Table 1: Statistics of popular recommendation datasets. K and M are short for thousand and million respectively. Type_Feeds denotes the number of types of positive user feedback. True_neg denotes whether it includes true negative feedback. We only show the statistics of the QK-video (QKV) and QK-article (QKA) in this table. #Interactions in Tenrec denotes the clicking behavior.

Dataset	Domains	Type_Feeds	#Users	#Interactions	True_neg
Movielens-20M[1]	Single	1	138K	20M	✗
Amazon[2]	Multiple	2	/	233M	✗
Yelp[3]	Single	1	1.9M	8M	✗
YOOCHOOSE[4]	Single	2	9.2M	34M	✗
Taobao: User-Behavior[5]	Single	4	987K	100M	✗
Ali_Display_Ad_Click[6]	Single	4	1.1M	26M	✓
TMALL[7]	Single	2	963K	44M	✓
Yahoo! Music[8]	Single	1	1.9M	11M	✗
Book-Crossing[9]	Single	1	92K	1.0M	✗
MIND[10]	Single	1	1.0M	24M	✗
KuaiRec[11]	Single	1	7K	12M	✓
ZhihuRec[12]	Single	1	798K	99M	✓
Tenrec-QKV	Multiple	4	5.0 M	142M	✓
Tenrec-QKA	Multiple	6	1.3 M	46M	✗

Appendix

A Dataset Comparison

We show the difference between Tenrec and other popular recommendation datasets in Table1. First, most datasets contain only a single scenario. Without overlapped users and items, it is difficult to develop and evaluate transfer learning recommendation methods. In addition, Tenrec contains very rich positive user feedback, which can be used to evaluate the multi-task learning and preference-level transfer learning tasks. Third, compared with most recommendation datasets, Tenrec has true negative examples, which can be used to evaluate more realistic CTR prediction task.

It is worth mentioning that **the multiple domains in Amazon are defined differently from Tenrec**. In our Tenrec, items of different domains are either from different recommender systems or recommended by completely different algorithms. However, domains in Amazon are divided simply based on their item categories. It is unknown whether items of different categories are recommended by the same or different algorithms. **It is not suitable to be used for the cross domain recommendation tasks if items are recommended by the same model and from the same platform.** In fact, our Tenrec-QKA also includes many different article categories.

B Related Work

B.1 Existing Datasets

There are some typical datasets in the recommender system field, which have played a key role in promoting the development of the recommender system community. But these datasets are either small in scale or have limited forms of user feedback, so it is difficult for these datasets to provide enough value for large-scale real scenarios. Movielens datasets [13] such as ML-100K, ML-1M and ML-20M have become the typical datasets for recommender system. But the largest ML-20M dataset has only a single user feedback, and it contains 138K users and 20M interactions. Seri Choi et al constructed Yelp dataset [1] for businesses recommmendation, which has 1.9M users and 8M

[13] https://grouplens.org/datasets/movielens/

interactions. Taobao: User-Behavior [14], Ali_Display_Ad_Click [15] and TMALL [16] have a variety of user feedback, but are single-domain datasets. Yahoo!music [2] for song recommendation has 1.9M users and 11M interactions. A news recommendation dataset named MIND [3] was constructed by Fangzhao Wu et al. It contains 1M users, 161K news, 24M clicks and news information. KuaiRec [4] is a dataset with a sparsity of 99.6% and contains rich features, but the size of the dataset is small and only contains 7k users and 12M interactions. Bin Hao et al [5] released ZhihuRec dataset collected from knowledge sharing platform, it contains 798K users and 99M interactions and text features etc.

B.2 Existing Benchmarks

At present, there are some popular benchmarks in the recommender system community. To be specific, Weichen Shen et al developed DeepCTR [6] for CTR prediction tasks and implemented a variety of classic CTR models. Recbole [7] is released for recommender system tasks such as sequential recommendation, context-aware recommendation, knowledge-based recommendation, etc. And it integrates multiple recommender system models and datasets. Similar benchmark tools include DaisyRec [8], FuxiCTR [9], EasyRec [10], etc. Different from the above-mentioned benchmarks, BARS [11] reports the performance of a large number of recommender system models on multiple datasets and provides reproducible scripts. In order to better display the Tenrec dataset, we also constructed our own benchmark containing more than ten recommender system tasks. For each task, there are many baselines, however, we mainly report a few representative ones and leave more evaluations for the community. We would keep updating leaderboards and build a new one if neessary. Feel free to email Fajie & Guanghu if you want to launch a new leaderboard for an important RS task using Tenrec.

C Supplementary experiment

In the main body, we only report results with randomly sampled 1 million users, here we show results of the CTR (Table 2) prediction and SBR (Table 3) tasks with 5 million users on the full QK-video dataset, following the same experimental setup. For each task, we report several top ranked baselines in the main body.

In addition to the above experiments, we supplement the experiments of shared historical embedding (i.e., all interacted items share the same embedding) in the CTR prediction task. As show in Table 4, we could make two observations: (1) CTR models with the shared historical embedding in general slightly underperform models with separate historical embedding (SHE); (2) CTR models with shared historical embedding show similar accuracy rank as previously reported in Table 2 with SHE.

We also add another experiments with more cold-start settings. To be specific, we notice in some practical recommendation scenarios where both cold and warm users co-exist. To create such a scenario, we first draw overlapped users between QK-video and QK-article. Then we randomly sample n% users (e.g., $n = 30, 70, 100$) and then select the latest k interacted items of them where k is a random integer from 1 to 5, ensuring that these users are cold. The behaviors of the remaining warm users are kept the same. For training, we use all behaviors of these warm users and 50% behaviors from the cold users. For evaluation, we only evaluate the predictive accuracy for these cold users with 25% interactions for validation and 25% for testing. Results are reported in Table 7.

Here, we report baseline results for the standard top-N item recommendation task on QB-video. We implemented key baselines by referring to the official code, code of DaisyRec[17]. We filter out users with session length shorter than 10. Then, we split interactions of each user into 8:1:1 as the training set, validation set, and testing set. We evaluate four popular baseline: MF [12], NCF [13], NGCF [14], LightGCN [15] to verify Tenrec. The hyper-parameters are searched similarly as before.

[14] https://tianchi.aliyun.com/dataset/dataDetail?dataId=649

[15] https://tianchi.aliyun.com/dataset/dataDetail?dataId=56

[16] https://tianchi.aliyun.com/dataset/dataDetail?dataId=53

[17] https://github.com/recsys-benchmark/DaisyRec-v2.0

Table 2: Results for CTR prediction.

Model	AUC	Logloss
Wide & Deep	0.8234	0.4745
DeepFM	0.8235	0.4741
NFM	0.8231	0.4750
xDeepFM	0.8235	0.4740
AFM	0.8226	0.4757

Table 3: Results for SBR.

Model	HR@20	NDCG@20
NextItNet	0.05490	0.0214
SASRec	0.05164	0.0201
BERT4Rec	0.05027	0.0191

Learning rate is set to $5e{-}4$ for NGCF, $1e{-}6$ for NCF, $5e{-}3$ for MF and LightGCN. Batch size is set to 4096 for all models. The embedding size is set to 128 for all models. Then the layer number is set to 2 for NCF, NGCF and LightGCN. We show results using two types of negative samplers: random sampler and popularity sampler used in word2vec [16] with power set to 0.75. The number of negative examples is set to 4 for each user. All results are reported in Table 5 and Table 6. It is worth mentioning that more powerful negative samplers could easily lead to better recommendation accuracy than the random and popularity samplers, e.g. the two dynamic samplers used in LambdaFM [17]. In other words, if you want to compare network architectures, you should ensure that all other settings (loss function, sampling ratio and distribution) are kept the same for comparison.

For other tasks, we would create per-task leaderboards for the full dataset version and the 1 million user version.

Table 4: Results for CTR prediction with shared historical embeddings on QK-video-1M.

Model	AUC	Logloss
Wide & Deep	0.7910	0.5111
DeepFM	0.7920	0.5105
NFM	0.7924	0.5094
xDeepFM	0.7922	0.5092
AFM	0.7921	0.5097
DCN	0.7911	0.5100
DCNv2	0.7922	0.5097
DIN [18]	0.7910	0.5110
DIEN [19]	0.7918	0.5108

Table 5: Results of top-n item recommendation with the random negative sampler.

Model	Recall@20	NDCG@20
MF	0.0838	0.0437
NCF	0.0764	0.0403
LightGCN	0.1065	0.0542
NGCF	0.0878	0.0455

Table 6: Results of top-n item recommendation with the popularity negative sampler.

Model	Recall@20	NDCG@20
MF	0.0927	0.0467
NCF	0.0757	0.0405
LightGCN	0.1211	0.0617
NGCF	0.0948	0.0476

D General Datasheet of Dataset

D.1 Motivation

For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

To foster diverse recommendation research, we propose Tenrec, a large-scale and multipurpose real-world dataset. Compared with existing public datasets, Tenrec has several merits: (1) it consists of overlapped users/items from four different real-world recommendation scenarios, which can be used to study the cross-domain recommendation (CDR) and transfer learning (TF) methods; (2) it contains multiple types of positive user feedback (e.g. clicks, likes, shares, follows, reads and favorites), which can be leveraged to study the multi-task learning (MTL) problem; (3) it has both positive user

Table 7: Results for cold start prediction with different percentages of cold users. E.g. $cold\_rate$ 0.3 means that the percentage of cold users in the training data accounts for 30% of the overlapped users (including both cold and warm users). NDCG@20 is the evaluation metric.

Model	cold_rate 0.3	cold_rate 0.7	cold_rate 1
PeterRec w/o PT	0.0112	0.0123	0.0158
PeterRec w/ PT	0.0133	0.0132	0.0165
BERT4Rec w/o PT	0.0115	0.0119	0.0153
BERT4Rec w/ PT	0.0137	0.0134	0.0166

feedback and true negative feedback, which can be used to study more practical click-through rate (CTR) prediction scenario; (4) it has additional user and item features beyond the identity information (i.e. user IDs and item IDs), which can be used for context/content-based recommendations.

Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

The dataset was created by Guanghu Yuan and Beibei Kong who were an intern and employee respectively at Tencent.

Who funded the creation of the dataset? If there is an associated grant, please provide the name of the grantor and the grant name and number.

No.

D.2 Composition

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

The instances are user feedback collected from two different feeds recommendation platforms of Tencent, including both positive feedback (i.e. video click, share, like and follow) and negative feedback (with exposure but no user action).

How many instances are there in total (of each type, if appropriate)?

There are 493,458,970 instances in QK(QQ Kandian) Video Datset, 11,722,249 instances in QB(QQ Browser) Video Datset, 46,111,728 instances in QK(QQ Kandian) Article Dataset, and 348,736 instances samples in QB(QQ Browser) Article Dataset, where each sample is user-item interactions.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

The dataset is a sample of instances. we randomly draw instances from two different feeds recommendation platforms of Tencent, with the requirement that each user had at least 5 video clicking behaviors. No tests were run to determine representativeness.

What data does each instance consist of? "Raw" data (e.g., unprocessed text or images)or features? In either case, please provide a description.

The format of each instance in QK/QB-video is {*user ID, item ID, click, like, share, follow, video category, watching times, user gender, user age, timestamp*}. *click, like, share, follow* are binary values denoting whether the user has such an action. *watching times* is the number of watching behaviors on the video. *user ID, item ID, user gender, user age and timestamp* have been desensitized for privacy issues. *User age* has been split into bins, with each bin representing a 10-year period.

The format of each instance in QK/QB-article is {*user ID, item ID, click, like, share, follow, read, favorite, click_count, like_count, comment_count, exposure_count, read_percentage, category_second, category_first, item_score1, item_score2, item_score3, read_time, timestamp*}. The suffix "*∗_count*" denotes the total number of ∗ actions per article. *read_percentage* denotes how much percentage the user has read the article, with value ranging from 0 to 100. *category_first* and *category_second* are categories of the article, where "*_first*" is the coarse-grained category (e.g. sports, entertainment, military, etc) and "*_second*" is the fine-grained category (e.g. NBA, World Cup, Kobe, etc.). *item_score1, item_score2, item_score3* denote the quality of the item by different scoring system. *read_time* is the duration of reading.

Is there a label or target associated with each instance? If so, please provide a description.

The labels are binary values denoting whether the user has such an action, or user profile.

Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

A small percentage of instances lack video category, user age and user gender. The corresponding information is missing in the real system.

Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.

we split the data into 8:1:1 as the training set, validation set, and testing set following some common practice.

Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description.

No

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a dataset consumer? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

The dataset is entirely self-contained.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor–patient confidentiality, data that includes the content of individuals' nonpublic communications)? If so, please provide a description.

No

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.

No

Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

No

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? If so, please describe how.

It is impossible to identify individuals from the dataset information

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals race or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? If so, please provide a description

No

D.3 Collection Process

How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If the data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

The data was mostly observable from user feedback on feeds recommendation platform of Tencent.

What mechanisms or procedures were used to collect the data (e.g., hardware apparatuses or sensors, manual human curation, software programs, software APIs)?How were these mechanisms or procedures validated?

Unknown to the authors of the datasheet.

If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?

we randomly draw users from the database, with the requirement that each user had at least 5 video clicking behaviors.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?

Unknown to the authors of the datasheet.

Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created.

We collect user behavior logs from QK/QB from September 17 to December 07, 2021.

Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?

The data was collected from feeds recommendation platform of Tencent.

D.4 Preprocessing/cleaning/labeling

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remaining questions in this section.

We anonymize user ID and item ID to protect user privacy. User profiles are also processed into discrete or binary values.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.

No

D.5 Uses

Has the dataset been used for any tasks already? If so, please provide a description.

No

Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.

Yes.

What (other) tasks could the dataset be used for?

The dataset can be used for CTR Prediction, Session-based Recommendation, Mutli-task Learning for Recommendation, Transfer Learning for Recommendation, User Profile Prediction, Lifelong User Representation Learning, Cold-start Recommendation, Model Compression, Model Training Speedup and Model inference Speedup. See our paper for details.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a dataset consumer might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other risks or harms (e.g., legal risks, financial harms)? If so, please provide a description. Is there anything a dataset consumer could do to mitigate these risks or harms?

There is little risk here when we have anonymized the dataset.

D.6 Distribution

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created? If so, please provide a description.

The dataset is publicly available on the internet.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub)? Does the dataset have a digital object identifier (DOI)?

The distribution of the dataset is detailed in our paper.

When will the dataset be distributed?

The dataset will be distributed in June 2022.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

This dataset is licensed under a CC BY-NC 4.0 International License(`https://creativecommons.org/licenses/by-nc/4.0/`). There is a request to cite the corresponding paper if the dataset is used.

Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.

No

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.

Unknown to authors of the datasset

D.7 Maintenance

Who will be supporting/hosting/maintaining the dataset?

Guangnhu Yuan and Fajie Yuan are supporting/maintaining the dataset.

How can the owner/curator/manager of the dataset be contacted (e.g., email address)?

Guanghu Yuan and Fajie Yuan can be contacted at gh.yuan0@gmail.com and yuanfajie@westlake.edu.cn, respectively.

Is there an erratum? If so, please provide a link or other access point.

Not yet found.

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to dataset consumers (e.g., mailing list, GitHub)?

This will be posted on the datasset webpage.

Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to dataset consumers.

We do not maintain old versions of the dataset, if we update the version of the dataset, we will put the specific details of the dataset update on the relevant GitHub.

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to dataset consumers? If so, please provide a description.

If others want to extend/augment/build on/contribute to the dataset, please contact the original authors about incorporating fixes/extensions.

References

[1] Seri Choi et al. *An empirical study identifying bias in Yelp dataset*. PhD thesis, Massachusetts Institute of Technology, 2021.

[2] Gideon Dror, Noam Koenigstein, Yehuda Koren, and Markus Weimer. The yahoo! music dataset and kdd-cup'11. In *Proceedings of KDD Cup 2011*, pages 3–18. PMLR, 2012.

[3] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3597–3606, 2020.

[4] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. Kuairec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 540–550, 2022.

[5] Bin Hao, Min Zhang, Weizhi Ma, Shaoyun Shi, Xinxing Yu, Houzhi Shan, Yiqun Liu, and Shaoping Ma. A large-scale rich context query and recommendation dataset in online knowledge-sharing. *arXiv preprint arXiv:2106.06467*, 2021.

[6] Weichen Shen. Deepctr: Easy-to-use, modular and extendible package of deep-learning based ctr models. *GitHub Repository*, 2018.

[7] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Kaiyuan Li, Yushuo Chen, Yujie Lu, Hui Wang, Changxin Tian, Xingyu Pan, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. *arXiv preprint arXiv:2011.01731*, 2020.

[8] Zhu Sun, Hui Fang, Jie Yang, Xinghua Qu, Hongyang Liu, Di Yu, Yew-Soon Ong, and Jie Zhang. Daisyrec 2.0: Benchmarking recommendation for rigorous evaluation. *arXiv preprint arXiv:2206.10848*, 2022.

[9] Jieming Zhu, Jinyang Liu, Shuai Yang, Qi Zhang, and Xiuqiang He. Open benchmarking for click-through rate prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 2759–2769, 2021.

[10] Mengli Cheng, Yue Gao, Guoqiang Liu, HongSheng Jin, and Xiaowen Zhang. Easyrec: An easy-to-use, extendable and efficient framework for building industrial recommendation systems. *arXiv preprint arXiv:2209.12766*, 2022.

[11] Jieming Zhu, Kelong Mao, Quanyu Dai, Liangcai Su, Rong Ma, Jinyang Liu, Guohao Cai, Zhicheng Dou, Xi Xiao, and Rui Zhang. Bars: Towards open benchmarking for recommender systems. *arXiv preprint arXiv:2205.09626*, 2022.

[12] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[13] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

[14] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019.

[15] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.

[16] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[17] Fajie Yuan, Guibing Guo, Joemon M Jose, Long Chen, Haitao Yu, and Weinan Zhang. Lambdafm: learning optimal ranking with factorization machines using lambda surrogates. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 227–236, 2016.

[18] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1059–1068, 2018.

[19] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5941–5948, 2019.