M5: Multi-Modal Multi-Interest Multi-Scenario Matching for Over-the-Top Recommendation

Pengyu Zhao
Hulu Beijing
Beijing, China
zpysky1125@gmail.com

Xin Gao
Hulu Beijing
Beijing, China
cyngao@163.com

Chunxu Xu
Hulu Beijing
Beijing, China
xucx08@gmail.com

Liang Chen
Hulu Beijing
Beijing, China
liangchen-ms@hotmail.com

ABSTRACT

Matching preferred shows to the subscribers is extremely important in the Over-the-Top (OTT) platforms. The existing methods did not adequately consider the characteristics of the OTT services, i.e., rich meta information, diverse user interests, and mixed recommendation scenarios, leading to sub-optimal performance. This paper introduces the **M**ulti-**M**odal **M**ulti-Interest **M**ulti-Scenario **M**atching (M5) for the OTT recommendation to fully exploit these attributes. A multi-modal embedding layer is first introduced to transform the show IDs into both ID embeddings initialized randomly and content graph (CG) embeddings derived from the node representations pre-trained on a metagraph. To segregate the semantics between ID and CG embeddings, M5 exploits the mirrored two-tower modeling in the subsequent layers for efficiency and effectiveness. Specifically, a multi-interest extraction layer is proposed separately on ID and CG behaviors to model users' coarse-grained and fine-grained interests through behavioral categorization, subsidiary decoration, masked-language-modeling augmented self-attention modeling and subsidiary-intensity interest calibration. Facing the inherent diverse scenarios, M5 distinguishes the scenario differences at both feature and model levels, which crosses features with the scenario indicators and employs Split Mixture-of-Experts to generate the ID, and CG user embeddings. Finally, a weighted candidate matching layer is established to calculate the ID- and CG-oriented user-item preferences and then merge into a hybrid score with dynamic weighting. The extensive online and offline experiments over two real-world OTT platforms Hulu and Disney+ reveal that M5 significantly outperforms the previous state-of-the-art and online matching algorithms over various scenarios, indicating the effectiveness and robustness of the proposed method. M5 has been fully deployed on the main traffic of the most popular "For You" sets of both platforms, continuously enhancing the user experience for hundreds of millions of subscribers every day and steadily increasing business revenue.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
KDD '23, August 6–10, 2023, Long Beach, CA, USA
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0103-0/23/08…$15.00
https://doi.org/10.1145/3580305.3599863

CCS CONCEPTS

• **Information systems** → **Information systems applications**; **Retrieval models and ranking**.

KEYWORDS

recommendation, matching, multi-modal, multi-interest, multi-scenario

ACM Reference Format:
Pengyu Zhao, Xin Gao, Chunxu Xu, and Liang Chen. 2023. M5: Multi-Modal Multi-Interest Multi-Scenario Matching for Over-the-Top Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3580305.3599863

1 INTRODUCTION

With the evolution of network infrastructures and massively growing items, people nowadays spend much time on content platforms such as YouTube, Facebook, TikTok, Netflix, Disney+, etc. **Over-the-Top** (OTT) media services, e.g., Netflix, Disney+, Hulu, which serve hundreds of millions of subscribers every day, have played a prominent role in providing video-on-demand (VOD) series and movies as well as Live streaming from broadcast and cable brands. To help the subscribers find preferred **shows** (playable contents) and navigate the information efficiently and satisfyingly, OTT platforms usually build the recommender systems [10, 44] for delivering personalized items that match user interests, as presented in Fig.1, which is crucial for both user experience and commercial objective.

Modern recommender systems [7, 23] usually follow a multi-stage cascade paradigm of "matching-ranking-reranking". At the bottom of the system, **matching** (retrieval) stage is expected to retrieve a small fraction of relevant items from the candidate pool, which determines the quality of the candidates fed to the subsequent stages and finally presented to the users. Therefore, it plays a fundamental role and usually becomes the bottleneck in the whole system. The early works [8, 27, 33] adopt collaborative filtering (CF) in the matching stage for efficiency and interpretability. Motivated by the developments in deep learning, recent advances [2, 7, 23, 28] propose **two-tower** neural networks, also known as embedding-based retrieval (EBR) [16], to capture complicated feature interactions and improve personalization by leveraging the rich features

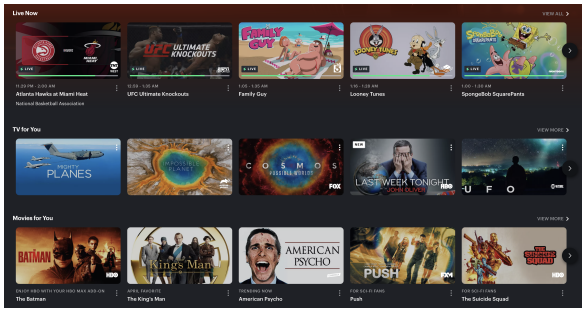


Figure 1: A personalized matrix layout in the OTT platform.

of users and items. It generates individual representations (embeddings) of users and items, then computes the user-item preference through the product. During online serving, approximate nearest neighbors (ANN) or maximum inner product search (MIPS) algorithms [18, 19, 30] are often employed to retrieve relevant items from the corpus with respect to the user embedding. Despite their remarkable success, the above methods do not consistently achieve the expected performance in the OTT platforms due to the lack of consideration on the characteristics described below:

- **Rich meta information**. The videos in the OTT platform usually contain multiple heterogeneous meta information, including ID, tag features (genres, brands), cast features (actor, director), visual features (artwork, video), text features (title, description), rating features, etc. Though existing literature [39, 42] proposes multi-modal fusion in the recommendation, it remains challenging to appropriately and flexibly utilize the multi-modal information along with the complicated model architecture in the OTT matching.
- **Diverse user interests**. The shows in OTT services can be roughly categorized into VOD series, VOD movies, and Live broadcasts. The user behaviors in different categories reflect her (his) various coarse-grained interests in the platform. Moreover, users may also exhibit fine-grained interests within the same category, e.g., a subscriber may watch both documentaries and comedies from the VOD series. To better understand user preferences, OTT matching should explore users' diverse interests in behavior modeling.
- **Mixed recommendation scenarios**. OTT recommendation naturally faces mixed scenarios. From the subscriber side, it needs to serve users with different subscription packages (under varying candidate pools) and regions. From the product side, the platform should present relevant items for different rows (sets) that accord with the themes in the matrix layout, as shown in Fig.1. Since the user behaviors are generally divergent among the various scenarios, adopting a single matching model can obscure scenario differences and bias towards the major scenarios with large traffic while building multiple ones may also be inferior due to ignoring the inter-scenario knowledge sharing. The dilemma makes the design of the multi-scenario matching a tough problem.

Facing those characteristics and challenges, we extend the two-tower architecture and introduce **M5**, namely **M**ulti-**M**odal **M**ulti-Interest **M**ulti-Scenario **M**atching, for the OTT recommendation. The overall architecture of M5 forms the efficient and effective two-channel two-tower modeling, which can be seen as a smooth expansion of the multi-channel retrieval widely spread in the industrial applications [24]. A **multi-modal embedding layer** is proposed at the bottom to transform each show ID into an ID embedding initialized randomly and a content graph (CG) embedding derived from the node representations pre-trained on the metagraph for flexibly utilizing the rich OTT metadata through the upper layers that integrate personalization and contextualization with meta information. On top of that, a **multi-interest extraction layer** is introduced to depict users' diverse interests from historical behaviors at both feature level and model level. Specifically, each behavior sequence is split into VOD series, VOD movies, and Live broadcasts to explicitly differentiate the user interests among the coarse-grained categories, following a subsidiary decoration that appends the attribute features, e.g., "playback duration", to the behavioral embeddings to capture the fine-grained user preferences. For the ID-embedded behaviors, M5 introduces the Masked-Language-Model (MLM) [9] auxiliary loss on the multi-head self-attention [38] to allow for better gradient propagation in modeling users' diverse interests, which dramatically boosts the matching performance. In parallel, a subsidiary-intensity (SIN) module is built for interest calibration by multiplying the attention scores on the behavioral sequences. For the CG embedded behaviors, M5 only applies the SIN module on the raw embeddings without further rectification to preserve the information and item similarity from the metagraph. To deal with the multi-scenario nature, M5 crosses the features with the scenario indicator to distinguish the scenario difference at the feature level in the **multi-scenario mixing layer**, and then designs a Split Mixture-of-Experts (SMoE) [29] augmented by the disagreement loss to discriminate the scenarios at the model level and generate the ID, CG user embeddings. A **weighted candidate matching layer** is finally established to compute the ID- and CG-oriented user-item preferences and then merge them into a unified hybrid user preference via a dynamic weighting network. During online serving, M5 will generate a hybrid user embedding and perform the ANN search on the index built upon the concatenated item embeddings. The broad online and offline experiments over Hulu and Disney+ show that M5 significantly outperforms the state-of-the-art matching models and the current online services, demonstrating the effectiveness, superiority, and robustness of the proposed method in the OTT recommendation. M5 has been fully deployed on both platforms, serving the main traffic of the crucial "For You" sets for hundreds of millions of subscribers every day.

2 RELATED WORK

Industrial Matching. The matching stage is responsible for retrieving the relevant candidates fed to the subsequent stages from the entire corpus, which is essential in the industrial recommendation. The primitive attempts mainly utilize collaborative filtering [27, 33] that calculates the item similarity matrix in advance and then retrieves relevant items based on users' historical behaviors. With the evolution of deep learning, the industry focuses on employing rich

information in the platform to build deep matching models. Among the fruitful success, two-tower architecture [7, 16, 17, 23] has been widely spread in industrial recommendations owing to its simplicity and efficiency, where the user and item features are separately processed in the model, and finally interacted by the product function. Facing the large item corpus in the web-scale system, ANN or MIPS [18, 19, 30] are often adopted to approximate the precise item retrieval during the online serving. This paper studies how to exploit the multi-modal multi-interest multi-scenario characteristics in the OTT platforms to improve industrial matching.

Multi-Modal Recommendation. The content metadata provides generalized information in the recommender system to alleviate the cold-start problem while helping capture the multifaceted interests of the user behaviors. To exploit the meta information in the recommendation, the recent approaches directly utilize the raw multi-modal embeddings as the complement to the item embeddings [4, 39]. On the contrary, [36, 42] use the graph neural networks [41, 46], which allows a more flexible usage of metadata, to generate the heterogeneous embeddings and calculate the graph-based user-item preference. Unlike those methods, M5 employs the pre-trained multi-modal CG embeddings in parallel with the raw ID embeddings and utilizes interest extraction and scenario mixing to improve the personalization and scenario distinction, which is more effective and flexible than the existing literature.

Multi-Interest Recommendation. User behavior modeling is crucial in the recommendation for user portraits. The previous works leverage the pooling-based [7], RNN-based [15, 47] and attention mechanism [48] for behavior compression. To capture users' diverse interests, the latest approaches consider the multi-interest modeling, where the capsule network [2, 23], memory network [5] and multi-head attention [3, 20, 28, 35] are adopted to encode the behavior sequences into multiple representations. In view of unique identities in OTT services, M5 first categorizes the user behaviors into buckets and decorates the subsidiary features to explicitly portray the coarse-grained and fine-grained preferences. Then, the masked-language-modeling loss is proposed to augment multi-head attention [9, 38] in capturing users' diverse interests while a subsidiary-intensity network is adopted for calibrated aggregation.

Multi-Scenario Recommendation. The multi-scenario recommendation is responsible for serving multiple scenarios with a unified model. Inspired by the recent success in multi-task learning [29, 37], [25, 34] introduce the shared knowledge across various scenarios via either expert networks or star topology. Different from the model-side innovations, M5 first crosses the features with the scenario indicators to directly encode the scenario information on the input, then proposes the Split Mixture-of-Experts to explicitly identify distinctions and commonalities among the tasks in the model, where the experts are further encouraged to be diverse by a disagreement loss.

3 PRELIMINARY

3.1 Problem Formulation

The OTT recommendation is responsible for presenting the relevant **shows** to the users, which consists of VOD series, VOD movies, and Live broadcasts. As the bottom layer, the objective of the matching stage is to retrieve a subset of N preferred shows for the given user $u \in \mathcal{U}$ from the available items $\mathcal{I}$, which can be formulated as:

$$\text{ArgTop}_N f(u, i), \quad i \in \mathcal{I}, \tag{1}$$

where $f(u, i)$ is defined as the matching algorithm that predicts the user preference for the target item (show).

3.2 Feature Representation

To give an accurate prediction, the matching model in the OTT services usually employs rich features from the users and shows that can be mainly categorized into the following aspects:

- **User features** describe users' attributes, including "age", "gender", and statistical information such as "user genre TF-IDF scores" and "watch count on the specific tags".
- **Behavior features** are the most important ones for user portrait, which involves the frequent "watch" behaviors as well as the infrequent "saved", "liked", and "disliked" actions for complete user interest modeling. Each behavior sequence is aggregated into **show** level to deduplicate repeated episodic watches of the same series (e.g., "The Simpsons" and "NBA"); otherwise, the long-range episodes and events would dominate the behavior sequences. We use "main" IDs to denote the show IDs after deduplication. Besides, the subsidiary features are also appended to each behavior, which will be discussed later in Sec.4.3. For simplicity, only **watch** behaviors are considered in the following sections as the other features are processed similarly.
- **Context features** identify the contextual information of the service, e.g., "device type", "day of the week", "hour of the day", etc. We also design features like "last behavior till now" to capture the freshness of the behaviors.
- **Item features** describe the target item information, which is identical to "main" IDs in the behavior features.

Conforming to the existing methods [48], M5 indexes the features into multi-field categorical form and transforms the samples into high-dimensional sparse vectors via one-hot or multi-hot encoding. Formally, user features, behavior features, context features, and item features are represented by $\mathbf{x}^u, \mathbf{x}^b, \mathbf{x}^c, \mathbf{x}^i$. While $\mathbf{x}^u, \mathbf{x}^c, \mathbf{x}^i$ only contain non-sequential features, $\mathbf{x}^b = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_{n_b}]$ represents the sorted (watch) behavior elements of length n_b.

4 M5

4.1 Overview of M5

The overall architecture of M5 forms two-channel two-tower modeling, as illustrated in Fig.2. A multi-modal embedding layer is proposed at the bottom to fully explore the rich metadata in the OTT services, which transforms each main feature into ID and CG embeddings. To efficiently retrieve the relevant items meanwhile incorporating the multi-modal semantics with personalization and contextualization, M5 expands the two-tower architecture to compute both ID and CG user-item preferences based on the corresponding user and item embeddings, where the user embeddings are generated from the paralleled multi-interest extraction and multi-scenario mixing layers considering users' diverse interests and inherit multiple scenarios while the item embeddings are derived from the lookup on the corresponding embedding tables.

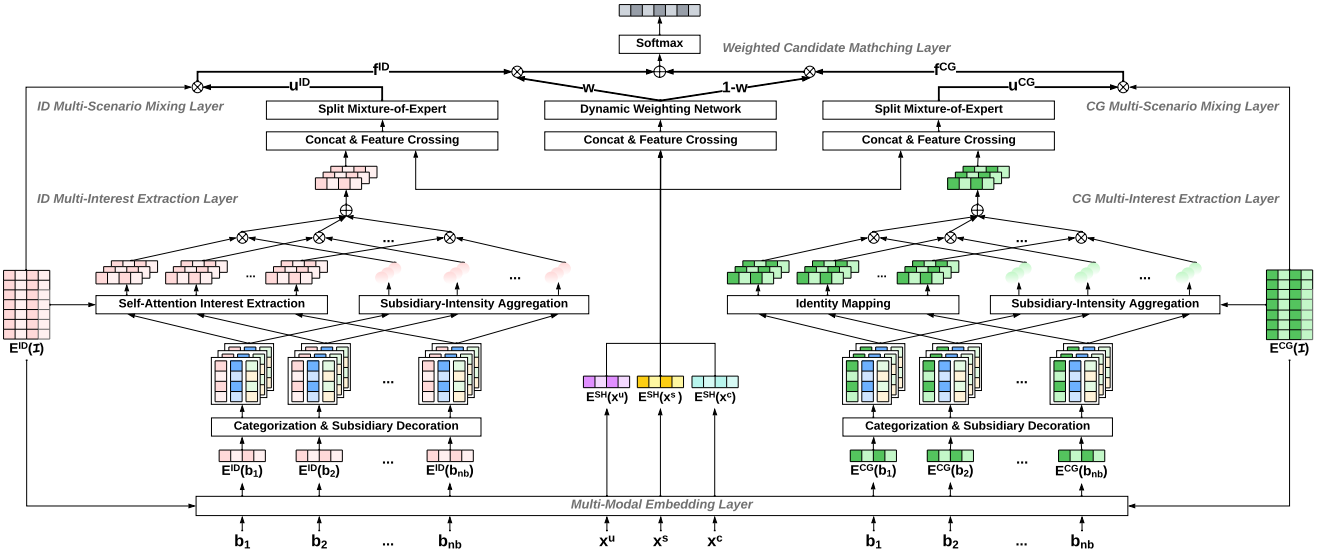


Figure 2: The architecture of M5. A multi-modal embedding layer is proposed at bottom to fully explore the rich meta information. Then, the separate multi-interest extraction layers and the multi-scenario mixing layers are applied to model users' diverse interests from the watch histories and explicitly identify the distinctions and commonalities among the scenarios to generate separate ID and CG user embeddings. A weighted candidate matching layer is finally established to calculate the ID-oriented and CG-oriented user-item preferences and merge them into a hybrid score through dynamic weighting.

A dynamic weighting network is finally applied to fuse the multi-modal predictions and generate a hybrid user-item preference.

4.2 Multi-Modal Embedding Layer

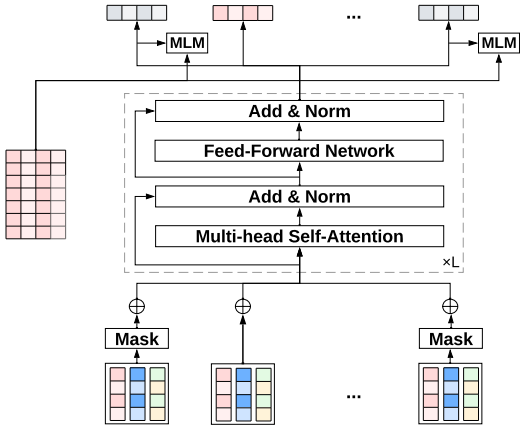
The embedding layer transforms the sparse features into dense vectors. Different from the previous methods, M5 exploits the rich meta information in the OTT services to generate the **content graph** (CG) embedding for each show as a supplement to the ordinary ID embeddings trained solely from the user behavior logging.

4.2.1 Multi-Modal Embedding. The multi-modal embedding is applied to both behavior and item features, which maps each main ID to both ID embedding and CG embedding to fully exploit the heterogeneous metadata in the OTT platform. ID embeddings are generated by the lookup operation on the show ID embedding table initialized randomly or from the previous incremental results, similar to the common industrial methodology. On the contrary, CG embeddings are initialized by the node embeddings derived from a pre-trained content metagraph comprised of show ID, tag, cast, visual, and text information. Each node in the metagraph represents a particular modality and the meta nodes are connected with the matched ID nodes to bridge the relevant shows. To better employ the visual and text information, the pre-trained ResNet-50 [13] and BERT [9] are utilized to generate the visual and text representations. M5 adopts pooling-based GraphSAGE [12] to train the node embeddings as we empirically found that the recent advances [46] fail to bring further improvement. The CG embeddings bring supplementary information to the ID embeddings and could generalize to the new comings, which is essential for solving the item cold-start problem in the OTT service. It is worth noting that although some previous methods [36, 42] also employ graph-based algorithms to encode the multi-modal information, the utilization of the extracted user/item embeddings is still coupled to the graph structure, which limits the capabilities of the metadata information in the model. Different from those methods, M5 only encodes the metadata information in the behavioral and target embeddings, remaining the flexibility in using the complicated multi-interest and multi-scenario modules upon the CG embeddings to intensify the use of meta information in the OTT services, which makes the retrieval results more accurate, personalized, and scenario-distinguishable than the models trained solely on the graph. In this paper, we use $\mathrm{E}^{ID}(\cdot)$ and $\mathrm{E}^{CG}(\cdot)$ to denote ID and CG embedding functions.

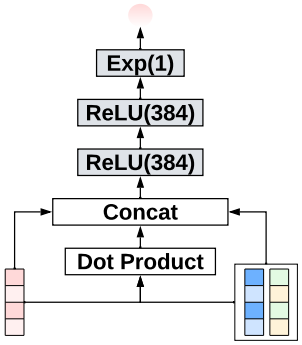
4.2.2 Shared Embedding. The shared embedding process $\mathrm{E}^{SH}(\cdot)$ handles the features other than the main IDs and performs the same lookup operation on the corresponding embedding tables.

4.3 Multi-Interest Extraction Layer

The multi-interest extraction layer produces behavioral representations based on the users' (watch) histories. To explicitly depict the **coarse-grained** and fine-grained preferences at the feature level, the sequential behaviors are divided into categories and decorated with the subsidiary features. Based on the multi-interest features, M5 achieves the model-side multi-interest extraction by introducing a **masked-language-modeling** (MLM) augmented **self-attention** model on the ID behavioral embeddings to excavate users' diverse interests and a **subsidiary-intensity** (SIN) aggregation for interest calibration on both ID and CG behavioral embeddings.



(a) Self-Attention Multi-Interest Extraction.



(b) Subsidiary-Intensity Calibration.

Figure 3: The self-attention multi-interest extraction and subsidiary-intensity interest calibration. The self-attention module leverages the Masked-Language-Model auxiliary loss for better gradient propagation on the sequence-level behavior modeling. The subsidiary-intensity module fuses the main ID (left) and subsidiary features (right) to calibrate users' interests.

4.3.1 Behavior Categorization and Subsidiary Decoration. The OTT platforms typically provide subscribers with VOD series, VOD movies, and Live broadcasts. According to the previous study, we found that the user interests in those categories are totally different, e.g., a user who regularly watches the "NCAA" or "NBA" Live events does not indicate her (his) interest in watching VOD sports series or movies. Therefore, M5 divides the user behaviors into different buckets according to the categorization to explicitly differentiate users' coarse-grained interests in the platform. More than bucketization, M5 also decorates the subsidiary attributes for each behavior to capture users' fine-grained interests. Concretely, M5 collects the show-level "episodic watch count", "playback duration" and "normalized playback duration" to differentiate the intensity of each behavior while preserving the temporal information with "show engage till now" and "show engage positional indices" that are widely used in the existing behavior modelings for positional encoding [3]. Therefore, each behavior element (take jth position as an example) in sequence $\mathbf{x}^b$ can be represented by $\mathbf{b}_j = [m_j, s_{j,1}, s_{j,2}, ..., s_{j,n_{sub}}]$, which comprises of one main ID and n_{sub} subsidiary features.

4.3.2 Self-Attention Multi-Interest Extraction. M5 adopts the advanced L-layer bidirectional Transformer encoder on the **ID** behaviors to extract diverse user interests and capture complex relations inside the sequences, as exhibited in Fig.3a. The sum of main and subsidiary embeddings of each position (take j-th as an example) is used as the input to distinguish fine-grained information. Each layer of the Transformer encoder contains two blocks of multi-head self-attention and feed-forward network, which models user preference from multiple views of interest by jointly attending information from various latent subspaces and then fusing the multiple interests with the point-wise MLP. Each block is surrounded by the residual connection with layer normalization for smooth and stable gradient updates. More details can be found in [38].

Masked-Language-Modeling Loss: Recent studies [9, 38] suggest that scaling up produces improved performance compared to more carefully engineered strategies. To see the effect of scaling on M5, we drop the CG channel, vary the depth of the self-attention model, and report the average Hit Ratio on two offline datasets (details can be found in Sec.5.1). Unlike the previous findings, the result in Tab.1 indicates that scaling up does not bring any benefits when $L > 1$ and even deteriorates the performance. Moreover, as the scaling increases, the degradation becomes more severe. These abnormal results have also been investigated by [3], and the authors believe that the phenomenon originates from the fact that the sequential dependency in users' behavior sequences is not as complex as the sentences in natural language processing, such that a smaller number of layers is enough to obtain good performance. However, we will present a different explanation that the failure of scaling is due to insufficient training in self-attention. Specifically, as the self-attention locates in a relatively shallow position and possesses most parameters in the model, it can not receive enough gradient updates from the optimization compared to the upper layers due to the gradient attenuation in the long-distance through back-propagation, which leads to the discrepancy in the convergence rates between the upper layers and self-attention, thereby when upper layers overfit, the self-attention is still underfitting. As the self-attention module grows deeper, this discrepancy will become more substantial, thus deteriorating the behavior modeling and even introducing noise with additional parameters. Therefore, M5 introduces a novel auxiliary loss $\mathcal{L}^{MLM}$ that utilizes the proven MLM training [9] on the behavior sequences to provide more supervision signals on the

Model	Dataset-H				Dataset-D			
	L=1	L=2	L=3	L=4	L=1	L=2	L=3	L=4
w/o MLM	0.4291	0.4315	0.4297	0.4258	0.4453	0.4510	0.4476	0.4402
w/ MLM	0.4368	0.4439	0.4527	0.4546	0.4661	0.4689	0.4712	0.4723

Table 1: The impact of MLM loss on the self-attention model.

self-attention module. Specifically, M5 randomly replaces the input sequence with an [MASK] embedding at positions $\mathcal{M}$ and feeds the final self-attention hidden states at the masked positions into an output softmax layer over the show ID embedding matrix to recover the origin main IDs:

$$\mathcal{L}^{MLM} = \frac{1}{|\mathcal{M}|} \sum_{j \in \mathcal{M}} -\log P^{MLM}_{m_j}(j) \tag{2}$$

Different from the standard MLM training, M5 does not replace the masked shows with random embeddings or remain the embeddings unchanged, as a simple [MASK] replacement is already shown to be sufficient for the MLM task, and the random replacement would primarily hurt the origin semantics when the input behavior sequences are relatively short that is quite common for the recommendation scenarios. Besides, only main IDs in the behavior sequences are masked, while the subsidiary features are reserved for more reconstruction information. As depicted in Tab.1, self-attention models trained with MLM widely surpass the comparatives under various depths, implying that the MLM loss helps the self-attention model gain more training signals from the unsupervised objectives and better portray the intrinsic relationship in the user behaviors. Note that the MLM training does not introduce any overhead on the inference speed and model size, thereby it can be taken as a plug-and-play component on any existing self-attention modeling.

4.3.3 Subsidiary-Intensity Interest Calibration. As shown in Fig.3b, SIN leverages the subsidiary features to generate the point-wise intensity scores and multiplies them on the behavioral representations for further calibrating the user preferences on the individual items. Given the behavior sequence $\mathbf{x}^b$, a multi-layer perceptron (MLP) with ReLU activation is applied on the concatenation of main, subsidiary fields and their interactions to compute the intensity score for each position (take j-th as an example):

$$\begin{aligned} int_j = \text{Exp}(\text{MLP}([&\text{E}(m_j), \text{E}^{SH}(s_{j,1}), ..., \text{E}^{SH}(s_{j,n_{sub}}), \\ &\text{E}(m_j) \cdot \text{E}^{SH}(s_{j,1}), ..., \text{E}(m_j) \cdot \text{E}^{SH}(s_{j,n_{sub}})])), \end{aligned} \tag{3}$$

where $\text{E}(\cdot)$ can be either $\text{E}^{ID}(\cdot)$ or $\text{E}^{CG}(\cdot)$, $\cdot$ refers to the inner product between embeddings. M5 uses an exponential function for a post-reweight to ensure the non-negative property of the intensities, which is also favorable for its efficiency in gradient propagation. To mitigate the effect of random initialization, the weights of the last MLP layer are initialized to zero so that the intensity scores would remain the same at the beginning and gradually grab the importance of each behavior. Given the intensity scores, M5 performs a weighted sum to compute the attentive aggregation. The SIN layer is adopted on the raw **CG** embeddings to reserve the semantics from the metagraph. Meanwhile, it is also applied upon the self-attention outputs for calibration on **ID** sequence.

4.4 Multi-Scenario Mixing Layer

As the OTT service is naturally multi-scenario, M5 proposes the **scenario indicator** and **Split Mixture-of-Experts** (SMoE) to explicitly identify distinctions and commonalities among the scenarios at both feature and model levels. A **disagreement regularization** is also introduced to encourage diversity in the SMoE experts.

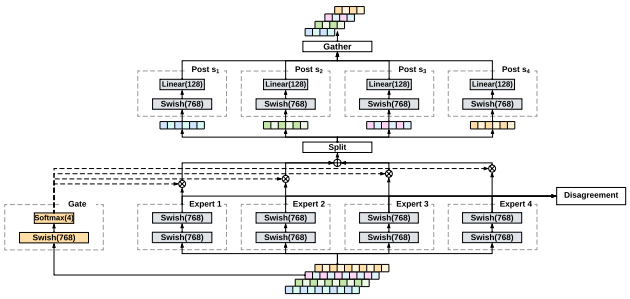


Figure 4: The structure of SMoE. Given the scenario-aware features (with different colors), SMoE employs shared experts to learn common knowledge and ensembles the outputs with a single gating network. The results are split by the scenarios, fed to the separate post-processing MLPs, and gathered back as the user embeddings. A disagreement regularization is employed to encourage expert diversity.

4.4.1 Scenario Indicator and Split Mixture-of-Experts. As described in Sec.1, OTT recommendation is inherently multi-scenario. Unlike the previous methods [29, 37] that share the inputs for all scenarios, M5 distinguishes the scenario variance at the feature level by attaching and crossing a scenario indicator $\text{E}^{SH}(\mathbf{x}^s)$ with the other features via inner product [31] as the input $\mathbf{f}_s$. To further identify scenario distinctions and commonalities by the model architecture, M5 introduces SMoE [29] illustrated in Fig.4, which adopts n_{exp} experts to facilitate effective information across scenarios from different subspaces and ensembles the outputs with a single gate network as input $\mathbf{f}_s$ is already discriminative. The samples are then split according to the scenarios and fed to the separate post-processing towers to further characterize the scenario disparity. The output of scenario s can be formulated as:

$$u_s = \text{Post}_s(\sum_{k=1}^{n_{exp}} \text{Expert}_k(\mathbf{f_s}) \, \text{Gate}_k(\mathbf{f_s})), \tag{4}$$

where u_s can be either u^{ID}_s and u^{CG}_s based on the type of the behavioral representations. The implementation of expert, gate, and post-processing networks is identical to the two-layer MLP with Swish activations [22, 32]. The post-processing outputs will be gathered back as the final user embeddings u^{ID} and u^{CG}.

4.4.2 Disagreement Regularization. One assumption in the MoE family is that each expert network is able to learn different patterns in the data and focus on various sub-modular in the latent space. However, there is no guarantee that the experts can learn distinct features from input. Therefore, M5 proposes the **disagreement regularization** to explicitly encourage expert diversity, which minimizes the absolute cosine similarity between the expert outputs for each sample. The regularization term is formally expressed as:

$$\mathcal{L}^{DIS} = \frac{1}{n^2_{exp}} \sum_{k=1}^{n_{exp}} \sum_{l=1}^{n_{exp}} \frac{|Expert_k(x) \cdot Expert_l(x)|}{\|Expert_k(x)\| \|Expert_l(x)\|}. \tag{5}$$

4.5 Weighted Candidate Matching Layer

To fully utilize the multi-modal information on the target side, M5 proposes a **weighted candidate matching layer** that calculates the **hybrid** user-item preferences for online retrieval. For user u and target item (show) $i \in \mathcal{I}$, M5 first generates the separate ID and CG user-item preferences by $f^{ID}(u,i) = u^{ID} \cdot \mathrm{E}^{ID}(i)$ and $f^{CG}(u,i) = u^{CG} \cdot \mathrm{E}^{CG}(i)$. Then, the preference scores are merged via a weight w computed by a dynamic weighting network (an MLP following the sigmoid activation) based on the input of user features, context features, and scenario indicator:

$$f(u,i) = wf^{ID}(u,i) + (1-w)f^{CG}(u,i). \tag{6}$$

M5 can be regarded as an expansion of the **multi-channel retrieval** that widely spread in the industrial recommender systems [24]. Different from the vanilla multi-channel retrieval that combines the **retrieved items** from different matching strategies, M5 merges the multi-modal **prediction scores** generated from the paralleled modelings with the dynamic weighting and retrieves a **single** candidate set based on the hybrid preference, which is more smooth and accurate. When the candidate set is large during online inference, M5 first generates a hybrid user embedding via $[wu^{ID}, (1-w)u^{CG}]$, and then performs query [19, 30] on the index built upon the concatenated item embeddings $[E^{ID}(\mathcal{I}), E^{CG}(\mathcal{I})]$. It can be proved that the retrieved outcomes are the same as the top items found by Eqn.6.

4.6 Loss Function

Given target item i, M5 employs the softmax cross entropy loss over the (sampled) item corpus $\mathcal{I}_{sample} \in \mathcal{I}$ as the matching objective:

$$\mathcal{L}^{Match} = -\log P_i^{Match} \tag{7}$$

$$P^{Match} = \mathrm{Softmax}(f(u, \mathcal{I}_{sample})) \tag{8}$$

Integrated with MLM loss and disagreement loss, M5 finally minimizes the **hybrid** loss for each sample:

$$\mathcal{L} = \mathcal{L}^{Match} + \alpha\mathcal{L}^{MLM} + \beta\mathcal{L}^{DIS} \tag{9}$$

5 EXPERIMENTS

5.1 Experimental Setup

5.1.1 Datasets and Metrics. We collect two industrial 31-day datasets from real-world OTT platforms Hulu (Dataset-H) and Disney+ (Dataset-D), as we find that no public dataset simultaneously considers multi-modal, multi-interest, and multi-scenario recommendations. The industrial datasets include user unfamiliar (not engaged before) watch histories with at least 2-minute playback.

- Dataset-H involves 150M instances with 30M users and 70K shows, which contains four scenarios differed in subscription type ("Live" and "VOD") and sample type ("Local" and "Global")[1]. Scenarios 1-4 in Dataset-H denote "VOD, Local", "Live, Local", "VOD, Global", and "Live, Global".
- Dataset-D involves 180M training instances with 100M users and 3K shows, including four scenarios separated by two representative regions ("Region I" and "Region II") and sample type ("Local" and "Global"). Scenarios 1-4 in Dataset-D denote "Region I, Local", "Region II, Local", "Region I, Global", and "Region II, Global".

We split the last-day behavior as the test set for both datasets while taking the others as the training set. The Hit Ratio at k (HR@k, same as Recall@k in the matching stage), representing the proportion of successful recommendations in the top-k retrieval set, is applied to evaluate the matching performance following the common practice [23, 24, 28]. This paper reports $k = 20, 100$ on both datasets based on the online retrieval settings.

5.1.2 Baseline Methods. We compare M5 with the below methods:

- YouTube DNN [7] utilizes the sum pooling on the user behaviors and then concatenates the results with other non-sequential features to derive the user embedding. The inner product between user embedding and target show embedding is adopted to compute the user-item preference.
- GRU4Rec$^+$ [14] is an improved version of [15], which utilizes a delicately-designed ranking-based loss function and sampling strategy to train the GRU on the user histories. The other parts remain the same as the YouTube DNN.
- BST [3] extends the YouTube DNN by utilizing Transformer layers to capture users' short-term interest in the behavior sequence. For computation efficiency, we use two-tower modeling other than an MLP in BST.
- SDM [28] is designed to capture users' dynamic preferences in the matching phase by combining short-term sessions and long-term behaviors with a gated fusion module. The multi-head self-attention is employed to capture the multiple interests in the short-term sessions.
- ComiRec [2] is proposed recently for multi-interest extraction. We implement ComiRec-SA, which leverages the self-attentive mechanism to extract multiple interests from user behaviors and incorporates a controllable aggregation module to balance recommendation diversity and accuracy.
- PDN [24] establishes 2-hop paths from user to target item through the Trigger Net and Similarity Net to accommodate user personalization and interest diversity.
- CL4SRec [43] incorporates item cropping, masking, and reordering as augmentations for contrastive learning on the Transformer-based model.

We also propose three M5 variants to study the individual effect of multi-modal, multi-interest, and multi-scenario modeling:

- M5 w/o MM (Multi-Modal) drops the CG embeddings and the architectures built upon in M5.
- M5 w/o MI (Multi-Interest) replaces the multi-interest extraction layer with the simple sum pooling layer on the complete ID and CG behavioral sequences without bucketization and subsidiary decoration.
- M5 w/o MS (Multi-Scenario) discards the scenario indicators and changes SMoE to an MLP.

To ensure a fair comparison, we conduct precise hyperparameter tuning for each baseline by the grid search and adjust the embedding

[1] "Live" subscribers can watch both Live and VOD contents while "VOD" subscribers can only access to the VOD contents. "Local" refers to the sets that the matching models really work. "Global" refers to the non-local behaviors which provide additional information for user portrait and avoid the Matthew Effect.

Group	Model	Dataset-H								Dataset-D							
		Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 1		Scenario 2		Scenario 3		Scenario 4	
		HR@20	HR@100	HR@20	HR@100	HR@20	HR@100	HR@20	HR@100	HR@20	HR@100	HR@20	HR@100	HR@20	HR@100	HR@20	HR@100
Baseline	YouTube DNN [7]	0.3178	0.7206	0.1949	0.5401	0.3178	0.7206	0.1491	0.4972	0.5612	0.8287	0.5094	0.7681	0.3518	0.6495	0.3583	0.6530
	GRU4Rec$^+$ [6]	0.3209	0.7205	0.1983	0.5453	0.3145	0.7220	0.1522	0.5059	0.5681	0.837	0.5175	0.7689	0.3550	0.6535	0.3738	0.6594
	BST [11]	0.3918	0.8349	0.2051	0.5471	0.3378	0.7279	0.1918	0.5471	0.5891	0.8610	0.5739	0.8083	0.4198	0.6947	0.4462	0.7101
	SDM [28]	0.4062	0.8414	0.2021	0.5435	0.3276	0.7240	0.1841	0.5324	0.5820	0.8576	0.5655	0.7962	0.4239	0.6955	0.4391	0.7003
	ComiRec [40]	0.3413	0.7845	0.1939	0.5462	0.3221	0.7293	0.1636	0.5121	0.5685	0.8451	0.5313	0.7803	0.3714	0.6734	0.4036	0.6840
	PDN [24]	0.3546	0.8251	0.1987	0.5504	0.3301	0.7295	0.1757	0.5269	0.5906	0.8662	0.5867	0.8071	0.4249	0.7018	0.4468	0.7080
	CL4SRec [43]	0.4017	0.8408	0.2123	0.5564	0.3396	0.7370	0.1935	0.5597	0.5976	0.8620	0.5854	0.8155	0.4226	0.7068	0.4449	0.7012
M5 Variants	M5	**0.4856**	0.9089	**0.3881**	**0.8215**	0.4247	0.7803	**0.4761**	**0.7645**	0.7075	0.9369	**0.6900**	**0.9271**	0.4759	0.7487	**0.5047**	**0.7627**
	M5 w/o MM	0.4628	0.9003	0.3787	0.8107	0.4114	0.7759	0.4760	0.7642	0.6979	0.9281	0.6782	0.9139	0.4639	0.7305	0.4921	0.7531
	M5 w/o MI	0.4130	0.8505	0.2926	0.7389	0.3295	0.7410	0.2297	0.5568	0.6388	0.8783	0.5947	0.8222	0.3727	0.6662	0.3801	0.6664
	M5 w/o MS	0.4317	0.8691	0.2385	0.6407	**0.4293**	0.7746	0.3586	0.6692	0.6303	0.8905	0.6157	0.8780	0.4566	0.7386	0.4889	0.7530
Multi-Modal	M5 w/ Raw Metadata	0.4623	0.8999	0.3790	0.8107	0.4112	0.7755	0.4740	0.7587	0.6979	0.9270	0.6785	0.9153	0.4644	0.7283	0.4933	0.7551
	M5 w/ Single Channel	0.4744	0.9025	0.3819	0.8124	0.4151	0.7761	0.4712	0.7614	0.7017	0.9315	0.6801	0.9197	0.4680	0.6715	0.4978	0.7594
Multi-Interest	M5 w/o Cate & Sub	0.4386	0.8895	0.3541	0.7690	0.3749	0.7655	0.3411	0.6500	0.6726	0.9072	0.6495	0.8601	0.4264	0.7029	0.4428	0.7042
	M5 w/ SIN	0.4464	0.8891	0.3371	0.7843	0.3659	0.7687	0.4316	0.7221	0.6797	0.9159	0.6633	0.8786	0.4400	0.6972	0.4572	0.7174
	M5 w/ SA	0.4603	0.8955	0.3655	0.8056	0.3936	0.7711	0.4287	0.7412	0.6904	0.9230	0.6779	0.9011	0.4597	0.7263	0.4849	0.7369
	M5 w/ SA-SIN	0.4821	0.9058	0.3856	0.8186	0.4217	0.7761	0.4701	0.7556	**0.7082**	**0.9388**	0.6899	0.9238	**0.4776**	**0.7501**	0.5036	0.7586
Multi-Scenario	M5 w/o idx	0.4588	0.8793	0.2429	0.6647	0.4140	0.7669	0.4227	0.6905	0.6457	0.9005	0.6335	0.8932	0.4617	0.7452	0.4924	0.7608
	M5 w/o dis	0.4842	0.9033	0.3828	0.8120	0.4182	0.7811	0.4682	0.7629	0.7044	0.9317	0.6856	0.9247	0.4750	0.7446	0.5015	0.7553
	M5 w/ CGC [37]	0.4843	**0.9111**	0.3693	0.8048	0.4268	**0.7828**	0.4705	0.7547	0.7061	0.9349	0.6880	0.9256	0.4746	0.7464	0.5034	0.7617
	M5 w/ PLE [37]	0.4783	0.9058	0.3726	0.8088	0.4237	0.7764	0.4726	0.7575	0.7060	0.9354	0.6885	0.9219	0.4744	0.7461	0.5038	0.7615

Table 2: The offline results on Dataset-H and Dataset-D. Bold / underlined scores are the best / second of each column.

sizes and hidden sizes of the comparisons to ensure that their model sizes are roughly the same as M5 variants.

5.1.3 Implementation Details. We set the embedding dimension to 128, 128, and 8 for the behavioral, categorical, and other demographic features in the multi-modal embedding layer. For the self-attention extraction, we adopt $L = 3$ layers with hidden size 128 in consideration of the serving efficiency. Following the common settings [38], the dimension of each attention head is set to 64, and thus the number of attention heads is $128/64 = 2$. The inner layer in the feed-forward network is set to 512 following the design of inverted bottleneck. For the subsidiary-intensity calibration, we use a 3-layer MLP with hidden size [384, 384, 1] for the CG behaviors and a simple linear layer for the ID behaviors. The MLPs in SMoE and weighted candidate matching all possess a hidden size of 768 with 0.1 Dropout on the inputs. To balance the auxiliary objectives, α and β are set to 0.1 based on a grid search as setting α or β larger would make the objective deviate from the matching loss. M5 utilizes the Adam optimizer [21] with linear decay and batch size 2048 for training, where the initial learning rate is set to 0.002 to avoid overfitting. It is worth mentioning that we do not perform any negative sampling [7, 45] for both M5 and baselines during model training as the candidate size is limited in Hulu and Disney+, i.e., less than 100K, thereby the softmax loss of Eqn.2 and Eqn.8 are naturally unbiased.

5.2 Offline Evaluation

The first group in Tab.2 summarizes the results of baseline methods. Compared with the YouTube DNN, GRU4Rec$^+$ does not lead to much improvement as the sequential patterns are mainly lost during the show-level behavior compression. In contrast, BST and SDM consistently improve the recommendation owing to the powerful bi-directional modeling of self-attention. Though ComiRec outperforms the single-interest algorithms, the adopted self-attentive approach displays inferiority to the intra-behavior modeling of BST and SDM, revealing the hardship for the multi-head attention to learn diverse interests on the single-modular features without explicit training signals. Compared to the baseline methods, M5 achieves a remarkable improvement of over 10% HR lift on both Dataset-H and Dataset-D under all scenarios, especially for the "Local" scenarios that we actually optimize online, which strongly proves that capturing the multi-modal multi-interest multi-scenario characteristics is critical for the OTT recommendation. Moreover, all designs for the multi-modal, multi-interest, and multi-scenario modelings are beneficial and bring orthogonal improvements on M5, as highlighted in the second group of Tab.2. Specifically, the multi-interest extraction layer is essential for both datasets as the user behaviors contain most of the user information in the platform. Meanwhile, the multi-scenario strategy significantly improves the performance for "Local" scenarios that always possess fewer training samples, demonstrating that M5 can adapt to the scenario difference under the imbalanced sample distribution, otherwise, the model would be dominated by scenarios with the majority of samples. Besides, although ID embeddings are well-trained on a large number of behavioral samples, the multi-modal embeddings still solidly provide additional meta information to the user behaviors and gain obvious enhancement on both datasets.

5.3 Ablation Study

5.3.1 Ablation Study on Multi-Modal. To show the effectiveness of the graph embeddings, we replace the CG embeddings with the concatenation of the representative meta features (show genres, brands, texts) and feed them to the same layers in the CG channel. As presented in Tab.2, no obvious lift can be found in "M5 w/ Raw Meta" compared to "M5 w/o MM", indicating the preponderance of using graph-based methods on the meta information in the OTT recommendation. We also introduce a single-channel variant "M5 w/ Single Channel" that directly combines the CG and ID behavioral representations to generate a single user embedding that would finally interact with the unified target embeddings derived by an MLP taking inputs from ID and CG item embeddings. In line

Indicators	Hulu		Disney+	
	VOD	Live	Region I	Region II
Local HPV	+3.8%	+12.0%	+49.7%	+52.6%
Global HPV	+0.9%	+1.2%	+0.7%	+0.9%

Table 3: Online A/B test of M5 compared to the baselines.

with expectations, isolating the meta semantics between ID and CG embeddings improves the model expressiveness, proving the superiority of the multi-channel design in M5.

5.3.2 Ablation Study on Multi-Interest. To examine the effect of behavior categorization and subsidiary decoration, we propose "M5 w/o Bucket & Sub" that performs the multi-interest extraction directly on the main ID and CG embeddings. From the results in the "Multi-Interest" group, the variant steadily degrades the matching performance on both datasets, indicating the importance of explicitly seizing the coarse-grained and fine-grained user preferences from feature engineering. We also investigate the detailed architectures of ID, and CG interest extractions and propose three alternatives that exploit both self-attention with SIN ("M5 w/ SA-SIN"), raw self-attention ("M5 w/ SA") or raw SIN ("M5 w/ SIN") in the two channels. Apparently, "M5 w/ SIN" and "M5 w/SA" are still much inferior to M5, revealing the capability of self-attention interest extraction and SIN interest calibration. In contrast, "M5 w/ SA-SIN" obtains similar results as M5 yet induces more parameters, implying that the semantics of the CG embeddings should be maintained in the behavior modeling.

5.3.3 Ablation Study on Multi-Scenario. To see the effect of scenario indicator and disagreement loss in the multi-scenario mixing layer, we propose the variant "M5 w/o idx" and "M5 w/o dis" for comparison, as shown in the last group of Tab.2 Though SMoE can distinguish the scenario difference from architecture, "M5 w/o idx" is still much inferior to M5 over all scenarios, especially for the "Local" scenarios with fewer samples, indicating the necessity of injecting the scenario knowledge at the feature level. On the other hand, disagreement loss achieves consistent improvement without introducing any computation overhead during inference. We also adapt CGC and PLE [37] to the multi-scenario matching, where CGC explicitly separates shared and scenario-specific experts and PLE stacks multiple layers of CGC. All methods employ scenario indicators and disagreement regularization for a fair comparison. It is obvious that CGC is hard to further mitigate the seesaw phenomenon as the scenario-aware features and discriminative experts can already alleviate the interference in the parameters. Besides, we do not observe much difference in multi-layer architectures, where we suspect that the OTT scenarios may not possess the more profound semantic knowledge described in [37]. Therefore, we choose SMoE as it possesses fewer parameters but better performance.

5.4 Online A/B Test

5.4.1 Deployment. We deploy M5 on the most popular "For You" sets of Hulu and Disney+. Both platforms adopt the **nearline** architecture [1] to minimize computational complexity, which decouples the candidate generation from **online** requests and makes the cost of matching stage negligible compared to the subsequent layers:

- When the user logins or watches/likes/saves a show, the recommender system will trigger the nearline matching model to update the retrieval set according to the latest features and then store the outputs in the cache.
- When the user initiates a request, the recommender system will directly fetch the retrieval outcomes from the storage and feed them to the online ranking stage.

Since the renewal of the retrieved set only relies on infrequent explicit user behavior triggers rather than frequent online requests, the matching latency can be roughly ignored, and thus only a limited amount of computation resources is sufficient for the service.

5.4.2 Experiment Details. We conduct a three-week online A/B experiment on both platforms with 10% real traffic (5% for control and treatment). Considering Hulu and Disney+ serve over 100M subscribers, the experiments certainly endorse the statistical significance. The hours per visitor (HPV) is employed for evaluation, which computes the average user streaming hours within the Local "For You" sets (Local HPV) or entire platform (Global HPV) that are most relevant to the commercial objective of long-term retention and advertising revenue. We compare M5 with the online matching models of YouTube DNN [7] under precise feature engineering on Hulu and variational autoencoders [26] on Disney+.

5.4.3 Result. Tab.3 presents the indicator lifts during the experiment. It is evident that M5 significantly surpasses the compared methods under all scenarios of subscription types and regions owing to exploiting multi-modal multi-interest multi-scenario characteristics in the OTT services. Remarkably, M5 achieves 1.0%/0.8% average Global HPV lifts on Hulu/Disney+. Considering algorithm-driven platform-level improvements are extremely rare in OTT applications, therefore, the online result once again demonstrates the effectiveness and robustness of the proposed method. M5 has been fully deployed on the "For You" sets of both platforms, serving hundreds of millions of subscribers every day and continuously enhancing both user experience and business revenue.

6 CONCLUSION

This paper introduces Multi-Modal Multi-Interest Multi-Scenario Matching (M5) to fully exploit the unique characteristics of the OTT recommendation. M5 proposes a multi-modal embedding layer to explore the rich metadata information, a multi-interest extraction layer to capture users' diverse interests, a multi-scenario mixing layer to facilitate effective information transformation across multiple scenarios, and a weighted candidate matching layer to merge the hybrid user-item preferences. The extensive experiments over Hulu and Disney+ broadly demonstrate the superiority of M5 in industrial OTT services and emphasize the utilization of task-specific properties in model design. M5 has been deployed on both platforms, serving hundreds of millions of subscribers every day.

REFERENCES

[1] Xavier Amatriain. 2013. Big & personal: data and models behind netflix recommendations. In *Proceedings of the 2nd international workshop on big data, streams and heterogeneous source Mining: Algorithms, systems, programming models and applications*. 1–6.

[2] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *ACM SIGKDD*. 2942–2951.

[3] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. 2019. Behavior sequence transformer for e-commerce recommendation in Alibaba. In *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*. 1–4.

[4] Xu Chen, Hanxiong Chen, Hongteng Xu, Yongfeng Zhang, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2019. Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation. In *ACM SIGIR*. 765–774.

[5] Xu Chen, Hongteng Xu, Yongfeng Zhang, Jiaxi Tang, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2018. Sequential recommendation with user memory networks. In *WSDM*. 108–116.

[6] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[7] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *RecSys*. 191–198.

[8] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, et al. 2010. The YouTube video recommendation system. In *RecSys*. 293–296.

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

[10] Carlos A Gomez-Uribe and Neil Hunt. 2015. The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)* 6, 4 (2015), 1–19.

[11] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *International Joint Conference on Artificial Intelligence*. 1725–1731.

[12] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *NeurIPS* 30 (2017).

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[14] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 843–852.

[15] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and D Tikk. 2016. Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations*.

[16] Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2553–2561.

[17] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2333–2338.

[18] Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2010. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence* 33, 1 (2010), 117–128.

[19] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.

[20] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *ICDM*. IEEE, 197–206.

[21] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

[22] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 972–981.

[23] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *CIKM*.

[24] Houyi Li, Zhihong Chen, Chenliang Li, Rong Xiao, Hongbo Deng, Peng Zhang, Yongchao Liu, and Haihong Tang. 2021. Path-based Deep Network for Candidate Item Matching in Recommenders. In *SIGIR*. 1493–1502.

[25] Pengcheng Li, Runze Li, Qing Da, An-Xiang Zeng, and Lijun Zhang. 2020. Improving multi-scenario learning to rank in e-commerce by exploiting task relationships in the label space. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2605–2612.

[26] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*. 689–698.

[27] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing* (2003).

[28] Fuyu Lv, Taiwei Jin, Changlong Yu, Fei Sun, Quan Lin, Keping Yang, and Wilfred Ng. 2019. SDM: Sequential deep matching model for online large-scale recommender system. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2635–2643.

[29] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1930–1939.

[30] Marius Muja and David G Lowe. 2014. Scalable nearest neighbor algorithms for high dimensional data. *IEEE transactions on pattern analysis and machine intelligence* 36, 11 (2014), 2227–2240.

[31] Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G Azzolini, et al. 2019. Deep learning recommendation model for personalization and recommendation systems. *arXiv preprint:1906.00091* (2019).

[32] Prajit Ramachandran, Barret Zoph, and Quoc V Le. 2017. Searching for activation functions. *arXiv preprint arXiv:1710.05941* (2017).

[33] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. 285–295.

[34] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, et al. 2021. One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. In *CIKM*. 4104–4113.

[35] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[36] Rui Sun, Xuezhi Cao, Yan Zhao, Junchen Wan, Kun Zhou, Fuzheng Zhang, Zhongyuan Wang, and Kai Zheng. 2020. Multi-modal knowledge graphs for recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1405–1414.

[37] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In *RecSys*. 269–278.

[38] 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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 6000–6010.

[39] Jizhe Wang, Pipei Huang, Huan Zhao, Zhibo Zhang, Binqiang Zhao, and Dik Lun Lee. 2018. Billion-scale commodity embedding for e-commerce recommendation in alibaba. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 839–848.

[40] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*.

[41] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *SIGKDD*. 950–958.

[42] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1437–1445.

[43] Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. 2022. Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering (ICDE)*. 1259–1273.

[44] Xiaoran Xu, Laming Chen, Songpeng Zu, and Hanning Zhou. 2018. Hulu video recommendation: from relevance to reasoning. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 482–482.

[45] Xinyang Yi, Ji Yang, Lichan Hong, Derek Zhiyuan Cheng, Lukasz Heldt, Aditee Kumthekar, Zhe Zhao, Li Wei, and Ed Chi. 2019. Sampling-bias-corrected neural modeling for large corpus item recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 269–277.

[46] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.

[47] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *AAAI*. 5941–5948.

[48] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *SIGKDD*. 1059–1068.