Empowering News Recommendation with Pre-trained Language Models

Chuhan Wu[1], Fangzhao Wu[2], Tao Qi[1], Yongfeng Huang[1]

[1]Department of Electronic Engineering & BNRist, Tsinghua University, Beijing 100084

[2]Microsoft Research Asia, Beijing 100080, China

{wuchuhan15,wufangzhao,taoqi.qt}@gmail.com,yfhuang@tsinghua.edu.cn

ABSTRACT

Personalized news recommendation is an essential technique for online news services. News articles usually contain rich textual content, and accurate news modeling is important for personalized news recommendation. Existing news recommendation methods mainly model news texts based on traditional text modeling methods, which is not optimal for mining the deep semantic information in news texts. Pre-trained language models (PLMs) are powerful for natural language understanding, which has the potential for better news modeling. However, there is no public report that show PLMs have been applied to news recommendation. In this paper, we report our work on exploiting pre-trained language models to empower news recommendation. Offline experimental results on both monolingual and multilingual news recommendation datasets show that leveraging PLMs for news modeling can effectively improve the performance of news recommendation. Our PLM-empowered news recommendation models have been deployed to the Microsoft News platform, and achieved significant gains in terms of both click and pageview in both English-speaking and global markets.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Computing methodologies** → **Natural language processing**;

KEYWORDS

News recommendation, pre-trained language model

1 INTRODUCTION

News recommendation techniques have played critical roles in many online news platforms to alleviate the information overload of users [15]. News modeling is an important step in news recommendation, because it is a core technique to understand the content

of candidate news and a prerequisite for inferring user interests from clicked news. Since news articles usually have rich textual information, news texts modeling is the key for understanding news content for news recommendation. Existing news recommendation methods usually model news texts based on traditional NLP models [15, 19, 20, 22, 23, 25, 26]. For example, Wang et al. [20] proposed to use a knowledge-aware CNN network to learn news representations from embeddings of words and entities in news title. Wu et al. [23] proposed to use multi-head self-attention network to learn news representations from news titles. However, it is difficult for these shallow models to understand the deep semantic information in news texts [18]. In addition, their models are only learned from the supervisions in the news recommendation task, which may not be optimal for capturing the semantic information.

Pre-trained language models (PLMs) have achieved great success in NLP due to their strong ability in text modeling [2, 5, 6, 11, 12, 14, 28]. Different from traditional models that are usually directly trained with labeled data in specific tasks, PLMs are usually first pre-trained on a large unlabeled corpus via self-supervision to encode universal text information [5]. Thus, PLMs can usually provide better initial point for finetuning in downstream tasks [16]. In addition, different from many traditional NLP methods with shallow models [9, 10, 24, 29], PLMs are usually much deeper with a huge number of parameters. For example, the BERT-Base model contains 12 Transformer layers and up to 109M parameters [5]. Thus, PLMs may have greater ability in modeling the complicated contextual information in news text, which have the potentials to improve news text modeling for news recommendation.

In this paper, we present our work on empowering large-scale news recommendation with pre-trained language models.[1] Different from existing news recommendation methods that use shallow NLP models for news modeling, we explore to model news with pre-trained language models and finetune them with the news recommendation task. Offline experiments on real-world English and multilingual news recommendation datasets validate that incorporating PLMs into news modeling can consistently improve the news recommendation performance. In addition, our PLM-empowered news recommendation models have been deployed to the Microsoft News platform.[2] To our best knowledge, this is the first reported effort to empower large-scale news recommender systems with PLMs. The online flight experiments show that our PLM-empowered news recommendation models achieved 8.53% click and 2.63% pageview gains in English-speaking markets, and 10.68% click and 6.04% pageview gains in other 43 global markets.

---

[1]Source codes will be available at https://github.com/wuch15/PLM4NewsRec.
[2]https://microsoftnews.msn.com

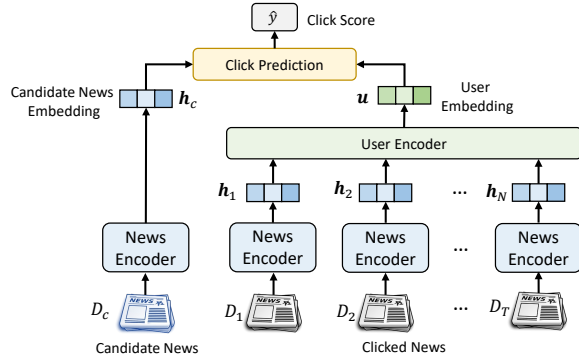Figure 1: A common framework of news recommendation.

Figure 2: The framework of PLM empowered news recommendation.

2 METHODOLOGY

In this section, we introduce the details of PLM-empowered news recommendation. We first introduce the general news recommendation model framework, and then introduce how to incorporate PLMs into this framework to empower news modeling.

2.1 General News Recommendation Framework

The general framework of news recommendation used in many existing methods [1, 15, 21, 23] is shown in Fig. 1. The core components in this framework include a news encoder that aims to learn the embeddings of news from their texts, a user encoder that learns user embedding from the embeddings of clicked news, and a click prediction module that computes personalized click score for news ranking based on the relevance between user embedding and candidate news embedding. We assume a user has T historical clicked news, which are denoted as $[D_1, D_2, ..., D_T]$. The news encoder processes these clicked news of a user and each candidate news D_c to obtain their embeddings, which are denoted as $[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_T]$ and $\mathbf{h}_c$, respectively. It can be implemented by various NLP models, such as CNN [10] and self-attention [18]. The user encoder receives the sequence of clicked news embeddings as input, and outputs a user embedding $\mathbf{u}$ that summarizes user interest information. It can also be implemented by various models, such as the GRU network used in [15], the attention network used in [21] and the combination of multi-head self-attention and additive attention networks used in [23]. The click prediction module takes the user embedding $\mathbf{u}$ and $\mathbf{h}_c$ as inputs, and compute the click score $\hat{y}$ by evaluating their relevance. It can also be implemented by various methods such as inner product [15], neural network [20] and factorization machine [7].

2.2 PLM Empowered News Recommendation

Next, we introduce the framework of PLM empowered news recommendation, as shown in Fig. 2. We instantiate the news encoder with a pre-trained language model to capture the deep contexts in news texts and an attention network to pool the output of PLM. We denote an input news text with M tokens as $[w_1, w_2, ..., w_M]$. The PLM converts each token into its embedding, and then learns the hidden representations of words through several Transformer [18]
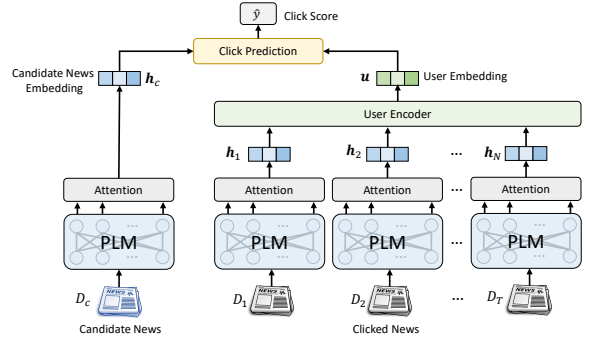
layers. We denote the hidden token representation sequence as $[\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_M]$. We use an attention [29] network to summarize the hidden token representations into a unified news embedding. The news embeddings learned by the PLM and attention network are further used for user modeling and candidate matching.

2.3 Model Training

Following [22, 23], we also use negative sampling techniques to build labeled samples from raw news impression logs, and we use the cross-entropy loss function for model training by classifying which candidate news is clicked. By optimizing the loss function via backward-propagation, the parameters in the recommendation model and PLMs can be tuned for the news recommendation task.

3 EXPERIMENTS

3.1 Datasets and Experimental Settings

Our offline experiments are conducted on two real-world datasets. The first one is MIND [27], which is an English dataset for monolingual news recommendation. It contains the news click logs of 1 million users on Microsoft News in six weeks.[3] The second one is a multilingual news recommendation dataset (denoted as *Multilingual*) collected by ourselves on MSN News platform from Dec. 1, 2020 to Jan. 14, 2021. It contains users from 7 countries with different language usage, and their market language codes are EN-US, DE-DE, FR-FR, IT-IT, JA-JP, ES-ES and KO-KR, respectively. We randomly sample 200,000 impression logs in each market. The logs in the last week are used for test and the rest are used for training and validation (9:1 split). The detailed statistics of the two datasets are shown in Table 1.

Table 1: Detailed statistics of the two datasets.

	MIND	Multilingual
# Users	1,000,000	1,392,531
# News	161,013	4,378,487
# Impressions	15,777,377	1,400,000
# Click Behaviors	24,155,470	1,814,927

---

[3]More details are on https://msnews.github.io/.

In our experiments, we used the "Base" version of different pre-trained language models if not specially mentioned. We finetuned the last two Transformer layers because we find there is only a very small performance difference between finetuning all layers and the last two layers. Following [27], we used the titles of news for news modeling. We used Adam [3] as the optimization algorithm and the learning rate was 1e-5. The batch size was 128.[4] These hyperparameters are developed on the validation sets. We used average AUC, MRR, nDCG@5 and nDCG@10 over all impressions as the performance metrics. We repeated each experiment 5 times independently and reported the average performance.

3.2 Offline Performance Evaluation

We first compare the performance of several methods on the *MIND* dataset to validate the effectiveness of PLM-based models in mono-lingual news recommendation. We compared several recent news recommendation methods including EBNR [15], NAML [21], NPA [22], LSTUR [1], NRMS [23] and their variants empowered by different pre-trained language models, including BERT [5], RoBERTa [14] and UniLM [2]. The results are shown in Table 2. Referring to this table, we find that incorporating pre-trained language models can consistently improve the performance of basic models.[5] This is because pre-trained language models have stronger text model-ing ability than the shallow models learned from scratch in the news recommendation. In addition, we find that the models based on RoBERTa are better than those based on BERT. This may be because RoBERTa has better hyperparameter settings than BERT and is pre-trained on larger corpus for a longer time. Besides, the models based on UniLM achieve the best performance. This may be due to UniLM can exploit the self-supervision information in both text understanding and generation tasks, which can help learn a higher-quality PLM.

In addition, we conduct experiments on the *Multilingual* dataset to validate the effectiveness of PLMs in multilingual news rec-ommendation. We compare the performance of EBNR, NAML, NPA, LSTUR and NRMS with different multilingual text modeling methods, including: (1) MUSE [13], using modularizing unsuper-vised sense embeddings; (2) Unicoder [8], a universal language encoder pre-trained by cross-lingual self-supervision tasks; and (3) InfoXLM [4], a contrastively pre-trained cross-lingual language model based on information-theoretic framework. In these methods, following [8] we mix up the training data in different languages. In addition, we also compare the performance of independently learned monolingual models based on MUSE for each market (de-noted as Single). The results of different methods in terms of AUC are shown in Table 3. We find that multilingual models usually outperform the independently learned monolingual models. This may be because different languages usually have some inherent relatedness and users in different countries may also have some sim-ilar interests. Thus, jointly training models with multilingual data can help learn a more accurate recommendation model. It also pro-vides the potential to use a unified recommendation model to serve users in different countries with diverse language usage (e.g., Indo-European and Altaic), which can greatly reduce the computation

[4]We used 4 GPUs in parallel and the batch size on each of them was 32.
[5]The results of t-test show the improvements are significant ($p < 0.001$).

Table 2: Performance of different methods on *MIND*.

Methods	AUC	MRR	nDCG@5	nDCG@10
EBNR	66.54	32.43	35.38	40.09
EBNR-BERT	69.56	34.77	38.04	43.72
EBNR-RoBERTa	69.70	34.84	38.21	43.88
EBNR-UniLM	70.56	35.31	38.65	44.32
NAML	67.78	33.24	36.19	41.95
NAML-BERT	69.42	34.66	37.91	43.65
NAML-RoBERTa	69.60	34.78	38.13	43.79
NAML-UniLM	70.50	35.26	38.60	44.27
NPA	67.87	33.20	36.26	42.03
NPA-BERT	69.50	34.72	37.96	43.72
NPA-RoBERTa	69.64	34.81	38.14	43.82
NPA-UniLM	70.52	35.29	38.63	44.29
LSTUR	68.04	33.31	36.28	42.10
LSTUR-BERT	69.49	34.72	37.97	43.70
LSTUR-RoBERTa	69.62	34.80	38.15	43.79
LSTUR-UniLM	70.56	35.29	38.67	44.31
NRMS	68.18	33.29	36.31	42.20
NRMS-BERT	69.50	34.75	37.99	43.72
NRMS-RoBERTa	69.56	34.81	38.05	43.79
NRMS-UniLM	**70.64**	**35.39**	**38.71**	**44.38**

Table 3: Performance of different methods on *Multilingual*.

Methods	EN-US	DE-DE	FR-FR	IT-IT	JA-JP	ES-ES	KO-KR
EBNR-Single	62.08	59.94	61.66	60.27	61.57	58.30	63.53
EBNR-MUSE	62.26	60.19	61.75	60.44	61.74	57.53	63.78
EBNR-Unicoder	63.35	61.44	62.34	61.18	62.76	58.70	64.80
EBNR-InfoXLM	64.29	62.03	62.97	61.98	63.34	59.33	65.58
NAML-Single	62.05	59.89	61.56	60.21	61.54	58.21	63.5
NAML-MUSE	62.17	60.17	61.71	60.4	61.69	57.46	63.73
NAML-Unicoder	63.3	61.37	62.32	61.16	62.74	58.61	64.77
NAML-InfoXLM	64.27	61.98	62.94	61.91	63.29	59.33	65.49
NPA-Single	62.09	59.90	61.56	60.24	61.57	58.24	63.56
NPA-MUSE	62.23	60.21	61.78	60.44	61.75	57.47	63.71
NPA-Unicoder	63.32	61.41	62.35	61.20	62.77	58.64	64.80
NPA-InfoXLM	64.29	62.00	62.93	61.94	63.31	59.37	65.50
LSTUR-Single	62.09	59.95	61.58	60.22	61.58	58.22	63.57
LSTUR-MUSE	62.21	60.21	61.79	60.44	61.73	57.49	63.75
LSTUR-Unicoder	63.34	61.40	62.36	61.20	62.77	58.65	64.80
LSTUR-InfoXLM	64.31	62.03	62.96	61.95	63.32	59.38	65.54
NRMS-Single	62.11	59.94	61.62	60.28	61.57	58.30	63.64
NRMS-MUSE	62.33	60.29	61.86	60.54	61.90	57.62	63.93
NRMS-Unicoder	63.41	61.50	62.46	61.22	62.81	58.84	64.79
NRMS-InfoXLM	**64.34**	**62.05**	**63.04**	**61.98**	**63.40**	**59.44**	**65.58**

and memory cost of online serving. In addition, the performance methods based on multilingual PLMs are better than those based on MUSE embeddings. This may be because PLMs are also stronger than word embeddings in capturing the complicated multilingual semantic information. In addition, InfoXLM can better empower multilingual news recommendation than Unicoder. This may be because InfoXLM uses better contrastive pre-training strategies than Unicoder to help learn more accurate models.

3.3 Influence of Model Size

Next, we explore the influence of PLM size on the recommendation performance. We compare the performance of two representative
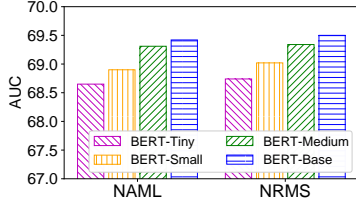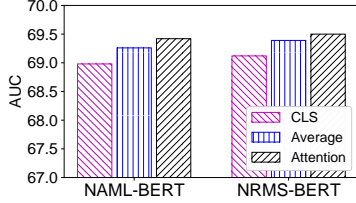
Figure 3: Influence of the size of PLMs.

Figure 4: Influence of the pooling methods.

methods (i.e., NAML and NRMS) with different versions of BERT, including BERT-Base (12 layers), BERT-Medium (8 layers), BERT-Small (4 layers) and BERT-Tiny (2 layers). The results on *MIND* are shown in Fig. 3. We find that using larger PLMs with more parameters usually yields better recommendation performance. This may be because larger PLMs usually have stronger abilities in capturing the deep semantic information of news, and the performance may be further improved if more giant PLMs (e.g., BERT-Large) are incorporated. However, since huge PLMs are too cumbersome for online applications, we prefer the base version of PLMs.

3.4 Influence of Different Pooling Methods

We also explore using different pooling methods for learning news embeddings from the hidden states of PLMs. We compare three methods, including: (1) *CLS*, using the representation of the "[CLS]" token as news embedding, which is a widely used method for obtaining sentence embedding; (2) *Average*, using the average of hidden states of PLM; (3) *Attention*, using an attention network to learn news embeddings from hidden states. The results of NAML-BERT and NRMS-BERT on *MIND* are shown in Fig. 4.[6] We find it is very interesting that the *CLS* method yields the worst performance. This may be because it cannot exploit all output hidden states of the PLM. In addition, *Attention* outperforms *Average*. This may be because attention networks can distinguish the informativeness of hidden states, which can help learn more accurate news representations. Thus, we choose attention mechanism as the pooling method.

3.5 Visualization of News Embedding

We also study the differences between the news embeddings learned by shallow models and PLM-empowered models. We use t-SNE [17] to visualize the news embeddings learned by NRMS and NRMS-UniLM, and the results are shown in Fig. 5. We find an interesting phenomenon that the news embeddings learned by NRMS-UniLM are much more discriminative than NRMS. This may be because the

---

[6]We observe similar phenomenons in other PLM-based methods.

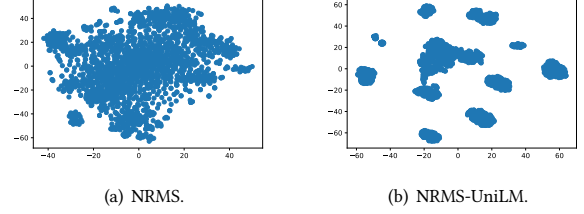(a) NRMS.   (b) NRMS-UniLM.

Figure 5: Visualization of news embeddings learned by NRMS and NRMS-UniLM.

shallow self-attention network in NRMS cannot effectively model the semantic information in news texts. Since user interests are also inferred from embeddings of clicked news, it is difficult for NRMS to accurately model user interests from non-discriminative news representations. In addition, we observe that the news embeddings learned by NRMS-UniLM form several clear clusters. This may be because the PLM-empowered model can disentangle different kinds of news for better user interest modeling and news matching. These results demonstrate that deep PLMs have greater ability than shallow NLP models in learning discriminative text representations, which is usually beneficial for accurate news recommendation.

3.6 Online Flight Experiments

We have deployed our PLM-empowered news recommendation models into the Microsoft News platform. Our NAML-UniLM model was used to serve users in English-speaking markets, including EN-US, EN-GB, EN-AU, EN-CA and EN-IN. The online flight experimental results have shown a gain of 8.53% in click and 2.63% in pageview against the previous news recommendation model without pre-trained language model. In addition, our NAML-InfoXLM model was used to serve users in other 43 markets with different languages. The online flight results show an improvement of 10.68% in click and 6.04% in pageview. These results validate that incorporating pre-trained language models into news recommendation can effectively improve the recommendation performance and user experience of online news services.

4 CONCLUSION

In this paper, we present our work on empowering personalized news recommendation with pre-trained language models. We conduct extensive offline experiments on both English and multilingual news recommendation datasets, and the results show incorporating pre-trained language models can effectively improve news modeling for news recommendation. In addition, our PLM-empowered news recommendation models have been deployed to a commercial news platform, which is the first public reported effort to empower real-world large-scale news recommender systems with PLMs. The online flight results show significant improvement in both click and pageview in a large number of markets with different languages.

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] Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *ICML*. PMLR, 642–652.

[3] Yoshua Bengio and Yann LeCun. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[4] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training. *arXiv preprint arXiv:2007.07834* (2020).

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. 4171–4186.

[6] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified Language Model Pre-training for Natural Language Understanding and Generation. In *NeurIPS*. 13042–13054.

[7] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *AAAI*. 1725–1731.

[8] Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks. In *EMNLP-IJCNLP*. 2485–2494.

[9] Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš Mikolov. 2017. Bag of Tricks for Efficient Text Classification. In *EACL*. 427–431.

[10] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP*. 1746–1751.

[11] Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291* (2019).

[12] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *ICLR*.

[13] Guang-He Lee and Yun-Nung Chen. 2017. MUSE: Modularizing Unsupervised Sense Embeddings. In *EMNLP*. 327–337.

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

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

[16] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences* (2020), 1–26.

[17] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *JMLR* 9, 11 (2008).

[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*. 5998–6008.

[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. In *IJCAI*. 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, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019. Neural News Recommendation with Multi-Head Self-Attention. In *EMNLP*. 6390–6395.

[24] Chuhan Wu, Fangzhao Wu, Junxin Liu, Sixing Wu, Yongfeng Huang, and Xing Xie. 2018. Detecting tweets mentioning drug name and adverse drug reaction with hierarchical tweet representation and multi-head self-attention. In *SMM4H*. 34–37.

[25] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020. SentiRec: Sentiment Diversity-aware Neural News Recommendation. In *AACL*. 44–53.

[26] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020. User Modeling with Click Preference and Reading Satisfaction for News Recommendation. In *IJCAI*. 3023–3029.

[27] 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.

[28] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *NeurIPS* 32 (2019), 5753–5763.

[29] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL-HLT*. 1480–1489.