from embeddings of user's previously clicked news via a GRU network and ID embeddings.

In fact, word-level interactions across clicked news from the same user contain rich detailed clues to understand user interest [19]. For example, according to the reading history of an example user in Fig. 1, we can infer the user may be interested in the movie of Iron Man from the relatedness of the word "movies" in the 2-nd clicked news and the word "Iron Man" in the 1-st clicked news. Besides, we can also target the potential user interest in the song of Adele from the relatedness of the word "Adele" in the 4-th clicked news and the word "songs" in the 5-th clicked news. However, most of the existing methods neglect word-level behavior interactions when modeling user interest, which may lead to inferior user modeling.

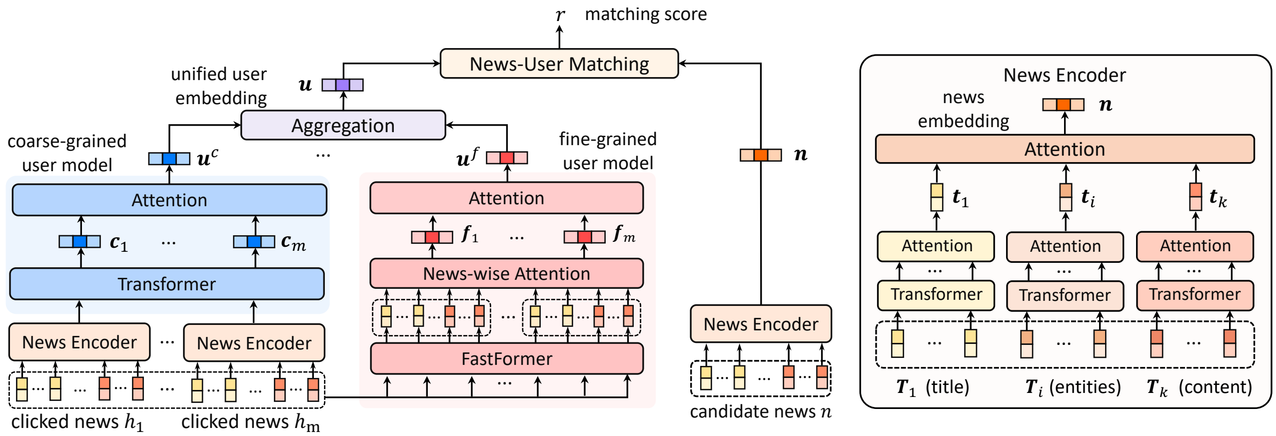

Figure 2: The framework of *FUM* for news recommendation.

In this paper, we propose a fine-grained and fast user modeling framework (named *FUM*) to model user interest from fine-grained behavior interactions for news recommendation. In *FUM*, we first concatenate texts of user's clicked news as a long user document and transform the user modeling task into a document modeling task. Then we capture both intra-news and inter-news word-level behavior interactions across the long user document to understand user interest in a fine-grained way. In addition, the vanilla transformer network is usually inefficient for modeling long documents due to its quadric complexity, thereby we adopt an efficient transformer network named Fastformer [29] to capture fine-grained behavior interactions in *FUM*. We conduct extensive experiments on two real-world datasets. Experimental results demonstrate that *FUM* can achieve better performance than many news recommendation methods and meanwhile efficiently model user interest.

2 FUM

2.1 Problem Formulation

We assume that a news article h induces k genres of textual information (e.g., titles and entities) $[T_1, T_2, ..., T_k]$, where T_i is the i-th genre of the news text. The textual sequence T_i is composed of multiple tokens: $T_i = [t_{i,1}, t_{i,2}, ..., t_{i,l}]$, where $t_{i,j}$ is the j-th word token in the sequence T_i, and l is the maximum length of the sequence. Besides, we assume that a target user u has previously clicked m news, where h_j denotes the j-th clicked news. The news recommendation task aims to mine user interest from user's reading history to further match candidate news for recommendation. Our work focuses on accurately and efficiently modeling user interest from fine-grained word-level interactions across clicked news.

2.2 Fine-grained and Fast User Modeling

The framework of our *FUM* approach is shown in Fig. 2. *FUM* is composed of a fine-grained user model and a coarse-grained user model. The fine-grained user model is used to capture user interest from word-level behavior interactions. Its core is to concatenate user's clicked news as a long document and capture intra- and inter-news word-level interactions to model user interest. Specifically, we first encode the i-th genre of news text T_i into a text embedding sequence $\mathbf{T}_i \in \mathbb{R}^{l \times d}$ via a genre-specific embedding layer, where d is embedding dimension. Then we concatenate texts sequences of user's reading history into a long sequence $\mathbf{T} \in \mathbb{R}^{mkl \times d}$:

$$\mathbf{T} = [\mathbf{T}_1^1; ...; \mathbf{T}_k^1; ...; \mathbf{T}_1^m; ...; \mathbf{T}_k^m], \tag{1}$$

where $\mathbf{T}_j^i$ is the j-th text embedding sequence of the i-th clicked news h_i and ; is the concatenation operation. Besides, different genres of news texts usually have different semantic characteristics and meanwhile the positional information of texts are also important for semantic understanding. Thus, to further enrich the embedding sequence of the user document, we concatenate text embeddings of each token with its genre and position embeddings and build a behavior embedding sequence $\mathbf{H} \in \mathbb{R}^{L \times g}$, where g is dimension of the concatenated token embedding, and L denotes the total length (i.e., mkl) of the behavior embedding sequence.

The transformer network [18] is an effective technique for document modeling. However, due to its quadratic complexity, the vanilla transformer network cannot efficiently model long documents. Fortunately, some efficient transformer methods have been proposed. To model fine-grained behavior interactions across the long user document, we employ a SOTA efficient transformer network named *Fastformer* [29]. Take an arbitrary attention head as example, the core idea of *Fastformer* is to first summarize global contexts into an embedding $\mathbf{q}$ and then transform embeddings of each token based on their relatedness with global contexts:

$$\mathbf{q} = Att(\mathbf{q}_1, ..., \mathbf{q}_L), \quad \mathbf{q}_i = \mathbf{W}_q \mathbf{h}_i, \tag{2}$$

$$\mathbf{k} = Att(\mathbf{q} * \mathbf{k}_1, ..., \mathbf{q} * \mathbf{k}_L), \quad \mathbf{k}_i = \mathbf{W}_k \mathbf{h}_i \tag{3}$$

$$\hat{\mathbf{h}}_i = \mathbf{W}_o(\mathbf{k} * \mathbf{v}_i), \quad \mathbf{v}_i = \mathbf{W}_v \mathbf{h}_i \tag{4}$$

where $\mathbf{h}_i$ and $\hat{\mathbf{h}}_i$ denote the input and output of the i-th token in the behavior embedding sequence, $*$ denotes element-wise product, $Att(\cdot)$ denotes the attention pooling network and $\mathbf{W}_q$, $\mathbf{W}_k$, $\mathbf{W}_v$ and $\mathbf{W}_o$ denote trainable projection parameters. We remark that *Fastformer* can also be replaced by other efficient transformers. Then we can concatenate outputs of different attention heads and build a unified contextual representations $\mathbf{g}_i$ for the i-th token. Next, we adopt an attention network to learn embeddings for each

Table 1: News recommendation performance of different methods on *MIND* and *Feeds*. The improvement of *FUM* over baseline methods is significant at level $p < 0.001$ based on t-test.

	MIND				*Feeds*			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
GRU	65.47±0.18	31.15±0.22	33.64±0.24	39.34±0.24	62.95±0.13	27.57±0.08	31.55±0.12	37.18±0.11
DKN	67.19±0.13	32.97±0.19	35.87±0.22	41.53±0.17	64.02±0.25	28.65±0.13	32.97±0.17	38.54±0.17
NPA	67.42±0.15	32.97±0.18	35.90±0.23	41.54±0.20	64.83±0.47	29.21±0.36	33.64±0.47	39.18±0.48
KRED	67.77±0.15	33.39±0.15	36.34±0.17	42.04±0.15	64.92±0.14	29.27±0.08	33.71±0.13	39.25±0.12
GNewsRec	68.38±0.09	33.46±0.22	36.44±0.23	42.15±0.20	65.02±0.11	29.28±0.10	33.74±0.13	39.28±0.13
NAML	68.16±0.11	33.31±0.07	36.26±0.10	41.94±0.08	65.31±0.12	29.47±0.07	33.99±0.09	39.57±0.12
NRMS	68.33±0.27	33.55±0.27	36.53±0.32	42.18±0.30	65.21±0.12	29.39±0.05	33.87±0.06	39.46±0.08
LSTUR	68.53±0.10	33.58±0.15	36.54±0.18	42.23±0.17	65.31±0.20	29.54±0.15	34.08±0.19	39.63±0.19
FIM	68.15±0.33	33.36±0.27	36.38±0.30	42.02±0.31	65.47±0.12	29.62±0.07	34.19±0.09	39.72±0.09
FUM	**70.01**±0.10	**34.51**±0.13	**37.68**±0.14	**43.38**±0.13	**66.93**±0.19	**30.49**±0.16	**35.31**±0.21	**40.87**±0.18

clicked news by aggregating embeddings of their tokens:

$$\mathbf{f}_i = Att(\mathbf{g}_{(i-1)kl+1}, \mathbf{g}_{(i-1)kl+2}, ..., \mathbf{g}_{ikl}), \tag{5}$$

where $\mathbf{f}_i$ represents the i-th clicked news. Finally, we pooling them to build the user embedding $\mathbf{u}^f = Att(\mathbf{f}_1, ..., \mathbf{f}_m)$. In this way, we can efficiently and effectively model and encode user interest from word-level fine-grained behavior interactions.

Besides, we also adopt a coarse-grained user model to better summarize user interest from news-level behavior interactions. We first apply a news encoder to transform user's clicked news into embeddings. Details of the news encoder is introduced in Sec. 2.3. Then we apply a transformer network to model news-level behavior interactions across user's clicked news, where $\mathbf{c}_i$ is the contextualized embedding of h_i. Finally, we build a user embedding $\mathbf{u}^c = Att(\mathbf{c}_1, ..., \mathbf{c}_m)$ from news-level behavior interactions and aggregate it with $\mathbf{u}^f$ to form a unified user embedding $\mathbf{u} = \mathbf{u}^f + \mathbf{u}^c$.

2.3 News Encoder

Next, we briefly introduce the architecture of the news encoder in *FUM*. For the i-th genre of news text, we apply a text encoder to learn a genre-specific news embedding $\mathbf{t}_i$ from $\mathbf{T}_i$. Motivated by Ge et al. [4], the text encoder is implemented by the stack of a transformer and an attention network. Then, we attentively aggregate genre-specific news embeddings to learn the news embedding $\mathbf{n}$.

2.4 News Recommendation

Following Wu et al. [24, 27], we match the target user u and the candidate news n based on the inner product of their embeddings $r = \mathbf{u}^T\mathbf{n}$. Then candidate news are ranked based on their matching scores r for news recommendation. Besides, we train models based on the BPR loss [17]: $\mathcal{L} = -\frac{1}{|\mathcal{D}|}\sum_{i=1}^{|\mathcal{D}|}\sigma(r_i^c - r_i^n)$, where $\mathcal{D}$ is the training data set, σ is the sigmoid function, r_i^p and r_i^n are matching scores for the i-th clicked and non-clicked news sample.

3 EXPERIMENT

3.1 Dataset and Experimental Settings

We conduct experiments on two real-world datasets: *MIND* and *Feeds*. *MIND* is a public dataset based on user data sampled from Microsoft News [30]. *Feeds* is based on user data sampled from the news feeds platform of Microsoft during Jan. 23 to Apr. 01, 2020 (13 weeks). We select 200,000 news impressions in the first ten weeks for training and validation, and 100,000 impressions in the last three weeks for evaluation. Codes are in https://github.com/taoqi98/FUM.

In experiments, we utilize news topic labels, description texts of entities, titles, and abstracts for news modeling. Their embeddings are initialized by 300-dimensional glove embeddings [11] and fine-tuned in experiments. Besides, we adopt users' recent 50 clicked news to model interest. In *FUM*, the transformer and *Fastformer* networks are set to 20 heads, and each head outputs 20-dimensional vectors. The attention networks are implemented by MLP networks. We adopt Adam [6] with 0.0001 learning rate to train models for 2 epoch. We tune hyper-parameters based on the validation set.

3.2 Performance Evaluation

We compare *FUM* with several SOTA news recommendation methods: (1) *GRU* [10]: propose to build user embeddings via a GRU network. (2) *DKN* [20]: propose an attentive memory network to learn user embeddings. (3) *NPA* [22]: propose a personalized attention mechanism to learn news and user embeddings. (4) *KRED* [9]: propose a knowledge-aware graph network to learn news embeddings from news titles and entities. (5) *GNewsRec* [5]: model user interest from user-news graph via a GRU and GNN network. (6) *NAML* [21]: learn user embeddings via an attention network. (7) *LSTUR* [1]: model long- and short-term user interest via GRU network and user IDs. (8) *NRMS* [25]: learn user embeddings via self-attention networks. (9) *FIM* [19]: model user interest in news from the matching of news texts and reading history via CNN network.

We repeat experiments of different methods 5 times and show average results and standard deviations in Table 1. Results show that *FUM* can achieve much better performance than baseline methods, e.g., *LSTUR* and *NRMS*. This is because baseline methods can only capture news-level behavior interactions to model user interest. This is because fine-grained behavior interactions across user's clicked news at word-level contain rich detailed clues to understand user interest. However, baseline methods usually neglect the fine-grained behavior interactions and thereby only achieve inferior performance. Different from these methods, in *FUM* we concatenate

Table 2: Efficiency comparison of user modeling methods on both model training and inference based on 1k samples.

	GRU	DKN	NAML	NPA	KRED	GNewsRec	LSTUR	NRMS	FIM	FUM
Training Time	11.46s	8.19s	7.98s	8.10s	10.40s	10.72s	11.53s	11.39s	15.85s	13.21s
Inference Time	2.41s	44.90s	1.23s	1.15s	1.24s	86.90s	2.43s	2.16s	350.38s	2.75s
Cacheable	✓	✗	✓	✓	✓	✗	✓	✓	✗	✓

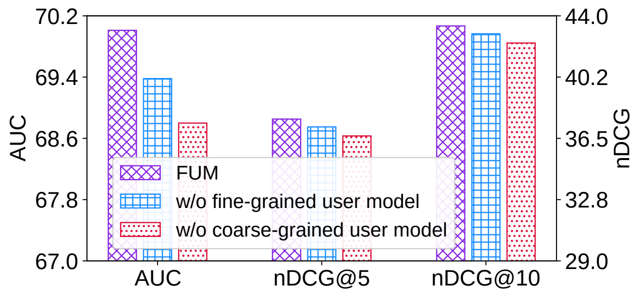

Figure 3: Ablation study of our *FUM* approach.

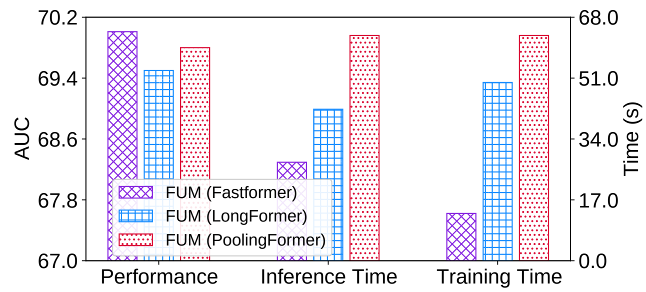

Figure 4: *FUM* with different efficient transformers. The training and inference time are based on 1k and 10k samples.

texts of user's reading history as a long document and apply an efficient transformer network to capture the fine-grained behavior interactions. Thus, our *FUM* approach can more accurately model user interest from fine-grained behavior interactions and achieve more effective news recommendation performance.

3.3 Efficiency Comparison

Next, we compare the efficiency of different methods on both model training and inference. In Table 2, we first summarize the average time of different methods for training and inferring 1000 samples. Due to the space limitation, we only show results on *MIND* in the following sections. According to Table 2, *FUM* achieves comparable or better efficiency than methods that neglects fine-grained behavior interactions. This is because in *FUM* we adopt a SOTA transformer network proposed for efficient long document modeling to capture fine-grained behavior interactions. Thus *FUM* can efficiently model fine-grained interactions of the long user document to mine user interest. Besides, real-world systems usually have strict online latency constraints [28]. Thus, in the practice on real-world systems, news and user representations are expected to be offline computed and cached in the platform to improve online efficiency. Like some baseline methods, news and user representations of *FUM* are also cacheable, which further verify the feasibility of *FUM* in practice.

3.4 Ablation Study

Next, we conduct an ablation study to verify the effectiveness of the fine- and coarse-grained user model in *FUM* (Fig. 3). First, after removing the fine-grained user model, the performance of *FUM* seriously declines. This is because fine-grained interactions across different clicked news from the same user usually contain rich clues to understand user interest. The fine-grained user model can effectively capture word-level interactions and better model user interest. Second, removing the coarse-grained user model also hurts performance. This is because intra-news behavior interactions are also important for user modeling, which can be effectively captured by the coarse-grained user model in *FUM*. Besides, the coarse-grained user model also outperforms the fine-grained user model, which may be because intra-news interactions cannot be effectively exploited by the fine-grained model.

3.5 FUM with Different Efficient Transformers

Next, we apply different efficient transformers to *FUM* to verify their impacts. Besides *FastFormer*, we apply two other SOTA efficient transformers, i.e., *LongFormer* [2] and *PoolingFormer* [32] in *FUM* (Fig. 4). We first find *FUM* with various transformers can consistently outperform baselines, which verifies the importance of fine-grained user modeling. Second, *Fastformer* significantly improves efficiency of *FUM* than other transformers. Thus, we choose *FastFormer* for the fine-grained user modeling in *FUM*.

4 CONCLUSION

In this paper, we propose a fine-grained and fast user modeling framework for news recommendation (named *FUM*), which can understand user interest from fine-grained behavior interactions. In *FUM*, we first concatenate user's clicked news as a long document. Then we employ an efficient transformer network named *Fastformer* to capture fine-grained behavior interactions from word-level to target user interest more accurately. Extensive experiments on two-real world datasets verify that *FUM* can outperform many news recommendation methods and meanwhile efficiently model user interest from fine-grained behavior interactions.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Project of China under Grant 2018YFB2101501 and Tsinghua-Toyota Joint Research Funds 20213930033.

REFERENCES

[1] 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 *ACL*. 336–345.

[2] Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150* (2020).

[3] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google news personalization: scalable online collaborative filtering. In *WWW*. 271–280.

[4] Suyu Ge, Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020. Graph enhanced representation learning for news recommendation. In *WWW*. 2863–2869.

[5] Linmei Hu, Chen Li, Chuan Shi, Cheng Yang, and Chao Shao. 2020. Graph neural news recommendation with long-term and short-term interest modeling. *IP&M* (2020), 102142.

[6] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[7] Joseph A Konstan, Bradley N Miller, David Maltz, Jonathan L Herlocker, Lee R Gordon, and John Riedl. 1997. GroupLens: applying collaborative filtering to Usenet news. *Commun. ACM* (1997), 77–87.

[8] Chen Lin, Runquan Xie, Xinjun Guan, Lei Li, and Tao Li. 2014. Personalized news recommendation via implicit social experts. *JIS* (2014), 1–18.

[9] Danyang Liu, Jianxun Lian, Shiyin Wang, Ying Qiao, Jiun-Hung Chen, Guangzhong Sun, and Xing Xie. 2020. KRED: Knowledge-aware document representation for news recommendations. In *RecSys*. 200–209.

[10] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *KDD*. 1933–1942.

[11] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. 1532–1543.

[12] Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2021. Personalized News Recommendation with Knowledge-aware Interactive Matching. In *SIGIR*. 61–70.

[13] Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2021. PP-Rec: News Recommendation with Personalized User Interest and Time-aware News Popularity. In *ACL*. 5457–5467.

[14] Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2020. Privacy-Preserving News Recommendation Model Learning. In *EMNLP: Findings*. 1423–1432.

[15] Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2021. Uni-FedRec: A Unified Privacy-Preserving News Recommendation Framework for Model Training and Online Serving. In *EMNLP: Findings*. 1438–1448.

[16] Tao Qi, Fangzhao Wu, Chuhan Wu, Peiru Yang, Yang Yu, Xing Xie, and Yongfeng Huang. 2021. HieRec: Hierarchical User Interest Modeling for Personalized News Recommendation. In *ACL*. 5446–5456.

[17] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. 452–461.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*. 6000–6010.

[19] Heyuan Wang, Fangzhao Wu, Zheng Liu, and Xing Xie. 2020. Fine-grained interest matching for neural news recommendation. In *ACL*. 836–845.

[20] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. DKN: Deep knowledge-aware network for news recommendation. In *WWW*. 1835–1844.

[21] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with attentive multi-view learning. *IJCAI* (2019), 3863–3869.

[22] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Npa: Neural news recommendation with personalized attention. In *KDD*. 2576–2584.

[23] Chuhan Wu, Fangzhao Wu, Mingxiao An, Yongfeng Huang, and Xing Xie. 2019. Neural News Recommendation with Topic-Aware News Representation. In *ACL*. 1154–1159.

[24] Chuhan Wu, Fangzhao Wu, Mingxiao An, Tao Qi, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with heterogeneous user behavior. In *EMNLP*. 4876–4885.

[25] Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with multi-head self-attention. In *EMNLP*. 6390–6395.

[26] Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. 2021. Personalized News Recommendation: A Survey. *arXiv preprint arXiv:2106.08934* (2021).

[27] Chuhan Wu, Fangzhao Wu, Yongfeng Huang, and Xing Xie. 2021. User-as-Graph: User Modeling with Heterogeneous Graph Pooling for News Recommendation. In *IJCAI*. 1624–1630.

[28] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Empowering News Recommendation with Pre-trained Language Models. In *SIGIR*. 1652–1656.

[29] Chuhan Wu, Fangzhao Wu, Tao Qi, Yongfeng Huang, and Xing Xie. 2021. Fastformer: Additive Attention Can Be All You Need. *arXiv preprint arXiv:2108.09084* (2021).

[30] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. MIND: A large-scale dataset for news recommendation. In *ACL*. 3597–3606.

[31] Jingwei Yi, Fangzhao Wu, Chuhan Wu, Ruixuan Liu, Guangzhong Sun, and Xing Xie. 2021. Efficient-FedRec: Efficient Federated Learning Framework for Privacy-Preserving News Recommendation. In *EMNLP*. 2814–2824.

[32] Hang Zhang, Yeyun Gong, Yelong Shen, Weisheng Li, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2021. Poolingformer: Long document modeling with pooling attention. In *ICML*. 12437–12446.

[33] Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. 2018. DRN: A deep reinforcement learning framework for news recommendation. In *WWW*. 167–176.

[34] Qiannan Zhu, Xiaofei Zhou, Zeliang Song, Jianlong Tan, and Guo Li. 2019. DAN: Deep attention neural network for news recommendation. In *AAAI*. 5973–5980.