Internal and Contextual Attention Network for Cold-start Multi-channel Matching in Recommendation

Ruobing Xie[*] , **Zhijie Qiu**[*] , **Jun Rao** , **Yi Liu** , **Bo Zhang** and **Leyu Lin**
WeChat Search Application Department, Tencent, China
ruobingxie@tencent.com

Abstract

Real-world integrated personalized recommendation systems usually deal with millions of heterogeneous items. It is extremely challenging to conduct full corpus retrieval with complicated models due to the tremendous computation costs. Hence, most large-scale recommendation systems consist of two modules: a multi-channel matching module to efficiently retrieve a small subset of candidates, and a ranking module for precise personalized recommendation. However, multi-channel matching usually suffers from cold-start problems when adding new channels or new data sources. To solve this issue, we propose a novel Internal and contextual attention network (ICAN), which highlights channel-specific contextual information and feature field interactions between multiple channels. In experiments, we conduct both offline and online evaluations with case studies on a real-world integrated recommendation system. The significant improvements confirm the effectiveness and robustness of ICAN, especially for cold-start channels. Currently, ICAN has been deployed on WeChat Top Stories used by millions of users. The source code can be obtained from https://github.com/zhijieqiu/ICAN.

1 Introduction

Recommendation systems have been widely used for users to get information. Personalized recommendation attempts to predict user preferences of items according to user historical behaviors and profiles, which has been confirmed with different contents, including articles [Okura *et al.*, 2017], videos [Covington *et al.*, 2016] and products [Zhou *et al.*, 2018b].

A real-world integrated personalized recommendation system usually deals with hundreds of millions of heterogeneous items [Wang *et al.*, 2018]. Therefore, it is challenging to conduct sophisticated algorithms to model user-item interactions directly on the entire large corpus, for the linear complexity w.r.t the corpus size is unacceptable [Zhu *et al.*, 2018]. To balance both effectiveness and efficiency, real-world recommendation systems are usually divided into two modules, namely matching (i.e., candidate generation) and ranking [Covington *et al.*, 2016]. The **matching** module aims to retrieve a small subset of (usually hundreds of) items from the entire corpus efficiently, while the **ranking** module aims to rank hundreds of retrieved items precisely with sophisticated models. Fig. 1 gives a real-world integrated personalized recommendation system in WeChat. Moreover, an integrated recommendation system should handle heterogeneous items such as videos and articles, which derive from different data sources and have different features. Therefore, real-world systems usually conduct **multi-channel matching** to retrieve different types of items with different strategies in separate channels (there are nearly dozens of channels used in real-world integrated recommendation). The two-step strategy and multi-channel matching improve the flexibility, robustness and diversification in industry-level recommendation systems.

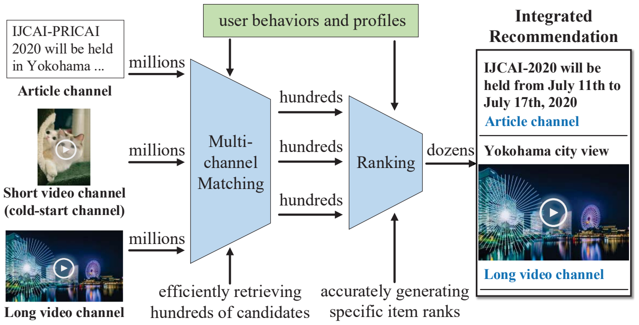
IJCAI-PRICAI 2020 will be held in Yokohama ...
Article channel
Short video channel (cold-start channel)
Long video channel
millions
millions
millions
user behaviors and profiles
Multi-channel Matching
hundreds
hundreds
hundreds
Ranking
dozens
efficiently retrieving hundreds of candidates
accurately generating specific item ranks
Integrated Recommendation
IJCAI-2020 will be held from July 11th to July 17th, 2020
Article channel
Yokohama city view
Long video channel

Figure 1: A real-world integrated personalized recommendation.

However, different matching channels are usually independent of each other, which severely suffer from the cold-start problems when adding channels for new data sources. *Cold-start channel* is one of the most challenging problems, since it has fewer user historical behaviors for precise personalized recommendation. Simply using user behaviors in other *mature channels* is nontrivial, for the heterogeneous items in different channels usually have different features. Most existing works focus on cold-start problems in ranking, while few efforts focus on cold-start channels in the matching module.

In this paper, we concentrate on improving the recommendation performances in the multi-channel matching module especially for cold-start channels. To jointly consider the rich

[*] indicates equal contribution

information in heterogeneous multiple channels, we propose a novel **Internal and contextual attention network (ICAN)** framework to learn user preferences from user historical behaviors in multiple channels and user diverse profiles. Items in different channels have different *feature fields* such as article ID and video tag. In ICAN, we conduct a contextual attention to highlight channel-specific contexts, and also use an internal field-level self-attention to capture informative interactions between feature fields in different channels.

In experiments, we construct a new dataset extracted from a well-known integrated recommendation system *WeChat Top Stories*, collecting nearly *700 million instances from 30 million users*. The significant improvements confirm the effectiveness and robustness of ICAN in both mature and cold-start channels. We also conduct an online A/B test on millions of users to show the power of ICAN in real-world scenarios, with detailed case studies for visualization of two attentions. The main contributions of this work are concluded as follows:

- We first highlight the cold-start channel issue in multi-channel matching, which is essential in real-world integrated personalized recommendation systems.
- We propose a novel and effective ICAN framework to solve this issue. To the best of our knowledge, we are the first to combine the field-level internal attention with the contextual attention in the matching module.
- Both online and offline results on a real-world integrated recommendation system confirm the effectiveness and robustness of ICAN especially for cold-start channels. We have deployed ICAN on WeChat Top Stories.

2 Related Work

Recommendation System Collaborative filtering (CF) is a classical and straightforward method that directly recommends items based on similar items [Sarwar *et al.*, 2001] or users [Breese *et al.*, 1998]. Matrix factorization attempts to decompose user-item interaction matrix to learn user and item representations [Koren *et al.*, 2009]. Factorization machine (FM) [Rendle, 2010] and Field-aware FM [Juan *et al.*, 2016] model second-order feature interactions with the corresponding latent vectors to relieve the data sparsity issue.

With the thriving in deep learning, neural models have been successfully used for CTR prediction in ranking of recommendation. Wide&Deep [Cheng *et al.*, 2016] jointly considers both memorization and generalization abilities with its Wide and Deep architecture. DeepFM [Guo *et al.*, 2017] and NFM [He and Chua, 2017] combine neural FM with DNN, while DCN [Wang *et al.*, 2017] and xDeepFM [Lian *et al.*, 2018] aim to capture high order interactions. AFM [Xiao *et al.*, 2017] and Autoint [Song *et al.*, 2019] conduct attention on such interactions. Most deep ranking models are hard to be adopted to large-scale matching due to their tremendous computation costs over millions of candidates, in which case even linear complexity w.r.t the corpus size is unacceptable.

Matching in Recommendation There are fewer works that focus on matching. Conventional matching modules usually depend on simple information retrieval based (IR-based) models [Khribi *et al.*, 2008], or embedding-similarity based models powered by Item-CF [Sarwar *et al.*, 2001] or FM [Rendle, 2010]. Recently, Youtube [Covington *et al.*, 2016] highlights the two-step architecture widely-used in industry, which brings in deep models to build user embeddings for matching. TDM [Zhu *et al.*, 2018] and JDM [Zhu *et al.*, 2019] store every item in a huge tree structure to fast select approximate top-k most similar items, which combine matching and ranking in a single model. The tree construction is essential in TDM/JDM for retrieving top-k items, which seriously suffers from data sparsity in cold-start channels.

Attention and Cold-start Problems in Recommendation Recent years have witnessed the great successes of attention in ranking of recommendation. DIN [Zhou *et al.*, 2018b] and DIEN [Zhou *et al.*, 2019] introduce attention over user historical behaviors. ATRank [Zhou *et al.*, 2018a] conducts self-attention over user behaviors, and CSAN [Huang *et al.*, 2018] proposes feature-level self-attention for modeling more complicated interactions. For cold-start problems, most works use external information or transfer learning in ranking [Schein *et al.*, 2002; Deldjoo *et al.*, 2019], while few efforts concentrate on cold-start channels in matching. Differing from these models, we consider user behaviors in mature channels to instruct the recommendation in cold-start channels. To the best of our knowledge, we are the first to combine internal and contextual attention in matching for cold-start channels.

3 Methodology

In this paper, we concentrate on improving the performances of multi-channel matching (especially for cold-start channels) in real-world integrated recommendation systems.

3.1 Problem Definition

Integrated recommendation provides heterogeneous items for users. Item candidates from different data sources are usually retrieved by different **channels** (e.g., video/article channels) *separately* in matching with various strategies. These heterogeneous items contain different features, and the same types of features are grouped into **feature fields** (e.g., article ID or video tag) [Guo *et al.*, 2017]. When adding a new data source, the integrated recommendation will generate a new channel to deal with these new items, which is regarded as the **cold-start channel**. Cold-start channels have fewer user behaviors for recommendation, while it is nontrivial to use rich behaviors in other mature channels as supplements. Therefore, we propose ICAN to capture the implicit interactions between d-ifferent feature fields in all channels for the target channel in matching. Note that the cold-start issue locates in the whole new channel, not in the individual users or items.

3.2 Overall Architecture

Fig. 2 demonstrates the overall architecture of ICAN in offline training and online serving. First, to model user historical behaviors, ICAN builds the aggregated embeddings for all feature fields in different channels. Second, a contextual attention is conducted to weight different fields according to the target channel and recommendation contexts. Next, ICAN conducts a field-level self-attention to capture the interactive information between different feature fields. Finally, ICAN combines user historical behaviors with user diverse profiles to form user preference embeddings, which are fed to MLP and softmax layers for CTR prediction. Inspired by Covington *et al.* [2016], we use a fast nearest neighbor server instead of complicated user-item interaction modeling in online serving for efficiency. The time complexity of ICAN is $O(\log n)$ w.r.t the corpus size in online serving.

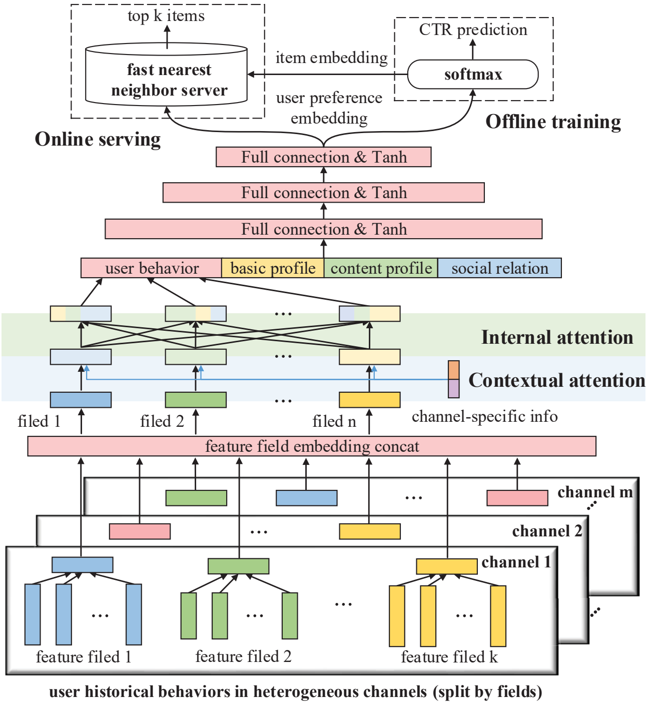

Figure 2: The overall architecture of ICAN for matching.

3.3 Heterogeneous Feature Layer

The heterogeneous feature layer takes user click behaviors in all channels as inputs. In multi-channel matching, we have channels $\{c_1, \cdots, c_{n_c}\}$, where n_c is the channel number of channels. The i-th channel has its feature field set F_i containing n_{f_i} feature fields. We combine all feature fields in these channels sequentially to generate the overall feature field set $F = \{f_1, \cdots, f_{n_f}\}$, where $n_f = \sum_i n_{f_i}$ represents the overall number of feature fields. Differing from conventional session-based recommendation, we divide the clicked item features into several parts, and each part indicates a feature field. For the i-th feature field, $\mathbf{b}_{ij}$ indicates the corresponding field embedding of the j-th user behavior in its channel. The aggregated representation $\mathbf{f}_i$ of the i-th feature fields is calculated by field embeddings in user behaviors as:

$$\mathbf{f}_i = \mathbf{W}_{f_i} \cdot \text{Average\_pooling}(\mathbf{b}_{i1}, \cdots, \mathbf{b}_{ik}), \quad (1)$$

where k is the length of the user historical behavior. $\mathbf{W}_{f_i}$ indicates the projection matrix from each feature field space to the same semantic space. We have tried LSTM to encode behavior sequences, while the performances are just comparable. Considering the computation costs, we use average pooling for building aggregated feature field representations.

3.4 Internal and Contextual Attention Network

To better use these heterogeneous field features, we conduct both contextual and internal attention layers to extract user preferences for recommendation in the target channel.

Contextual Attention Layer

The contextual attention layer aims to highlight the channel-specific contextual information in the target channel. In this paper, the contextual information mainly includes the target channel and the network status (e.g., Wi-Fi or 4G). Specifically, the context embedding $\mathbf{a}$ is the average aggregation of both target channel and network status embeddings, which are randomly initialized and updated during training. We formalize the output embedding $\mathbf{g}_i$ of contextual attention layer with a vanilla attention as follows:

$$\mathbf{g}_i = \alpha_i \mathbf{f}_i, \quad \alpha_i = \frac{\exp(\mathbf{a}^\top \mathbf{f}_i)}{\sum_{j=1}^{n_f} \exp(\mathbf{a}^\top \mathbf{f}_j)}. \quad (2)$$

Internal Attention Layer

The internal attention layer considers the internal interactions between different feature fields. Inspired by the great successes in self-attention [Vaswani *et al.*, 2017], we propose a field-level multi-head self-attention over feature fields. Specifically, we follow the classical self-attention setting of query, key and value, with $\mathbf{q}_i$, $\mathbf{k}_i$ and $\mathbf{v}_i$ representing the corresponding query, key and value of $\mathbf{g}_i$ as:

$$\mathbf{q}_i = \mathbf{W}_q \mathbf{g}_i, \quad \mathbf{k}_i = \mathbf{W}_k \mathbf{g}_i, \quad \mathbf{v}_i = \mathbf{W}_v \mathbf{g}_i. \quad (3)$$

$\mathbf{g}_i$ represents the i-th output feature field embedding of the contextual attention layer. We conduct the self-attention as:

$$\text{Attention}(\mathbf{q}_i, \mathbf{k}_i, \mathbf{v}_i) = \text{softmax}(\frac{\mathbf{q}_i^\top \mathbf{k}_i}{\sqrt{d_k}})\mathbf{v}_i, \quad (4)$$

where $\mathbf{q}_i, \mathbf{k}_i, \mathbf{v}_i \in \mathbb{R}^{d_k}$ and d_k is the dimension of query. We further conduct multi-head self-attention to capture the internal information from different latent subspaces as:

$$\mathbf{h}_i = \text{MultiHead}(\mathbf{q}_i, \mathbf{k}_i, \mathbf{v}_i) = \mathbf{W}_h \cdot \text{concat}(\mathbf{l}_1, \cdots, \mathbf{l}_m), \quad (5)$$

where the output of self-attention $\mathbf{l}_j$ in subspace is as:

$$\mathbf{l}_j = \text{Attention}(\mathbf{W}_{qj}\mathbf{q}_i, \mathbf{W}_{kj}\mathbf{k}_i, \mathbf{W}_{vj}\mathbf{v}_i). \quad (6)$$

$\mathbf{W}_{qj}, \mathbf{W}_{kj}, \mathbf{W}_{vj} \in \mathbb{R}^{d_k \times d_k}$ represent the projection matrices of query, key and value in the j-th subspace respectively. $\mathbf{W}_h \in \mathbb{R}^{d_k \times m d_k}$ indicates the projection matrix of multi-head, where m is the number of heads. Finally, we concatenate all output embeddings of this layer to generate the user behavior embedding $\mathbf{u}_b$ as follows:

$$\mathbf{u}_b = \text{concat}(\mathbf{h}_1, \cdots, \mathbf{h}_{n_f}). \quad (7)$$

Differing from some ranking models like ATRank [Zhou *et al.*, 2018a] and CSAN [Huang *et al.*, 2018] that conduct self-attention over behaviors, ICAN conducts field-level self-attention over different feature fields in all channels. It is because that we attempt to use rich behaviors in mature channels to guild the recommendation in the target cold-start channel through feature field interactions across channels.

3.5 User Preference Representation

The user preference embedding is the combination of user behavior embedding and user profile embeddings. The user profile embedding $\mathbf{u}_p$ consists of three components including user basic profile, user content profile and user social relation, represented as $\mathbf{u}_{\text{basic}}$, $\mathbf{u}_{\text{content}}$ and $\mathbf{u}_{\text{social}}$. The user basic profile is the average embedding of gender, age and geographic embeddings. The user content profile is built according to user's high-frequent tag and category embeddings extracted from their behaviors. The user social relation is a fixed embedding pre-trained on the user social network with DeepWalk [Perozzi *et al.*, 2014]. The user profile embedding $\mathbf{u}_p$ is concatenated by these user profiles as:

$$\mathbf{u}_p = \text{concat}(\mathbf{u}_{\text{basic}}, \mathbf{u}_{\text{content}}, \mathbf{u}_{\text{social}}). \tag{8}$$

Next, we concatenate $\mathbf{u}_b$ and $\mathbf{u}_p$ to generate $\mathbf{u}^{(0)}$ as the input of a 3-layer MLP. For the $(i+1)$-th MLP layer, we have:

$$\mathbf{u}^{(i+1)} = \tanh(\mathbf{W}_o^{(i)}\mathbf{u}^{(i)} + \mathbf{b}_o^{(i)}). \tag{9}$$

The output embedding of the 3rd layer $\mathbf{u}^{(3)}$ indicates the final user preference embedding $\mathbf{u}$, which is used for offline training and online serving. It is also convenient to add other channels or feature fields in our ICAN framework.

3.6 Optimization Objective

We take the training as a CTR prediction task, and use user-click-item behaviors on all channels as positive samples for joint training. The negative samples are randomly sampled from the overall corpus. We represent the i-th item embedding as $\mathbf{e}_i$, and set the positive and negative item sets as y_u^+ and y_u^-. The loss function is formalized as:

$$J = \sum_u \sum_{e_i \in y_u^+} \sum_{e_j \in y_u^-} (\log(\sigma(\mathbf{e}_j^\top \mathbf{u})) - \log(\sigma(\mathbf{e}_i^\top \mathbf{u}))), \tag{10}$$

where $\sigma(\cdot)$ represents the sigmoid function. The optimization objective aims to make the user preference embedding similar to its clicked item embeddings in all channels, which enables fast similarity-based retrieval in online serving.

3.7 Online Serving

The online serving of the matching module in real-world integrated recommendation systems should deal with millions of items. Therefore, it is hard to calculate the exact probabilities of all user-item pairs even with linear time complexity. Inspired by Covington *et al.* [2016], we follow the forward network of ICAN to generate the user preference embedding, and use an approximate nearest neighbor server like FAISS [Johnson *et al.*, 2019] to rank items according to their similarities with the user preference embedding $\mathbf{u}$. The time complexity of ICAN is $O(\log n)$ w.r.t the corpus size in online.

4 Experiments

We evaluate ICAN on both online and offline multi-channel matching in a well-known integrated recommendation system named *WeChat Top Stories*, with detailed analyses and case studies to verify the effectiveness and robustness.

4.1 Datasets

Since there is no large-scale open dataset for multi-channel matching, we construct a new heterogeneous multi-channel matching dataset HMM-700M extracted from WeChat Top Stories. We randomly select nearly *30 million users* and randomly sample nearly *700 million user click behaviors* in three mature and cold-start channels. We use the user behaviors in the first several days as train set, and randomly split the rests into validation and test sets.

Channel	# train	# valid	# test
Article	87,431,593	1,544,321	1,543,314
Long video	587,987,436	7,006,562	7,015,482
Short video	20,043,254	476,432	480,003

Table 1: Statistics of the HMM-700M dataset.

We focus on three representative heterogeneous channels including *article, long video and short video* channels, where the short video channel is the cold-start channel and others are mature channels. Long video channel usually consists of professional videos such as documentary, while short video channel contains homemade portrait-mode videos which are usually less than 30 seconds. We focus on nine typical feature fields including item ID, category and tag of all three channels. ID/category/tag in different channels are viewed as different feature fields. Table 1 shows the detailed statistics.

4.2 Competitors and Our Methods

We implement several classical models as baselines and categorize them into two groups including conventional methods and deep neural methods. We also introduce our ICAN models with two hold-out versions for ablation tests.

Conventional Methods It is intuitive to use information retrieval (IR) based methods [Khribi *et al.*, 2008] for matching. We build the "query" with words of titles in user historical behaviors to retrieve related items in target channel. We also implement Item-CF [Sarwar *et al.*, 2001] that retrieves similar candidates according to clicked items. Moreover, we conduct an enhanced FM model from Rendle [2010] which only considers user-item two-order interactions. Only in this case, the enhanced FM can use embedding-based top-k nearest neighbor servers like Faiss for fast retrieval. These models are classical and efficient methods, which have been widely verified in practice to deal with millions of candidates in matching.

Deep Neural Methods The candidate generation model of Youtube [Covington *et al.*, 2016] is a classical deep-based matching model. To verify the significance of multi-channel information, we use Youtube (Origin) to represent the original Youtube model that only considers user behaviors in the target channel, and use Youtube (Multi) for that with behaviors in multiple channels. We also conduct DeepFM [Guo *et al.*, 2017], NFM [He and Chua, 2017] and AFM [Xiao *et al.*, 2017], which use the same user features in ICAN including user multi-channel behaviors and user diverse profiles to learn user preference embeddings. These three models follow the same training and online serving strategies as ICAN.

Model	Article			Long video			Short video (Cold-start)		
	N=100	N=200	N=500	N=100	N=200	N=500	N=100	N=200	N=500
IR-based	0.0573	0.0763	0.1278	0.0597	0.0854	0.1553	0.0684	0.1048	0.1104
Item-CF	0.0750	0.1108	0.1876	0.0832	0.1205	0.2158	0.0942	0.1432	0.1632
Enhanced FM	0.0942	0.1454	0.2396	0.1293	0.1903	0.3074	0.1828	0.2674	0.3333
Youtube (Origin)	0.0928	0.1434	0.2395	0.1277	0.1882	0.3017	0.1480	0.2200	0.2730
Youtube (Multi)	0.0932	0.1435	0.2396	0.1286	0.1893	0.3028	0.1798	0.2613	0.3214
DeepFM	0.0937	0.1441	0.2392	0.1289	0.1893	0.3043	0.1805	0.2631	0.3236
NFM	0.0939	0.1438	0.2390	0.1284	0.1889	0.3038	0.1811	0.2635	0.3235
AFM	0.0941	0.1448	0.2399	0.1295	0.1906	0.3049	0.1835	0.2669	0.3358
ICAN (w/o Context)	0.0967	0.1492	0.2473	0.1325	0.1948	0.3104	0.1965	0.2912	0.3651
ICAN (w/o Internal)	0.0975	0.1498	0.2498	0.1334	0.1952	0.3128	0.1973	0.2913	0.3710
ICAN	**0.1037**	**0.1574**	**0.2544**	**0.1381**	**0.1985**	**0.3198**	**0.2132**	**0.3054**	**0.3820**

Table 2: Experimental results on HMM-700M. ICAN has consistent improvements especially in cold-start short video channel.

We should clarify that all ranking models which have no less than linear complexity w.r.t the million-level items are unacceptable in matching. Therefore, most deep ranking models such as DIN and ATRank cannot deal with matching task, for they cannot directly use embedding-based fast retrieval. Moreover, we have also tried TDM [Zhu *et al.*, 2018] and find it is not very suitable for cold-start multi-channel matching. The tree construction of TDM is essential for matching, while it is extremely difficult to build a well-learned item tree with sparse behaviors in cold-start channels.

Our ICAN Models We utilize ICAN to represent our final model. To confirm the importance of contextual and internal attention, we also conduct an ablation test, which implements ICAN (w/o Internal) and ICAN (w/o Context) as different ICAN versions without internal or contextual attention.

4.3 Experimental Settings

In ICAN, we utilize 20 most recent click behaviors in each channel as user historical behaviors. The dimensions of the output embeddings in heterogeneous feature, contextual attention and internal attention layers are 64. The dimensions of three user profiles are also 64. The dimension of the output embeddings in 3-layer MLP are 128, 64 and 64. We use Adam for training with the negative sample number to be 20 and the batch size to be 512. All ICAN models and baselines follow the same experimental settings for fair comparisons.

4.4 Offline Evaluation

We evaluate on HMM-700M to verify the capability of ICAN in retrieving appropriate items in the matching module.

Evaluation Protocol ICAN focuses on the matching module that aims to generate *hundreds of* item candidates. Differing from ranking, matching only cares *whether good items are retrieved*, not the specific item ranks. Therefore, we use the hit rate (HIT@N) as our evaluation metric, where an instance is hit if the clicked item is ranked in top N. To simulate the real-world scenarios, we conduct HIT@N where N equals 100, 200 and 500 (we retrieve 500 items in online matching), and report the results in both mature and cold-start channels.

Experimental Results Table 2 demonstrates the evaluation results on HMM-700M, from which we have:

(1) All ICAN models consistently outperform all baselines in both mature and cold-start channels, where ICAN achieves the best performances. We also conduct a significance test to verify that ICAN outperforms baselines with the significance level $\alpha = 0.01$, which confirms the robustness of ICAN. It indicates that considering heterogeneous behaviors in different channels could improve matching performances. It also implies that the internal and contextual attentions could better extract user preferences from all channels compared to other models that also utilize multi-channel features.

(2) ICAN has significant improvements especially in the cold-start channel of short video. It is because that the cold-start channel seriously struggles with insufficient training instances, while ICAN successfully learns user preferences on cold-start channels from user behaviors in mature channels. All models that use multi-channel features such as AFM also outperform single-channel based models on cold-start channel. Moreover, in two mature channels, ICAN still achieves improvements compared to other baselines, which implies the effectiveness of field interactions even in mature channels.

(3) Comparing with different ICAN settings, we find that both contextual and internal attentions play essential roles in multi-channel matching. The contextual attention highlights informative features from different channels that are related to the target channels, while the internal attention considers field-level interactions to bridge the gap between user behaviors in different channels. In case study, we will give detailed visualizations and analyses to reveal the implicit relations between multiple feature fields in different channels.

4.5 Online Evaluation

We further conduct an online A/B test on *WeChat Top Stories* with multiple evaluation metrics, some of which are hard to be measured in offline evaluations.

Evaluation Protocol We have deployed ICAN in matching to compare with the online ensemble single-channel Youtube based matching model, with other modules of our integrated recommendation system unchanged. We focus on the cold-start short video channel with four representative online evaluation metrics, including click-through-rate (CTR), list-wise CTR (LCTR), average reading time per capita (ART) and average watched category number per capita (ACN) as follows:

$$\mathrm{CTR} = \frac{\#\text{ of clicks}}{\#\text{ of impressions}}, \ \mathrm{LCTR} = \frac{\#\text{ of clicked lists}}{\#\text{ of all lists}},$$
$$\mathrm{ART} = \frac{\text{all reading time}}{\#\text{ of users}}, \ \mathrm{ACN} = \frac{\#\text{ of watched cates}}{\#\text{ of users}}. \tag{11}$$

These evaluation metrics can measure both recommendation quality and diversity. We conduct the online evaluation for 7 days with nearly 4.5 million users, and report the improvement percentages instead of the specific values in Table 3.

Metrics	CTR	LCTR	ART	ACN
ICAN	+27.03%	+22.31%	+5.60%	+15.84%

Table 3: The online improvements of ICAN in cold-start channel.

Experimental Results Table 3 demonstrates the online evaluation results, from which we can observe that:

(1) ICAN significantly outperforms the baseline model in all click-related metrics including CTR and LCTR with the significance level $\alpha = 0.01$. CTR measures the recommendation accuracy on individual items, while LCTR concentrates on list-level recommendation accuracy. These improvements confirm that ICAN could benefit the essential point-wise and list-wise click-related performances on real-world systems.

(2) ICAN also has huge improvements in the average reading time per capita with the significance level $\alpha = 0.01$. A higher ART score indicates that users are more interested in the recommended items, and are willing to spend more time on watching clicked videos. It shows the advantages of ICAN from another aspect which can not be evaluated in offline.

(3) The recommendation diversity is also important in real-world recommendation systems. The matching module takes more responsibility for the diversity, since it is supposed to retrieve all possible items that users may be interested in from the overall corpus. We empirically use ACN, which reports the average number of duplicated categories in user's watched videos, to quantify the recommendation diversity. The huge improvement on ACN indicates the power of ICAN in polishing diversity, for ICAN considers more user behaviors from other mature channels that may include user preferences.

4.6 Case Study

Contextual Attention We sum up the contextual attention values of all instances on three channels in test set and normalize them for visualization. Fig. 3 shows the heatmap of the contextual attention. The horizontal axis indicates the feature fields of different channels, while the vertical axis indicates the target channel for matching. We observe that: (1) mature channels (e.g., article) concentrate more on their own feature fields (e.g., article ID, category and tag). It is because that mature channels already have sufficient user behaviors to learn effective matching models. The long video channel also focuses on the feature fields of the short video channel, since they are both videos and share more similarities. (2) The cold-start short video channel is strongly influenced by other feature fields in mature channels such as long video tag. It reconfirms the significance of the contextual attention and feature field interactions in cold-start channels.

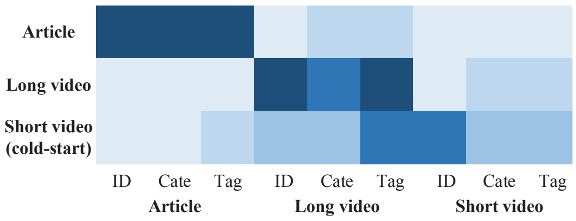

Figure 3: Heatmap of the contextual attention.

Internal Attention We attempt to visualize the multi-head self-attention between feature fields from a user in the short video channel. Most subspaces focus on feature fields in target channels, while Fig. 4 shows two subspaces that capture information from other mature channels. In the left heatmap, the internal attention amplifies the ID and category fields of long video channel when building ID and category embeddings of two video channels. While in the right heatmap, the internal attention highlights the tag and ID fields of article channel when building tag embeddings of two video channels. It shows that the field-level multi-head self-attention could deal with different interactive patterns between feature fields with different heads. Moreover, it also confirms the necessity of considering feature fields separately in ICAN, for they usually behave differently in multi-head feature interactions.

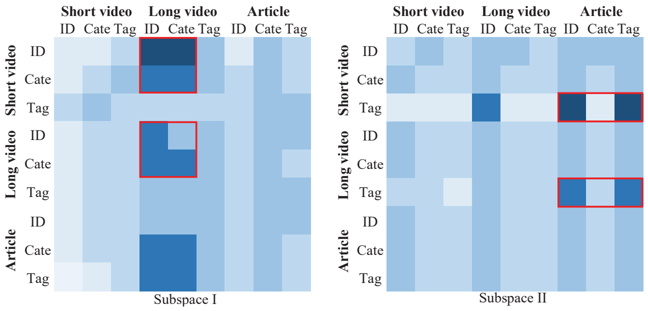

Figure 4: Heatmap of the internal attention.

5 Conclusion and Future Work

In this paper, we propose ICAN for multi-channel matching in real-world integrated recommendation systems, especially for cold-start channels. We conduct internal and contextual attention to extract useful feature field interactions for target channels. Both offline and online evaluations confirm the effectiveness and robustness of ICAN in real-world scenarios.

In future, we will use more sophisticated matching models to learn user preferences with multi-channel features. We will also explore transfer learning, pre-training and other multi-task learning models for cold-start channels.

References

[Breese *et al.*, 1998] John S Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of UAI*, 1998.

[Cheng *et al.*, 2016] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, 2016.

[Covington *et al.*, 2016] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of RecSys*, 2016.

[Deldjoo *et al.*, 2019] Yashar Deldjoo, Maurizio Ferrari Dacrema, Mihai Gabriel Constantin, Hamid Eghbal-Zadeh, Stefano Cereda, Markus Schedl, Bogdan Ionescu, and Paolo Cremonesi. Movie genome: alleviating new item cold start in movie recommendation. *User Modeling and User-Adapted Interaction*, 2019.

[Guo *et al.*, 2017] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. In *Proceedings of IJCAI*, 2017.

[He and Chua, 2017] Xiangnan He and Tat-Seng Chua. Neural factorization machines for sparse predictive analytics. In *Proceedings of SIGIR*, 2017.

[Huang *et al.*, 2018] Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. Csan: Contextual self-attention network for user sequential recommendation. In *Proceedings of ACMMM*, 2018.

[Johnson *et al.*, 2019] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019.

[Juan *et al.*, 2016] Yuchin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. Field-aware factorization machines for ctr prediction. In *Proceedings of RecSys*, 2016.

[Khribi *et al.*, 2008] Mohamed Koutheaïr Khribi, Mohamed Jemni, and Olfa Nasraoui. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. In *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, 2008.

[Koren *et al.*, 2009] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.

[Lian *et al.*, 2018] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of KDD*, 2018.

[Okura *et al.*, 2017] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. Embedding-based news recommendation for millions of users. In *Proceedings of KDD*, 2017.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of KDD*, 2014.

[Rendle, 2010] Steffen Rendle. Factorization machines. In *Proceedings of ICDM*, 2010.

[Sarwar *et al.*, 2001] Badrul Munir Sarwar, George Karypis, Joseph A Konstan, John Riedl, et al. Item-based collaborative filtering recommendation algorithms. In *Proceedings of WWW*, 2001.

[Schein *et al.*, 2002] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of SIGIR*, 2002.

[Song *et al.*, 2019] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. Autoint: Automatic feature interaction learning via self-attentive neural networks. In *Proceedings of CIKM*, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of NIPS*, 2017.

[Wang *et al.*, 2017] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. Deep & cross network for ad click predictions. In *Proceedings of ADKDD*, 2017.

[Wang *et al.*, 2018] Jizhe Wang, Pipei Huang, Huan Zhao, Zhibo Zhang, Binqiang Zhao, and Dik Lun Lee. Billion-scale commodity embedding for e-commerce recommendation in alibaba. In *Proceedings of KDD*, 2018.

[Xiao *et al.*, 2017] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. Attentional factorization machines: Learning the weight of feature interactions via attention networks. In *Proceedings of IJCAI*, 2017.

[Zhou *et al.*, 2018a] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiusi Chen, and Jun Gao. Atrank: An attention-based user behavior modeling framework for recommendation. In *Proceedings of AAAI*, 2018.

[Zhou *et al.*, 2018b] 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 KDD*, 2018.

[Zhou *et al.*, 2019] 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 AAAI*, 2019.

[Zhu *et al.*, 2018] Han Zhu, Xiang Li, Pengye Zhang, Guozheng Li, Jie He, Han Li, and Kun Gai. Learning tree-based deep model for recommender systems. In *Proceedings of KDD*, 2018.

[Zhu *et al.*, 2019] Han Zhu, Daqing Chang, Ziru Xu, Pengye Zhang, Xiang Li, Jie He, Han Li, Jian Xu, and Kun Gai. Joint optimization of tree-based index and deep model for recommender systems. In *Proceedings of NIPS*, 2019.