PASS: Personalized Advertiser-aware Sponsored Search

Zhoujin Tian
deritt7@gmail.com
Microsoft
Beijing, China

Chaozhuo Li*
cli@microsoft.com
Microsoft Research Asia
Beijing, China

Zhiqiang Zuo
zhiqzuo@microsoft.com
Microsoft
Beijing, China

Zengxuan Wen
zewen@microsoft.com
Microsoft
Beijing, China

Lichao Sun
lis221@lehigh.edu
Lehigh University
Bethlehem, PA, United States

Xinyue Hu
xinyuehu@microsoft.com
Microsoft
Beijing, China

Wen Zhang
zhangw@microsoft.com
Microsoft
Beijing, China

Haizhen Huang
hhuang@microsoft.com
Microsoft
Beijing, China

Senzhang Wang
szwang@csu.edu.cn
Central South University
Changsha, China

Denvy Deng
dedeng@microsoft.com
Microsoft
Beijing, China

Xing Xie
xingx@microsoft.com
Microsoft Research Asia
Beijing, China

Qi Zhang
qizhang@microsoft.com
Microsoft
Beijing, China

ABSTRACT

The nucleus of online sponsored search systems lies in measuring the relevance between the search intents of users and the advertising purposes of advertisers. Existing conventional doublet-based (query-keyword) relevance models solely rely on short queries and keywords to uncover such intents, which ignore the diverse and personalized preferences of participants (i.e., users and advertisers), resulting in undesirable advertising performance. In this paper, we investigate the novel problem of **P**ersonalized **A**dvertiser-aware **S**ponsored **S**earch (PASS). Our motivation lies in incorporating the portraits of users and advertisers into relevance models to facilitate the modeling of intrinsic search intents and advertising purposes, leading to a quadruple-based (i.e., user-query-keyword-advertiser) task. Various types of historical behaviors are explored in the format of hypergraphs to provide abundant signals on identifying the preferences of participants. A novel heterogeneous textual hypergraph transformer is further proposed to deeply fuse the textual semantics and the high-order hypergraph topology. Our proposal is extensively evaluated over real industry datasets, and experimental results demonstrate its superiority.

*Equal contribution and corresponding author

KDD '23, August 6–10, 2023, Long Beach, CA, USA.

ACM ISBN 979-8-4007-0103-0/23/08…$15.00
https://doi.org/10.1145/3580305.3599882

CCS CONCEPTS

• **Information systems** → **Sponsored search advertising**.

KEYWORDS

Sponsored Search, Hypergraph Learning, Relevance Modeling

ACM Reference Format:
Zhoujin Tian, Chaozhuo Li, Zhiqiang Zuo, Zengxuan Wen, Lichao Sun, Xinyue Hu, Wen Zhang, Haizhen Huang, Senzhang Wang, Denvy Deng, Xing Xie, and Qi Zhang. 2023. PASS: Personalized Advertiser-aware Sponsored Search. 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, 13 pages. https://doi.org/10.1145/3580305.3599882

1 INTRODUCTION

Sponsored search, in which ads appear alongside organic search results in search engines, has become a highly profitable channels for e-commerce [17]. Advertisers can bid on specific keywords so that their ads will show up when target users search for the kind of things they sell. The nucleus of the sponsored search is to precisely align the ***search intents*** from ***users*** and the ***advertising purposes*** from ***advertisers***, which contributes to improving the user experience and driving revenue for the advertisers simultaneously.

Existing models [3, 21, 27, 30, 40, 50] generally view the queries input by users and the keywords bid by advertisers as the carriers of search intents and advertising purposes, respectively. Extensive Natural Language Understanding (NLU) models have been employed to extract such intentions from pure textual queries and keywords (e.g., DSSM [28, 40], pre-trained language models [14, 30, 42]). However, pure NLU models might not be the panacea to fully solve all the challenges in sponsored search since their modeling capacity is hindered by the scarce semantics within the short queries and keywords [50]. Several recent endeavors [27, 34, 50] propose

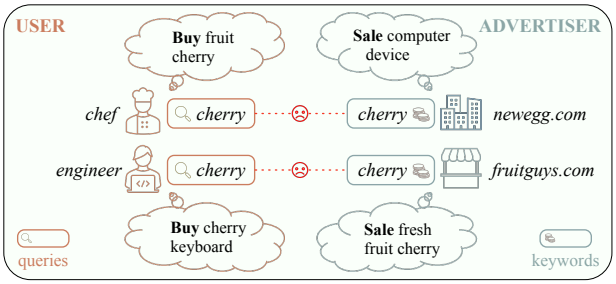


Figure 1: An example demonstrates that identical queries and keywords may lead to inferior advertising results due to the diverse preferences of users and advertisers.

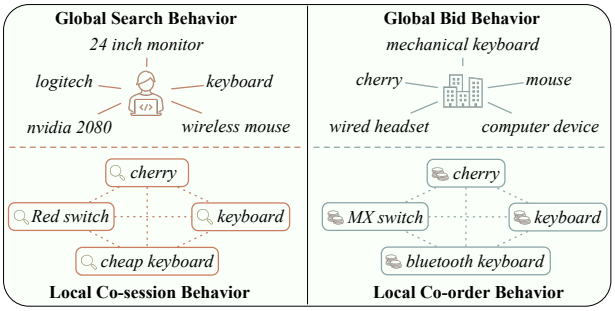


Figure 2: An example of the global and local behaviors.

to incorporate the user historical behaviors as complementary to incorporate extra knowledge beyond the pure semantics.

Existing sponsored search systems generally follow the **doublet-based paradigm**, aiming at measuring the semantic closeness between queries and keywords. However, the complaints from advertisers and the low click-through rate (CTR) of users are consistently emerging on the industry search engines [6]. The primary reason lies in that queries and keywords are insufficient to fully uncover the search intents and advertising purposes. Same queries (keywords) from different users (advertisers) are assumed to share the same intentions, leading to the identical representations. Nevertheless, this hypothesis may not hold true in real-world scenarios, as the personalized and diverse preferences of different users (advertisers) may affect the outcome [5, 6, 23, 49]. Figure 1 displays a real case from Bing Ads. The query "*cherry*" input by a chef and a programmer may reflect distinct search intents (i.e., fruit and keyboard) beneath the identical superficial semantics. This also occurs on the side of advertisers, as different advertisers bidding on the same keywords may have different advertising objectives. In a nutshell, identical texts from various participants may express diverse intentions, while traditional doublet-based methods ignore the diversity and personalization of participant preferences, leading to inferior online performance.

To facilitate the fine-grained modeling of personalized intentions, we investigate the novel problem of **quadruple-based paradigm** (i.e., user-query-keyword-advertiser) for sponsored search. Compared to conventional doublet-based models, quadruple-based ones incorporate the unique preferences of users and advertisers to help uncover accurate intentions. Nevertheless, the studied task is plagued by two critical challenges. First, it is intractable to learn representations for users and advertisers directly. Users are usually represented by anonymous identities (e.g., client id) without any demographic information. Advertisers are usually identified by domain URLs, which are too obscure to indicate the intrinsic features (e.g., "*indeed.com*"). Second, the representations of queries and keywords should encode both the textual semantics and the characteristics of associated participants. For example, the query "*cherry*" input by a chef should be close to "*fruit cherry*", while the same query from a engineer is similar to "*cherry keyboard*".

Inspired by previous works [6, 27, 44], the cheap and massive historical behaviors of users and advertisers are incorporated to address the mentioned challenges. First, the search histories of users and the bid behaviors of advertisers, dubbed **global behaviors**, are introduced to learn the participant representations. Our motivation lies in that users who search similar queries or click similar ads tend to share similar preferences [6]. Similarly, advertisers tend to be close if they bid similar keywords. Second, the co-session queries or the co-order keywords, dubbed **local behaviors**, are incorporated to learn personalized query or advertiser-aware keyword representations. Namely, a session is a group of queries from a user that takes place within a given time frame and usually has a consistent search goal. As shown in Figure 2, the co-session queries would contribute to deciding whether the input query "*cherry*" indicates fruit or keyboard. Orders are placed by advertisers to the search engine and contain a set of keywords belonging to the same category. From Figure 2, with the co-order keywords "*bluetooth keyboard*" and "*cherry keyboard*", we can understand that the keyword "*cherry*" bid by "*cherry-world.com*" refers to computer accessories. By enjoying the merits of these complementary information, the quadruple-based model is capable of precisely uncovering and matching the search intents and advertising purposes.

Conventionally, user behaviors are modeled in the format of pairwise graphs [27, 34, 46]. The relationships between nodes (queries and keywords) are in pairwise formulations, meaning that each edge only connects two nodes. However, the relationships in the studied quadruple-based task are in **high-order**, beyond the pairwise format. For example, by taking the queries as nodes and the co-session relation as edges, each edge tends to link more than two nodes. The affinity relations are no longer dyadic (pairwise) but rather triadic, tetradic, or of a higher-order. Conventional pairwise graphs are incapable of modeling such high-order relationships [2, 31]. In addition, previous works usually view the behavior graph as a bipartite graph with a single type of relationship [27, 34]. Nevertheless, historical global and local behaviors contain multiple types of nodes and relations, resulting in a **heterogeneous** behavior graph. The heterogeneity of node features and relationships intrinsically depict the unique traits of the quadruples, which is crucial in revealing real intentions.

To tackle aforementioned challenges, we propose a novel quadruple-based sponsored search paradigm, dubbed **P**ersonalized **A**dvertiser-aware **S**ponsored **S**earch (PASS). Different from conventional pairwise graphs, the heterogeneous hypergraph is introduced as the backbone to model historical behaviors. Hypergraph leverages the hyperedges to connect multiple vertices simultaneously to capture high-order relations, which is more suitable for depicting global and

local behaviors. Specifically, user, advertiser, query, and keyword are viewed as four categories of nodes, and various hyperedges are elaborately designed to model high-order relations. A heterogeneous textual hypergraph transformer is further proposed to deeply aggregate the fine-grained semantics within the nodes and heterogeneous high-order topological correlations from the hypergraphs, which contributes to capturing the personalized preferences in the historical global and local behaviors. The learned high-quality embeddings of users and queries (advertisers and keywords) are fused together as the final representations of search intents (advertising purposes). Empirically, PASS is extensively evaluated over several industry datasets and yields significant gains over SOTA baselines. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to study the novel and practical problem of personalized advertiser-aware sponsored search.
- We propose a novel heterogeneous textual hypergraph based model PASS to precisely uncover search intents and advertising purposes by deeply fusing the textual semantics and enormous historical behaviors.
- Our proposal is thoroughly evaluated over several industry datasets and consistently outperforms SOTA approaches.

2 PRELIMINARY

2.1 Problem Definition

Different from existing doublet-based approaches, the notations of "advertiser" and "user" are introduced to form up the quadruple: (u_i, q_i, k_i, a_i), denoting query q_i input by user u_i and the keyword k_i bid by advertiser a_i. The underlying search intent and advertising purpose can be further modeled based on (u_i, q_i) and (a_i, k_i), respectively. For each user u_i, we collect her historical behavior set $\mathcal{U}_i$ containing all historical queries searched by u_i (i.e., global behaviors) and the corresponding session information (i.e., local behaviors). Similarly, for each advertiser a_i, we also explore the historical behaviors $\mathcal{A}_i$, including all keywords bid by a_i and the corresponding co-order information. Formally, we aim to learn a relevance model f: $f(u_i, q_i, k_i, a_i) \rightarrow \{0, 1\}$ by fusing the historical behavior sets $\mathcal{U}_i$ and $\mathcal{A}_i$ to predict the relevance score between search intent from (u_i, q_i) and advertising purpose from (a_i, k_i). Notations used in this paper are listed in Appendix A.

2.2 Hypergraph

A hypergraph is a flexible topological structure in which a hyperedge can connect more than two nodes. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{H})$ be an instance of hypergraph, which includes a node set $\mathcal{V} = \{v_i\}_{i=1}^{n}$ and a hyperedge set $\mathcal{E} = \{e_j\}_{j=1}^{m}$. Each hyperedge e_j can connect a set of nodes $\{v_1, v_2, ..., v_k\} \subseteq \mathcal{V}$. The relationship between nodes and hyperedges is represented by an incidence matrix $\mathbf{H} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{E}|}$:

$$\mathbf{H}_{i,j} = \begin{cases} 1, & \text{if } e_j \text{ connects } v_i, \\ 0, & \text{if } e_j \text{ disconnects } v_i. \end{cases} \tag{1}$$

For each node $v_i \in \mathcal{V}$, $\mathcal{E}_v(v_i) = \{e_j \in \mathcal{E} | \mathbf{H}_{i,j} = 1\}$ denotes the set of hyperedges connected to v_i. Similarly, $\mathcal{V}_e(e_j) = \{v_i \in \mathcal{V} | \mathbf{H}_{i,j} = 1\}$ denotes the set of nodes connected to e_j.

3 METHODOLOGY

3.1 Framework

Figure 3 exhibits the framework of the proposed PASS model. We first construct a heterogeneous hypergraph from the historical behaviors of users and advertisers to model the complex high-order relations among multiple nodes, which contains four types of nodes and eight types of hyperedges. Given the input quadruple, a sub-hypergraph is first sampled based on the Personalized Page Rank importance, consisting of the essential global and local behaviors. Then, the powerful language models are employed as the encoders for textual nodes and will be efficiently co-trained with the proposed heterogeneous textual hypergraph transformer to deeply fuse the fine-grained semantics and heterogeneous topology information, facilitating the learning of high-quality node representations. Based on the learned representations of entries in the input quadruple, relevance between the embeddings of search intent and advertising purpose is measured by the scoring layers.

3.2 Heterogeneous Hypergraph Construction

Previous works [6, 27] primarily construct pairwise graphs based on click behaviors, which are incapable of capturing the sophisticated and high-order correlations within the quadruples. Different from previous works, we propose to model historical behaviors as the hypergraph due to its inherently superior in modeling high-order relations. Four types of nodes (u, q, k and a) and eight types of hyperedges are delicately designed to depict various behaviors:

3.2.1 Hyperedges for search intents. These four types of hyperedges contribute to learning desirable user and query representations to reveal the underlying search intents:

- **Global Search Hyperedge.** For each user u_i, the global search hyperedge connects all queries searched by u_i, which is utilized to learn user global preferences.
- **Local Session Hyperedge.** For each query q_i search by u_i, we construct a local session hyperedge to connect queries in the same session with q_i searched by u_i, facilitating the modeling of fine-grained search intents.
- **Query Click Hyperedge.** For each query q_i, we construct a query click hyperedge to connect q_i and the top-frequency clicked keywords under the query q_i, which can enrich the semantic representation of the single query.
- **User Click Hyperedge.** For each click behavior that u_i clicked k_i under q_i, we construct a user click hyperedge to connect (u_i, q_i, k_i), which preserves the complete search-click behaviors to model user interests.

3.2.2 Hyperedges for advertising purposes. Similarly, four types of hyperedges are also designed to model the advertising purposes:

- **Global Bid Hyperedge.** For each advertiser a_i, we construct a global bid hyperedge that connects all keywords bid by a_i to learn the representations of advertisers.
- **Local Order Hyperedge.** We represent each order as a local order hyperedge that connects all keywords belonging to the same order placed by advertisers, which advances the keyword embeddings by incorporating co-order semantics.

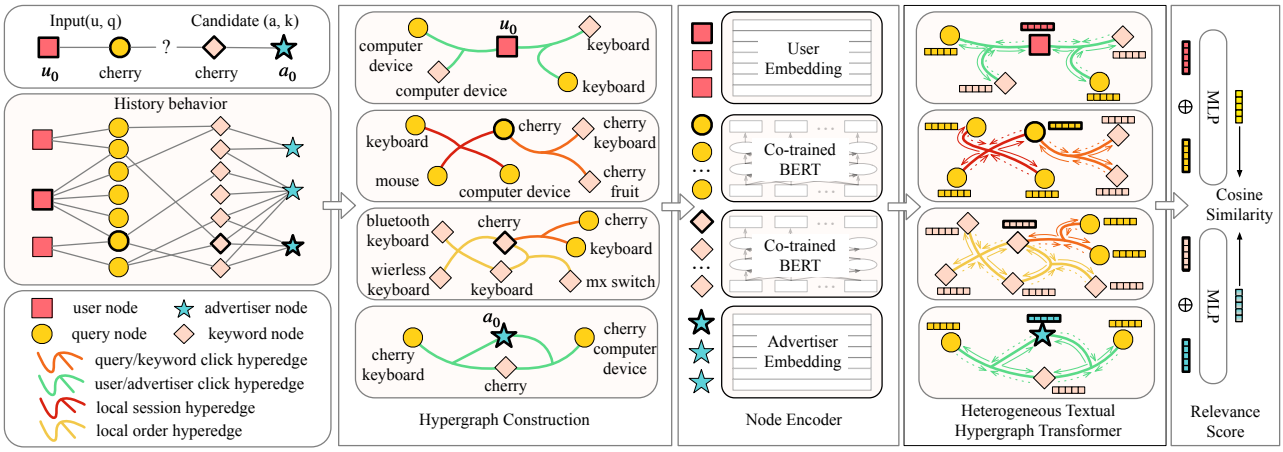


Figure 3: The overview framework of PASS model.

- **Keyword Click Hyperedge.** For keyword k_i, we construct a keyword click hyperedge to connect k_i and frequent queries that clicked k_i, providing extra semantic signals.
- **Advertiser Click Hyperedge.** For each click behavior that k_i bid by a_i was clicked under q_i, we construct the advertiser click hyperedge to connect the (a_i, k_i, q_i) to enhance the learning of advertising purposes.

Overall, the constructed heterogeneous hypergraph is formalized as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{H}\}$, and t_v, t_e are used to represent the type of v, e.

3.3 Hypergraph Sampling

The constructed hypergraph contains millions of nodes due to the enormous historical behaviors, leading to substantial resource costs. A general solution is to sample a subset of neighbors as the contextual information via uniform sampling [18]. However, the informativeness of different neighbors is varied, and thus the sampling strategy should capture the most crucial information.

Inspired by previous works [26], the Personalized PageRank (PPR) algorithm [33] is introduced as the sampling strategy. Given a traditional pairwise graph, the PPR value $\pi_t(s)$ of a target node t with respect to a source node s is defined as the probability that a α-discounted random walk starts from node s and terminates at t. The computation of PPR value $\pi_t(s)$ can be calculated by the following formula attentively:

$$\pi_t(s) = (1-\alpha) \sum_{j \in \mathcal{N}(t)} \frac{\pi_j(s)}{|\mathcal{N}(j)|} + \alpha\delta_{ts} \tag{2}$$

where δ_{ts} is the Kronecker delta that is equal to 1 if $t = s$ otherwise equal to 0. Considering multiple nodes are simultaneously connected by each hyperedge, we further generalize Eqn. (2) to calculate $\pi_t(s)$ on hypergraph as:

$$\pi_t(s) = (1-\alpha) \sum_{e \in \mathcal{E}_v(t)} \sum_{v \in \mathcal{V}_e(e)} \frac{\pi_v(s)}{|\mathcal{E}_v(t)| \cdot |\mathcal{V}_e(e)|} + \alpha\delta_{ts} \tag{3}$$

which heeds the topological structures of hyperedges and can be efficiently solved as shown in previous literature [29]. Then, the PPR importance of hyperedge e with respect to node s can be calculated by aggregating the importance of connected nodes:

$$\pi_e(s) = \sum_{v \in \mathcal{E}_v(e)} \frac{\pi_v(s)}{\sqrt{|\mathcal{E}_v(e)|}}. \tag{4}$$

Finally, a fixed number of informative hyperedges and nodes related to the input quadruples are sampled from the vanilla hypergraph based on the calculated PPR importance.

3.4 Heterogeneous Textual Hypergraph Transformer

Based on the sampled sub-hypergraph, a novel heterogeneous textual hypergraph transformer is further proposed to learn representations for the elements (i.e., u, q, a and k) within the input quadruple. The cornerstone lies in deeply fusing the textual semantics inside each node and the user behaviors within the hypergraph topology. Conventional hypergraph neural networks (HGNNs) [15, 22] generally follow a two-stage message passing paradigm as node→hyperedge→node. The semantics of nodes are firstly pre-encoded into the attribute vectors, which will be fixed in the graph aggregation. Then, attribute vectors of nodes belonging to the same hyperedge are combined together as the embedding of this hyperedge. Finally, the embeddings of hyperedges connected to the target node are fused as the final node representation.

However, directly applying such HGNNs to the studied task might be infeasible due to the following reasons. (1) Sponsored search is essentially a semantic matching task, and thus language understanding would be the nucleus component. However, the node semantic vectors in HGNNs are pre-learned and fixed, which might be irrelevant to the sponsored search task. In addition, the modeling of semantics and topology are separated by such static node attributes, resulting in inferior semantic modeling capacity. Thus, an end-to-end **NLU-HGNN co-training** model would be more promising. (2) The aggregations of existing HGNNs are conducted in the node-level, presenting all tokens within a node as an indivisible unit during message passing. However, such coarsen-grained aggregations might ignore the intricate **fine-grained correlations** between tokens of connected nodes. For example, if the semantics in token "Keyboard" can be incorporated into the token representation

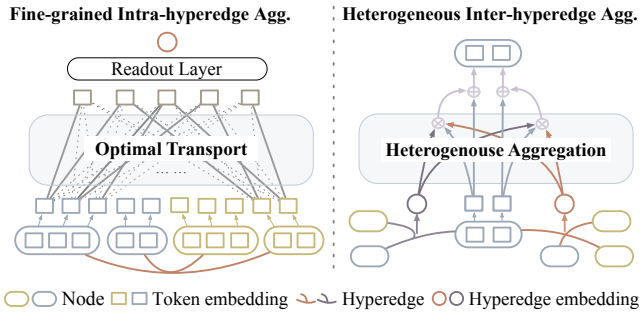


Figure 4: Framework of the two core modules in the heterogeneous textual hypergraph transformer.

of "Cherry" within the co-order hyperedges, we can achieve a better understanding of its actual meanings. Hence, an efficient token-level aggregation strategy is expected to capture the fine-grained information. (3) Most HGNNs are designed for homogeneous hypergraphs, which cannot capture the **heterogeneity** (e.g., multiple types of nodes and hyperedges) within the behavior hypergraphs.

To embrace the aforementioned challenges, we propose a novel heterogeneous textual hypergraph transformer, dubbed **HT2**. As exhibited in Figure 4, a HT2 layer consists of two core modules: the **fine-grained intra-hyperedge aggregation** to efficiently pass messages in the token level to capture the fine-grained correlations within each hyperedge; and the **heterogeneous inter-hyperedge aggregation** to fuse the heterogeneous hyperedge representations as the final node embeddings. The semantic modeling and topology mining are nested in HT2, which are learned under a unified co-training framework.

3.4.1 Fine-grained Intra-hyperedge Aggregation. The intra-hyperedge aggregation module aims to learn representations of hyperedges by capturing the fine-grained semantic correlations. Conventional HGNNs [24, 43] first learn node representations and then combine them together, whose performance is hindered by the inferior coarsen-grained node-level aggregations. A straightforward solution is to concatenate the text of all constituent nodes and then employ the self-attention to pass messages across all the tokens. Nevertheless, a single hyperedge may consist of numerous nodes, leading to the comparative long texts. Vanilla self-attention suffers from exhausting quadratic complexity when handling such long texts and disregard for heterogeneity of various node types. Inspired by Optimal Transport (OT) aggregation [35], we propose a novel OT-based heterogeneous intra-hyperedge aggregation module, which is capable of incorporating the heterogeneous fine-grained token semantics with the linear complexity.

Assume a hyperedge e and its connected textual node set $\mathcal{V}_e(e)$. Each node $v \in \mathcal{V}_e(e)$ is associated with a text of length l_v, which will be fed into a pre-trained language model to obtain contextual token representations. All contextual embeddings of tokens within nodes in $\mathcal{V}_e(e)$ form up an embedding set $\mathcal{S} = \{\mathbf{s}_i \in \mathbb{R}^d\}_{i=1}^{L}$. This language model will be co-trained with HT2. The OT aggregation is a pooling paradigm that combines relevant elements in $\mathcal{S}$ into fixed-size representations by introducing the learnable reference set $\mathcal{Z} = \{\mathbf{z}_i \in \mathbb{R}^d\}_{i=1}^{p}$. Each reference $\mathbf{z}_i$ can be viewed as a pooling cell that aggregates correlated semantics in $\mathcal{S}$, and the optimal transport polytope $\mathbf{T} \in \mathbb{R}^{L\times p}$ between $\mathcal{S}$ and $\mathcal{Z}$ will act as the aggregation weights. Given pre-calculated cost matrix $\mathbf{C} \in \mathbb{R}^{L\times p}$ and mass distribution $\mathbf{s}' \in \mathbb{R}^n, \mathbf{z}' \in \mathbb{R}^p$, $\mathbf{T}$ can be obtained by:

$$\begin{aligned} \mathbf{T} &= \underset{\mathbf{T}\in\mathbf{U}(\mathcal{S},\mathcal{Z})}{\arg\min} \langle \mathbf{C}, \mathbf{T}\rangle = \sum_{i,j} \mathbf{C}_{i,j}\mathbf{T}_{i,j}, \\ \mathbf{U}(\mathcal{S},\mathcal{Z}) &= \left\{\mathbf{T} \in \mathbb{R}_+^{L\times p} | \mathbf{T}\mathbf{1}_p = \mathbf{s}', \mathbf{T}^\top \mathbf{1}_L = \mathbf{z}'\right\}. \end{aligned} \tag{5}$$

Intuitively, $\mathcal{S}, \mathcal{Z}$ can be viewed as two discrete distributions over the semantic space. Each $\mathbf{s}_i$ can be regarded as a source point with some earth of mass $\mathbf{s}'_i$, and $\mathbf{z}_j$ can be viewed as a hole of capacity $\mathbf{z}'_j$, in which $\sum_i \mathbf{s}'_i = \sum_j \mathbf{z}'_j$. We assume $\mathbf{s}' = \frac{1}{L}\mathbf{1}_L, \mathbf{z}' = \frac{1}{p}\mathbf{1}_p$ to ensure the mass to be evenly distributed between elements [32]. $\mathbf{C}_{i,j}$ denotes a pre-defined cost that transports unit mass of earth from $\mathbf{s}_i$ to $\mathbf{z}_j$ (e.g., Euclidean distance). Essentially, the optimal polytope $\mathbf{T}$ represents a plan that turning all earth from source points into the target holes with minimal cost, which is calculated by the mass of earth moved times the moving cost. $\mathbf{T}_{i,j}$ represents the mass of the earth moved from the pile $\mathbf{s}_i$ to the hole $\mathbf{z}_j$. As a result, more related semantics between $\mathbf{s}_i$ and $\mathbf{z}_j$ will yield a higher weight $\mathbf{T}_{i,j}$ in the optimal transport polytope. Meanwhile, the calculated $\mathbf{T}$ is a sparse matrix with a maximum of $p + L - 1$ non-zero entries [8], and is self-normalized with row-sum and column-sum equal to the corresponding marginal distributions, which makes it more appealing for the studied task.

Considering representations of different nodes may fall in distinct feature spaces due to heterogeneity, we further propose the heterogeneous attention distance to calculate the matrix $\mathbf{C}$:

$$\begin{aligned} \mathbf{q}_i^z &= \mathbf{W}_z^q \mathbf{z}_i, \mathbf{k}_i^s = \mathbf{W}_{t_{s_i}}^k \mathbf{s}_i \\ \mathbf{C}_{i,j} &= 1 - \frac{\mathbf{k}_i^{s\top}\mathbf{q}_j^z}{||\mathbf{k}_i^s||_2 \cdot ||\mathbf{q}_j^z||_2} \end{aligned} \tag{6}$$

where $\mathbf{W}_z^q, \mathbf{W}_{t_{s_i}}^k \in \mathbb{R}^{d\times d}$ are learnable parameters associated with corresponding node type. Based on the calculated aggregation weights $\mathbf{T}$, relevant heterogeneous information in $\mathcal{S}$ will be pooled in different cells $\mathbf{z}$. After that, the pooled representations are concatenated and fed into a MLP readout layer to generate the representation of hyperedge e:

$$\begin{aligned} \hat{\mathbf{z}}_j &= \sum_{s_i\in\mathcal{S}} \mathbf{T}_{i,j}\mathbf{W}_{t_{s_i}}^v \mathbf{s}_i \\ \mathbf{e} &= \mathbf{W}_{t_e}[\hat{\mathbf{z}}_1; \hat{\mathbf{z}}_2; ...; \hat{\mathbf{z}}_p]. \end{aligned} \tag{7}$$

where $[;]$ denotes the concatenate operation. The OT problem in Eqn. (5) can be efficiently solved by IPOT algorithm [45]. Please refer to Appendix C for details. Benefiting from controllable size of reference set p ($<< L$) and the low-rank sparse matrix $\mathbf{T}$, such OT-based aggregation paradigm can efficiently aggregate token semantics with linear complexity in distance calculation ($O(pL)$) and information aggregation ($O(p + L - 1)$), while the vanilla self-attention suffers from quadratic complexity ($O(L^2)$).

In a nutshell, this module enjoys the following obvious merits. **1) Fine-grained semantic correlations** in the token-levels are efficiently incorporated in the OT-based aggregation. **2) Heterogeneity** is captured by type-specific projections and tightly

integrated in the aggregation paradigm. **3)** The **linear complexity** improves the efficiency of the model while maintaining similar capacity compared to self-attention in terms of relevant information aggregation. **4)** The parameters for semantic understanding are viewed as the learnable parameters, leading to a NLU-HGNN co-training paradigm.

3.4.2 Heterogeneous Inter-hyperedge Aggregation. The inter-hyperedge aggregation aims to advance node representations by aggregating topology information in connected hyperedges. To adaptively select the most informativeness neighborhood as complementary, an heterogeneous attention mechanism is proposed to aggregate connected hyperedge representations into the node representations, which aims to elevate semantic representation with task-relevant heterogeneous topology information.

Specifically, assume the target node v with its connected hyperedge set $\mathcal{E}_v(v)$. Tokens belonging to v are denoted as $\mathcal{S} = \{\mathbf{s}_i\}_{i=1}^{l_v}$. Each hyperedge $e_i \in \mathcal{E}_v(v)$ is associated with learned representation $\mathbf{e}_i$. The heterogeneous attention score between node v and connected hyperedges can be expressed as:

$$\begin{aligned} \mathbf{q}_i^v = \mathbf{W}_{t_v}^q \mathbf{s}_i, \mathbf{k}_j^e = \mathbf{W}_{t_{e_j}}^k \mathbf{e}_j \\ \mathrm{Att}(\mathbf{s}_i, \mathbf{e}_j) = \underset{e_j \in \mathcal{E}_v(v)}{\mathrm{softmax}}\left(\frac{\mathbf{q}_i^v \cdot \mathbf{k}_j^e}{\sqrt{d}}\right) \end{aligned} \tag{8}$$

Then, the representation of each token $\mathbf{s}_i$ is a weighted aggregation based on projected hyperedge representations and attention score with a shortcut connection, which can be formulated as:

$$\tilde{\mathbf{s}}_i = (1-\beta)\mathbf{s}_i + \beta \sum_{e_j \in \mathcal{E}_v(v)} \mathrm{Att}(\mathbf{s}_i, \mathbf{e}_j) \mathbf{W}_{t_{e_j}}^v \mathbf{e}_j \tag{9}$$

where $\beta \in \mathbb{R}$ is a weight between $[0, 1]$ to balance the fusion of original feature and the aggregated high-order information.

3.4.3 Application in PASS. To comprehensively incorporate the fine-grained semantics and topology information, we further consider to stack L HT2 layers and concatenate the embeddings learned by different layers to generate the final token representation:

$$\tilde{\mathbf{s}}_i = \tilde{\mathbf{W}}_{t_{s_i}} [\mathbf{s}_i^{(0)}; \mathbf{s}_i^{(1)}; ...; \mathbf{s}_i^{(L)}]. \tag{10}$$

The final node representation is the average pooling of contained token representations:

$$\tilde{\mathbf{v}} = \frac{1}{l_v} \sum_{s_i \in v} \tilde{\mathbf{s}}_i. \tag{11}$$

Specifically, we employ BERT as textual node encoder to generate semantic token representations for queries and keywords. Considering users and advertisers are generally meaningless strings with scarce semantics, we encode them by aggregating the neighboring semantics based on the global hyperedges to ensure model efficiency. Take user representations as an example, given global search hyperedge e related to user u_i, we generate representations for all $q_j \in \mathcal{V}_e(e)$ by a frozen BERT and aggregate them with mean-pooling as representation of u_i. Then, a learnable embedding lookup table $\mathbf{E}_u(u_i)$ is initialized with generated representations. Each user node can be viewed as a textual node with single token embedding in HT2 module. The advertiser embedding table $\mathbf{E}_a(a_i)$ is generated based on global bid hyperedges following similar procedure.

Algorithm 1: Forward procedure of PASS

Input: input quadruple (u_i, q_i, k_i, a_i), behavior hypergraph $\mathcal{G}$
Output: relevance score y_i between (u_i, q_i) and (a_i, k_i)

$\mathcal{G}' \leftarrow$ PPR-basedSampling$(\mathcal{G}, u_i, q_i, a_i, k_i)$
for $v \in \mathcal{V}'$ **do**
  $\mathcal{S}_v^{(0)} = \{\mathbf{s}_i^{(0)}\}_{i=1}^{l_v} \leftarrow \begin{cases} \mathrm{BERT}_q([s_0, s_1, ..., s_{l_v}]_{s_i \in v}), & \text{if } t_v \text{ is q} \\ \mathrm{BERT}_k([s_0, s_1, ..., s_{l_v}]_{s_i \in v}), & \text{if } t_v \text{ is k} \\ \{\mathbf{s}_0 = \mathbf{E}_u(v)\}, & \text{if } t_v \text{ is u} \\ \{\mathbf{s}_0 = \mathbf{E}_a(v)\}, & \text{if } t_v \text{ is a} \end{cases}$
end for
for $l \leftarrow 1, 2, ...$ **do**
  $\mathbf{e}^{(l)} \leftarrow$ Intra-AGG$(\{\mathcal{S}_v^{(l-1)} | v \in \mathcal{V}_e(e)\})$
  $\mathcal{S}_v^{(l)} \leftarrow$ Inter-AGG$(\{\mathbf{e}^{(l)} | e \in \mathcal{E}_v(v)\})$
end for
for $v \in \mathcal{V}'$ **do**
  $\tilde{\mathbf{v}} = \frac{1}{l_v} \sum_{s_i \in v} \tilde{\mathbf{W}}[\mathbf{s}_i^{(0)}; \mathbf{s}_i^{(1)}; ...; \mathbf{s}_i^{(L)}]$
end for
$\mathbf{r}_i, \mathbf{p}_i \leftarrow \mathbf{W}^r[\tilde{\mathbf{u}}_i; \tilde{\mathbf{q}}_i], \mathbf{W}^p[\tilde{\mathbf{a}}_i; \tilde{\mathbf{k}}_i]$
$y_i \leftarrow \frac{(\mathbf{r}_i \cdot \mathbf{p}_i)}{||\mathbf{r}_i||_2 \cdot ||\mathbf{p}_i||_2}$
return y_i

With the proposed heterogeneous textual hypergraph transformer, we can obtain the following enhanced node representations. **1) Preference preserved user and advertiser representations**. HT2 is capable of extracting the preferences of users and advertisers from heterogeneous hypergraph, and further encoding them into learned representations. **2) Personalized query representations and advertiser-aware keywords representations**. The pure semantic representations of queries (keywords) will be advanced with personalized (advertiser-aware) information modeled in the local behavior hyperedges. Such behavior-aware representations would contribute to providing comprehensive search and advertising intents for relevance modeling.

3.5 Relevance Module

After aggregating the high-order topological correlations, we directly fuse the representations of nodes in the input quadruple to achieve the underlying search intent and advertising purpose. Specifically, the generated representations $(\tilde{\mathbf{u}}_i, \tilde{\mathbf{q}}_i)$ and $(\tilde{\mathbf{a}}_i, \tilde{\mathbf{k}}_i)$ will be concatenated and fed into a MLP layer to generate search intent representation $\mathbf{r}_i$ and advertising purpose representation $\mathbf{p}_i$, respectively, which can be formulated as follows:

$$\mathbf{r}_i = \mathbf{W}^r[\tilde{\mathbf{u}}_i; \tilde{\mathbf{q}}_i], \ \mathbf{p}_i = \mathbf{W}^p[\tilde{\mathbf{a}}_i; \tilde{\mathbf{k}}_i] \tag{12}$$

where $\mathbf{W}^r, \mathbf{W}^p \in \mathbb{R}^{d \times 2d}$ denote the learnable parameters of MLP layers. The final relevance score y_i can be measured by cosine similarity between $\mathbf{r}_i$ and $\mathbf{p}_i$:

$$y_i = \frac{(\mathbf{r}_i \cdot \mathbf{p}_i)}{||\mathbf{r}_i||_2 \cdot ||\mathbf{p}_i||_2}. \tag{13}$$

The overview algorithm of calculating relevance score between input (u_i, q_i) and (a_i, k_i) is presented in Algorithm 1.

3.6 Objective Function

We adopt a popular pair-wise ranking loss, Bayesian Personalized Ranking (BPR) [36], to optimize parameters in PASS. The in-batch

Table 1: Statistics of the datasets.

	#train	#val	#test	#user	#adver.
Product-Ads	1,977,560	199,815	200,976	149,503	105,405
Textual-Ads	9,068,716	751,816	751,987	486,799	214,738

negative sampling strategy [11, 41] is employed to improve training efficiency. Specifically, given a batch of positive quadruples $\mathcal{B} = \{((u_i, q_i), (a_i, k_i))\}_{i=1}^{n}$, we can generate corresponding search intent representations and advertising representations $\{(\mathbf{r}_i, \mathbf{p}_i)\}_{i=1}^{n}$. For each $(\mathbf{r}_i, \mathbf{p}_i)$, the other advertising purpose representations in the same batch $\{\mathbf{p}_j | j \neq i\}$ will be viewed as negative set $\mathcal{N}^-$. Then, the BPR loss can be calculated as follows:

$$\begin{aligned}\mathcal{L}(\mathcal{B}) &= -\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|\mathcal{N}^-|}\sum_{\mathbf{p}_i^- \in \mathcal{N}^-}\log\delta(\mathrm{c}(\mathbf{r}_i, \mathbf{p}_i) - \mathrm{c}(\mathbf{r}_i, \mathbf{p}_i^-)) \\ &= -\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\log\delta(\mathrm{c}(\mathbf{r}_i, \mathbf{p}_i) - \mathrm{c}(\mathbf{r}_i, \mathbf{p}_j)),\end{aligned} \tag{14}$$

where c is cosine similarity, and δ denotes the sigmoid function.

4 EXPERIMENTS

4.1 Experimental Settings

4.1.1 Dataset. Our proposal is extensively evaluated on two real-world industry datasets collected from different business scenarios within Bing Ads, including **Product-Ads** and **Textual-Ads**. Both datasets are constructed based on historical click records in three months from March 1st, 2022 to June 1st, 2022. Each sample is a quadruple (u, q, k, a), denoting that user u clicked keyword k bid by advertiser a under the input query q. Here is an example of quadruple in the dataset: (dc1f-2423-ab23-f3e3, cherry keyboard, keyboard, newegg.com). Inactive users with less than three interactions are removed, which account for 14% of all users in Product-Ads and 12% in Textual-Ads. For each dataset, the quadruples occurred in the final ten days are randomly split into the validation and test set, and the rest instances are viewed as the training set. Table 1 presents the detailed statistics of datasets. Meanwhile, other essential information of behaviors like sessions and orders are also collected to construct the entire hypergraph.

4.1.2 Baselines. Three types of popular approaches are selected as the baseline methods.

Semantic-based Models: This type of methods matches queries and keywords solely relying on the textual semantics.

- **C-DSSM** [39] employs a convolutional pooling structure over word sequences to learn embeddings.
- **TwinBERT** [30] is a twin-structured encoder based on the pre-trained BERT model.

Graph-based Models: Following previous works [18, 27], historical behaviors are formatted into a pairwise graph. The attribute vectors of nodes (queries and keywords) are pre-learned by language models and will be fixed during the training of GNNs [18].

- **GAT** [18] employs the multi-head attention to aggregate information from neighborhoods.
- **HGT** [20] utilizes type-dependent parameters to model heterogeneous attention across edges.

Hybrid Models: Textual semantics and historical behaviors are fused under the NLU-GNN co-training paradigm.

- **TextGNN** [50] incorporates the text and graph information under a BERT-GAT co-training framework.
- **AdsGNN** [27] fuses textual semantics and click graph from various perspectives. Here we select the best variation AdsGNN$_t$ which merges information in the token level.
- **BGTR** [6] utilizes the bidding graphs to learn desirable advertiser and keyword representations.
- **HBGLR** [18] encodes semantics and structural heterogeneity into node representations with learnable structure.

We also implement an additional version for hybrid models to handle the quadruple-based task by adding user and advertiser signals (e.g., TextGNN$_{ua}$) for fairness. The additional user and advertiser representations have the same generation and training procedure with our proposal and are fused in relevance module to learn search intent and advertising purpose representations.

4.1.3 Implementation details and Evaluations. For the experimental settings, we use "bert-base-uncased" as the pre-trained BERT model for all methods. The max sequence length for each sentence is set to 15. The hyper-parameters of all methods are carefully tuned on the validation set, and we report the performance of best validation settings on the test set. For the proposed PASS model, the dimension of hidden states d is 768, the number of sampled hyperedges for each node is set to 3 and each hyperedge contains 5 nodes, the shortcut connection weight β is set to 0.5, the number of HT2 layer is set to 2. We employ IPOT [45] to solve the optimal transport problem and size of reference set p is 5. The size of training batch is 256 and the number of training epochs is 15 with early stop. Adam optimizer is employed to minimize the training loss. More details are presented in Appendix D. We evaluate all methods with recall task, which aims at retrieving top relevant candidate (a, k) with the input (u, q) pair. All (a, k) pairs are viewed as the candidates set and top 5/20/100 results are selected for **Recall** and **NDCG** (Normalized Discounted Cumulative Gain) evaluation, which are commonly used to measure ranking quality in retrival task [30].

4.2 Main Results

Table 2 presents the recall performance of different methods. One can easily achieve the following observations: (1) GNNs are the least effective methods, which is reasonable as the node textual attributes are fixed during training, leading to the inferior expressiveness in terms of semantic matching. (2) Hybrid models generally outperform the pure semantic-based ones, verifying the effectiveness of historical behaviors in user intent understanding. (3) The performance of all methods consistently increases by integrating extra user and advertiser representations, which indicates the signals of users and advertisers contributes to facilitating the sponsored search performance. (4) Our proposal significantly surpasses baselines over all evaluation metrics. PASS deeply fuses semantics and behaviors with powerful heterogeneous hypergraphs, which is capable of precisely modeling search intents and advertising purposes and achieving promising performance. In addition, a slightly distilled version of our PASS model has already been deployed in recall stage of Bing Ads and demonstrates significant performance gains, detailed in Appendix E.

Table 2: Results on Product-Ads and Textual-Ads. Best results are in bold and second best results are underlined. The improvements are statistically significant (sign test, p-value < 0.01).

	Product-Ads						Textual-Ads					
	Recall@k			NDCG@k			Recall@k			NDCG@k		
Model	k=5	k=20	k=100	k=5	k=20	k=100	k=5	k=20	k=100	k=5	k=20	k=100
C-DSSM	$0.090_{\pm 0.010}$	$0.150_{\pm 0.006}$	$0.289_{\pm 0.009}$	$0.071_{\pm 0.007}$	$0.081_{\pm 0.008}$	$0.108_{\pm 0.008}$	$0.063_{\pm 0.009}$	$0.107_{\pm 0.005}$	$0.229_{\pm 0.008}$	$0.045_{\pm 0.010}$	$0.052_{\pm 0.007}$	$0.078_{\pm 0.010}$
TwinBERT	$0.101_{\pm 0.011}$	$0.165_{\pm 0.007}$	$0.295_{\pm 0.011}$	$0.086_{\pm 0.015}$	$0.093_{\pm 0.010}$	$0.118_{\pm 0.005}$	$0.074_{\pm 0.007}$	$0.127_{\pm 0.007}$	$0.255_{\pm 0.011}$	$0.059_{\pm 0.010}$	$0.069_{\pm 0.004}$	$0.093_{\pm 0.006}$
GAT	$0.060_{\pm 0.020}$	$0.114_{\pm 0.026}$	$0.251_{\pm 0.018}$	$0.043_{\pm 0.021}$	$0.055_{\pm 0.017}$	$0.083_{\pm 0.025}$	$0.037_{\pm 0.024}$	$0.075_{\pm 0.030}$	$0.196_{\pm 0.028}$	$0.019_{\pm 0.025}$	$0.025_{\pm 0.022}$	$0.048_{\pm 0.017}$
HGT	$0.082_{\pm 0.017}$	$0.141_{\pm 0.020}$	$0.279_{\pm 0.020}$	$0.064_{\pm 0.019}$	$0.077_{\pm 0.017}$	$0.105_{\pm 0.028}$	$0.056_{\pm 0.019}$	$0.102_{\pm 0.030}$	$0.223_{\pm 0.020}$	$0.038_{\pm 0.015}$	$0.047_{\pm 0.024}$	$0.074_{\pm 0.021}$
TextGNN	$0.117_{\pm 0.011}$	$0.192_{\pm 0.014}$	$0.331_{\pm 0.016}$	$0.096_{\pm 0.015}$	$0.122_{\pm 0.011}$	$0.149_{\pm 0.014}$	$0.097_{\pm 0.011}$	$0.151_{\pm 0.018}$	$0.277_{\pm 0.011}$	$0.078_{\pm 0.013}$	$0.089_{\pm 0.015}$	$0.112_{\pm 0.020}$
TextGNN$_{ua}$	$0.130_{\pm 0.015}$	$0.207_{\pm 0.011}$	$0.341_{\pm 0.009}$	$0.107_{\pm 0.010}$	$0.134_{\pm 0.009}$	$0.158_{\pm 0.009}$	$0.104_{\pm 0.015}$	$0.161_{\pm 0.017}$	$0.285_{\pm 0.012}$	$0.089_{\pm 0.015}$	$0.097_{\pm 0.018}$	$0.120_{\pm 0.010}$
AdsGNN	$0.122_{\pm 0.013}$	$0.198_{\pm 0.015}$	$0.337_{\pm 0.013}$	$0.101_{\pm 0.012}$	$0.125_{\pm 0.013}$	$0.150_{\pm 0.010}$	$0.101_{\pm 0.011}$	$0.156_{\pm 0.015}$	$0.287_{\pm 0.017}$	$0.085_{\pm 0.009}$	$0.094_{\pm 0.014}$	$0.119_{\pm 0.020}$
AdsGNN$_{ua}$	$0.137_{\pm 0.014}$	$0.212_{\pm 0.012}$	$0.348_{\pm 0.014}$	$0.113_{\pm 0.012}$	$0.138_{\pm 0.012}$	$0.162_{\pm 0.010}$	$0.119_{\pm 0.017}$	$0.173_{\pm 0.013}$	$0.306_{\pm 0.015}$	$0.101_{\pm 0.014}$	$0.109_{\pm 0.012}$	$0.136_{\pm 0.017}$
BGTR	$0.121_{\pm 0.013}$	$0.196_{\pm 0.011}$	$0.336_{\pm 0.012}$	$0.101_{\pm 0.011}$	$0.127_{\pm 0.009}$	$0.153_{\pm 0.016}$	$0.103_{\pm 0.011}$	$0.158_{\pm 0.015}$	$0.286_{\pm 0.016}$	$0.087_{\pm 0.012}$	$0.095_{\pm 0.014}$	$0.118_{\pm 0.016}$
BGTR$_{ua}$	$0.142_{\pm 0.009}$	$0.216_{\pm 0.010}$	$0.350_{\pm 0.014}$	$0.115_{\pm 0.015}$	$0.140_{\pm 0.016}$	$0.161_{\pm 0.011}$	$0.121_{\pm 0.013}$	$0.177_{\pm 0.014}$	$0.308_{\pm 0.011}$	$0.102_{\pm 0.012}$	$0.112_{\pm 0.010}$	$0.138_{\pm 0.015}$
HBGLR	$0.136_{\pm 0.015}$	$0.210_{\pm 0.014}$	$0.344_{\pm 0.018}$	$0.112_{\pm 0.012}$	$0.138_{\pm 0.018}$	$0.159_{\pm 0.019}$	$0.117_{\pm 0.016}$	$0.167_{\pm 0.012}$	$0.293_{\pm 0.012}$	$0.095_{\pm 0.016}$	$0.104_{\pm 0.018}$	$0.127_{\pm 0.014}$
HBGLR$_{ua}$	$\underline{0.150}_{\pm 0.010}$	$\underline{0.225}_{\pm 0.015}$	$\underline{0.359}_{\pm 0.014}$	$\underline{0.121}_{\pm 0.014}$	$\underline{0.147}_{\pm 0.015}$	$\underline{0.169}_{\pm 0.019}$	$\underline{0.128}_{\pm 0.010}$	$\underline{0.185}_{\pm 0.016}$	$\underline{0.319}_{\pm 0.012}$	$\underline{0.110}_{\pm 0.011}$	$\underline{0.121}_{\pm 0.017}$	$\underline{0.147}_{\pm 0.016}$
PASS	$\mathbf{0.169}_{\pm 0.008}$	$\mathbf{0.247}_{\pm 0.011}$	$\mathbf{0.380}_{\pm 0.005}$	$\mathbf{0.137}_{\pm 0.009}$	$\mathbf{0.164}_{\pm 0.007}$	$\mathbf{0.190}_{\pm 0.005}$	$\mathbf{0.140}_{\pm 0.004}$	$\mathbf{0.199}_{\pm 0.014}$	$\mathbf{0.335}_{\pm 0.010}$	$\mathbf{0.119}_{\pm 0.009}$	$\mathbf{0.132}_{\pm 0.010}$	$\mathbf{0.160}_{\pm 0.008}$

Table 3: Ablation study on hyperedges.

Model	Product-Ads R@20	Product-Ads R@100	Textual-Ads R@20	Textual-Ads R@100
PASS	**0.247**	**0.380**	**0.199**	**0.335**
w/o session-edge	0.234	0.362	0.188	0.321
w/o order-edge	0.227	0.355	0.184	0.315
w/o u/a-click-edge	0.240	0.370	0.193	0.326
w/o q/k-click-edge	0.230	0.356	0.185	0.317

Table 4: Ablation study on hypergraph sampling.

	Product-Ads R@20	Product-Ads R@100	Textual-Ads R@20	Textual-Ads R@100
Uniform sampling	0.233	0.364	0.187	0.322
Degree-based sampling	0.239	0.371	0.193	0.328
PPR-based sampling	**0.247**	**0.380**	**0.199**	**0.335**

4.3 Ablation Study

In this subsection, extensive ablation studies are conducted to investigate the effectiveness of various components in PASS.

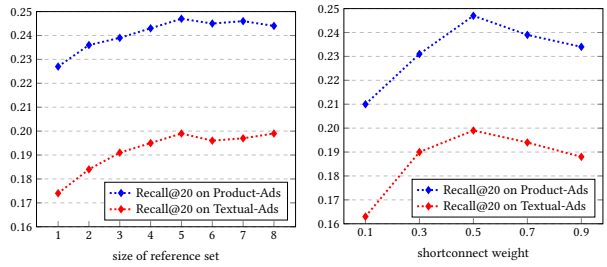
4.3.1 Different types of hyperedges. Here we aim to investigate the importance of different types of hyperedges. Different types of hyperedges are removed from the hypergraph, and model performance on the retained hypergraph is reported. Table 3 presents the experimental results. (1) Without the session or order hyperedges, the performance significantly drops by 0.013/0.011 on R@20. Local behaviors (co-session and co-order data) depict the contextual semantics of the queries and keywords, which provides indispensable complementary information to learn personalized representations of queries and keywords. (2) The performance of PASS also declines after removing the click related hyperedges, revealing that click behaviors are crucial in learning desirable user/advertiser representations. Overall, the recall performance of PASS consistently declines after removing any type of hyperedges, indicating that different hyperedges are capable of capturing unique valuable behavior information to facilitate semantic matching.

Table 5: Ablation study on HT2 module.

Model	Product-Ads R@20	Product-Ads Time(h/epoch)	Textual-Ads R@20	Textual-Ads Time(h/epoch)
Nodel-level aggregation	0.223	**1.1**	0.180	**5.2**
Homogeneous attention	0.233	1.5	0.188	8.4
Mean aggregation	0.231	1.3	0.189	6.5
Sparse self-attention	0.240	1.6	0.192	8.6
Full self-attention	**0.249**	8.5	0.198	44.8
PASS	0.247	1.8	**0.199**	9.2

4.3.2 Hypergraph sampling strategy. Here we study the impact of the proposed PPR-based sampling strategy. The conventional uniform [18] and degree-based sampling strategies [20] are selected as the baselines. Table 4 presents the experimental results. PPR-based sampling outperforms the uniform sampling and degree-based sampling by 0.014/0.012 and 0.008/0.006 on R@20, respectively. The PPR-based sampling strategy ensures that the sampled subgraph preserves the informative neighbors and avoids the potential noise in message passing. Thus, our proposal is capable of ensuring the sampled subgraph are more stable and informative during training.

4.3.3 Heterogeneous textual hypergraph transformer. Here we study the effectiveness and efficiency of proposed HT2 module with different ablation variants. Table 5 presents the experimental results. First, we perform coarsen-grained node-level aggregations by averaging token embeddings within each node as its representation in HT2 module. The lack of fine-grained token-level semantics leads to a significant drop in R@20 performance by 0.024/0.019 on two datasets, indicating that token-level aggregations facilitate the intent understanding. Second, the heterogeneity-related components



(a) Performance vs. p (b) Performance vs. β

Figure 5: Parameter sensitivity analysis.

Table 6: Evaluation in cold start scenario.

Model	Product-Ads R@20	Product-Ads R@100	Textual-Ads R@20	Textual-Ads R@100
AdsGNN$_{ua}$	0.202	0.338	0.161	0.286
HBGLR$_{ua}$	0.215	0.349	0.170	0.299
PASS	**0.225**	**0.359**	**0.180**	**0.309**

Table 7: Model efficiency analysis.

	AdsGNN$_{ua}$	HBGLR$_{ua}$	PASS
Training (h/epoch)	1.6	2.3	1.8
Inference (ms/batch)	785	791	804

are replaced by the homogeneous counterparts, dubbed homogeneous attention. Model performance also significantly drops, which reveals that the types of nodes and hyperedges are crucial indicators to fuse messages from different feature spaces. Third, the OT-based aggregation is replaced by different fusion strategies. Mean aggregation directly averages all token embeddings to generate hyperedge representations. Full self-attention [42] and sparse self-attention [12] are employed on the concatenated tokens respectively to capture the semantic correlations. The lowest performance of mean aggregation reveals that the correlations between tokens are crucial to the accurate semantic understanding. Compared with the heuristic assumption based sparse self-attention, the proposed OT-based aggregation achieves better performance, crediting to the theoretical guarantees of optimal transport solver. Compared with the full self-attention variant, PASS obtains comparable performance while greatly reducing training time, demonstrating the efficiency of the OT-based aggregation.

4.4 Parameter Sensitivity Analysis

We study the performance sensitivity of PASS about two core hyper-parameters: size of reference set p in OT-aggregation and the shortcut connection weight β. Figure 5(a) demonstrates the performance curves of p. With the increases of p, the performance of PASS first increases and then keeps steady. This is reasonable as a larger number of reference cells will empower PASS with stronger modeling capability to capture more comprehensive semantics. When p is getting larger, the expressive ability of aggregation becomes sufficient, while more learnable parameters may lead to slow and unstable training, resulting in inconspicuous gain and slight fluctuations. From Figure 5(b), the performance of PASS first increases and then decreases when we increase short connection weight β from 0.1 to 0.9. A larger β introduces more heterogeneous topology information in token representations, which provides supplement semantics with task-relevant behaviors, while suffering from enlarging risk of introducing noises due to the uncertainty of user behaviors. Thus, these two hyper-parameters should be delicately tuned for desirable model performance.

4.5 Performance in Cold Start Scenario

In the personalized advertiser-aware sponsored search, the cold start problem refers to a situation where new emerging users has limited historical behaviors. Here we propose a simple but effective strategy to generate representations for new users and evaluate the effectiveness of our proposal under cold start setting. Representations of emerging users are initialized as the average of active users' embeddings pre-learned by PASS. We further collect the historical clicks from inactive users (i.e., with less than three search records) as the cold start evaluation dataset, leading to 38,706 samples for Product-Ads and 182,461 instances for Textual-Ads. All methods are trained on the normal datasets and evaluated on the cold start datasets. Table 6 displays the experimental results of PASS and several strong hybrid baselines. User-aware models present significant performance decline in cold start setting, hindered by the learned inferior user representations. PASS still outperforms all baselines, which may benefit from the powerful and comprehensive understanding of semantics within queries and keywords.

4.6 Model Efficiency Analysis

Here we present the training and inference time cost of our proposal on Product-Ads dataset. Two SOTA NLU-GNN co-training hybrid models, AdsGNN$_{ua}$ and HBGLR$_{ua}$, are adopted for comparison. The experiments are conducted on a machine with one single Nvidia-A100-80GB GPU, an Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.60GHz 2600 MHz CPU, and 112 GB memory. Table 7 reports the training and inference time of different models. The batch size for inference is set to 256. Experimental results demonstrate that the time consuming of PASS is comparable to baselines, which is affordable considering its effectiveness.

5 CONCLUSION

In this paper, we study the novel problem of personalized advertiser-aware sponsored search. To fully uncover the personalized and diverse preferences of users and advertisers, we propose to model the historical behaviors in the format of hypergraphs to provide complementary high-order information. Furthermore, a well-defined heterogeneous textual hypergraph transformer is proposed to deeply fuse the fine-grained semantic information and the heterogeneous hypergraph topology. Extensive experiments demonstrate that our proposal consistently outperforms SOTA approaches under both offline and online scenarios.

ACKNOWLEDGMENTS

This work was supported by the NSFC No. 62172443.

REFERENCES

[1] Jason M. Altschuler, Jonathan Weed, and Philippe Rigollet. 2017. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 1964–1974.

[2] Song Bai, Feihu Zhang, and Philip H. S. Torr. 2021. Hypergraph convolution and hypergraph attention. *Pattern Recognit.* 110 (2021), 107637.

[3] Xiao Bai, Erik Ordentlich, Yuanyuan Zhang, Andy Feng, Adwait Ratnaparkhi, Reena Somvanshi, and Aldi Tjahjadi. 2018. Scalable Query N-Gram Embedding for Improving Matching and Relevance in Sponsored Search. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*. ACM, 52–61.

[4] James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research* 13 (2012), 281–305.

[5] Keping Bi, Qingyao Ai, and W. Bruce Croft. 2020. A Transformer-based Embedding Model for Personalized Product Search. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*. ACM, 1521–1524.

[6] Shuxian Bi, Chaozhuo Li, Xiao Han, Zheng Liu, Xing Xie, Haizhen Huang, and Zengxuan Wen. 2021. Leveraging Bidding Graphs for Advertiser-Aware Relevance Modeling in Sponsored Search. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*. Association for Computational Linguistics, 2215–2224.

[7] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2001. Latent Dirichlet Allocation. In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*. MIT Press, 601–608.

[8] Richard A. Brualdi. 2006. *Combinatorial Matrix Classes*. Cambridge University Press. https://doi.org/10.1017/CBO9780511721182

[9] Liqun Chen, Guoyin Wang, Chenyang Tao, Dinghan Shen, Pengyu Cheng, Xinyuan Zhang, Wenlin Wang, Yizhe Zhang, and Lawrence Carin. 2019. Improving Textual Network Embedding with Global Attention via Optimal Transport. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5193–5202.

[10] Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Improving Sequence-to-Sequence Learning via Optimal Transport. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[11] Ting Chen, Yizhou Sun, Yue Shi, and Liangjie Hong. 2017. On Sampling Strategies for Neural Network-based Collaborative Filtering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 767–776.

[12] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating Long Sequences with Sparse Transformers. *CoRR* abs/1904.10509 (2019).

[13] Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* 26 (2013).

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 4171–4186.

[15] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph Neural Networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 3558–3565.

[16] Jianfeng Gao, Patrick Pantel, Michael Gamon, Xiaodong He, and Li Deng. 2014. Modeling Interestingness with Deep Neural Networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2–13.

[17] Anindya Ghose and Sha Yang. 2009. An Empirical Analysis of Search Engine Advertising: Sponsored Search in Electronic Markets. *Manag. Sci.* 55, 10 (2009), 1605–1622.

[18] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. 1024–1034.

[19] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional Neural Network Architectures for Matching Natural Language Sentences. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 2042–2050.

[20] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous Graph Transformer. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 2704–2710.

[21] 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 *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*. ACM, 2553–2561.

[22] Jing Huang and Jie Yang. 2021. UniGNN: a Unified Framework for Graph and Hypergraph Neural Networks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, Zhi-Hua Zhou (Ed.). ijcai.org, 2563–2569.

[23] Glen Jeh and Jennifer Widom. 2003. Scaling personalized web search. In *Proceedings of the Twelfth International World Wide Web Conference, WWW 2003, Budapest, Hungary, May 20-24, 2003*. ACM, 271–279.

[24] Shuyi Ji, Yifan Feng, Rongrong Ji, Xibin Zhao, Wanwan Tang, and Yue Gao. 2020. Dual Channel Hypergraph Collaborative Filtering. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*. ACM, 2020–2029.

[25] Jianwen Jiang, Yuxuan Wei, Yifan Feng, Jingxuan Cao, and Yue Gao. 2019. Dynamic Hypergraph Neural Networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*. ijcai.org, 2635–2641.

[26] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. 2019. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[27] Chaozhuo Li, Bochen Pang, Yuming Liu, Hao Sun, Zheng Liu, Xing Xie, Tianqi Yang, Yanling Cui, Liangjie Zhang, and Qi Zhang. 2021. AdsGNN: Behavior-Graph Augmented Relevance Modeling in Sponsored Search. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*. ACM, 223–232.

[28] Xiaodan Liang, Hongfei Zhou, and Eric P. Xing. 2018. Dynamic-Structured Semantic Propagation Network. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 752–761.

[29] Peter Lofgren and Ashish Goel. 2013. Personalized PageRank to a Target Node. *CoRR* abs/1304.4658 (2013).

[30] Wenhao Lu, Jian Jiao, and Ruofei Zhang. 2020. TwinBERT: Distilling Knowledge to Twin-Structured Compressed BERT Models for Large-Scale Retrieval. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*. ACM, 2645–2652.

[31] Jing Ma, Mengting Wan, Longqi Yang, Jundong Li, Brent J. Hecht, and Jaime Teevan. 2022. Learning Causal Effects on Hypergraphs. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*. ACM, 1202–1212.

[32] Grégoire Mialon, Dexiong Chen, Alexandre d'Aspremont, and Julien Mairal. 2021. A Trainable Optimal Transport Embedding for Feature Aggregation and its Relationship to Attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

[33] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank citation ranking: Bringing order to the web*. Technical Report. Stanford InfoLab.

[34] Bochen Pang, Chaozhuo Li, Yuming Liu, Jianxun Lian, Jianan Zhao, Hao Sun, Weiwei Deng, Xing Xie, and Qi Zhang. 2022. Improving Relevance Modeling via Heterogeneous Behavior Graph Learning in Bing Ads. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*. ACM, 3713–3721.

[35] Gabriel Peyré and Marco Cuturi. 2019. Computational Optimal Transport. *Found. Trends Mach. Learn.* 11, 5-6 (2019), 355–607.

[36] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*. 452–461.

[37] Ruslan Salakhutdinov and Geoffrey E. Hinton. 2009. Semantic hashing. *Int. J. Approx. Reason.* 50, 7 (2009), 969–978.

[38] Tim Salimans, Han Zhang, Alec Radford, and Dimitris N. Metaxas. 2018. Improving GANs Using Optimal Transport. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

[39] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*. ACM, 101–110.

[40] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for

web search. In *23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014, Companion Volume*. ACM, 373–374.

[41] Zhoujin Tian, Chaozhuo Li, Shuo Ren, Zhiqiang Zuo, Zengxuan Wen, Xinyue Hu, Xiao Han, Haizhen Huang, Denvy Deng, Qi Zhang, et al. 2022. RAPO: An Adaptive Ranking Paradigm for Bilingual Lexicon Induction. *arXiv preprint arXiv:2210.09926* (2022).

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. 5998–6008.

[43] Lianghao Xia, Chao Huang, and Chuxu Zhang. 2022. Self-Supervised Hypergraph Transformer for Recommender Systems. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*. ACM, 2100–2109.

[44] Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui. 2021. Self-Supervised Graph Co-Training for Session-based Recommendation. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*. ACM, 2180–2190.

[45] Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. 2019. A Fast Proximal Point Method for Computing Exact Wasserstein Distance. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019 (Proceedings of Machine Learning Research, Vol. 115)*, Amir Globerson and Ricardo Silva (Eds.). AUAI Press, 433–453.

[46] Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. 2021. GraphFormers: GNN-nested transformers for representation learning on textual graph. *Advances in Neural Information Processing Systems* 34 (2021), 28798–28810.

[47] Liu Yang, Qingyao Ai, Jiafeng Guo, and W. Bruce Croft. 2016. aNMM: Ranking Short Answer Texts with Attention-Based Neural Matching Model. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*. ACM, 287–296.

[48] Junliang Yu, Hongzhi Yin, Jundong Li, Qinyong Wang, Nguyen Quoc Viet Hung, and Xiangliang Zhang. 2021. Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. ACM / IW3C2, 413–424.

[49] Yujia Zhou, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2021. PSSL: Self-supervised Learning for Personalized Search with Contrastive Sampling. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*. ACM, 2749–2758.

[50] Jason Zhu, Yanling Cui, Yuming Liu, Hao Sun, Xue Li, Markus Pelger, Tianqi Yang, Liangjie Zhang, Ruofei Zhang, and Huasha Zhao. 2021. TextGNN: Improving Text Encoder via Graph Neural Network in Sponsored Search. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. ACM / IW3C2, 2848–2857.

A NOTATIONS

For the sake of clarification, notations used in this paper are listed in Table 8.

Table 8: Notations used in this paper.

Symbol	Shape	Description
u, q, k, a	-	a sample of user, query, keyword, advertiser
$\mathcal{U}_i$	-	historical behavior set of u_i
$\mathcal{A}_i$	-	historical behavior set of a_i
$\mathcal{G}$	-	hypergraph
v, e	-	node and hyperedge in hypergraph
t_v, t_e	-	type of node v and hyperedge e
$\mathcal{V}$	-	set of nodes in $\mathcal{G}$
$\mathcal{E}$	-	set of hyperedges in $\mathcal{G}$
$\mathbf{H}$	$\mathbb{R}^{\lvert\mathcal{V}\rvert\times\lvert\mathcal{E}\rvert}$	incidence matrix about $\mathcal{G}$
$\mathcal{E}_v(v)$	-	the set of hyperedges connected to v
$\mathcal{V}_e(e)$	-	the set of nodes connected to e
d	$\mathbb{R}$	embedding size of hidden states
p	$\mathbb{R}$	size of reference set in HT2
β	$\mathbb{R}$	shortcut connection weight

B RELATED WORK

B.1 Relevance Modeling in Sponsored Search

Existing methods usually view queries and keywords as carriers of search intents and advertising purposes and formulate this problem as text matching task between the query-keyword tuples. Following successful attempts in information retrieval (IR), traditional approaches usually adopt shallow language representation learning models such as LSA [37] and LDA [7], which can map sentences to low-dimensional continuous vectors to calculate similarity. In recent years, deep semantic models, especially the siamese structure models, have been adopted in a range of works [16, 19, 39, 40] in sponsored search tasks. C-DSSM [39] is a latent semantic model incorporating a convolutional-pooling structure over word sequences to learn representations for queries and keywords. TwinBERT [30] has twin-structured BERT-like encoders to represent query and document, respectively. Apart from the textual information, some related works attempt to incorporate other types of data in sponsored search scenarios. ANMM [47] proposes to fuse the textual, visual, and relational signals to learn the compositional representations. TextGNN [50] extends the TwinBERT model to incorporate the search log data as complementary. AdsGNN [27] aims to extensively investigate how to naturally fuse the textual input and the user behavior graph to enhance the semantic representations. BGTR [6] proposes to utilize the bidding graphs to learn desirable advertiser representations and explore the order information to enhance the representation of keywords. HBGLR [18] encodes textual semantics and structural heterogeneity into node representations with learnable graph structure. Most of them only focus on enriching the semantic information in the short queries and keywords, which may not precisely reflect the search intents and advertising purposes due to the diverse and individual preferences of users and advertisers. To our best knowledge, we are the first to study the novel problem of personalized advertiser-aware sponsored search, which investigates the historical behaviors of user and advertiser to model the underlying search intents and advertising purposes. Furthermore, we introduce the hypergraphs instead of the traditional pairwise graphs as the backbones and propose an effective heterogeneous textual hypergraph transformer, which can capture the complex high-order relations and effectively aggregate the personalized and advertiser-aware information to facilitate the downstream relevance tasks.

B.2 Hypergraph Neural Networks

Recently, numerous works have explored hypergraph neural networks to handle hypergraph structures for downstream tasks. HGNN [15] proposes a hyperedge convolution operation to encode high-order data correlation in hypergraph structure for representation learning. Inspired by effective graph attention neural network, a trainable hypergraph attention mechanism [2] is further proposed to learn node representations in hypergraphs. DHGNN [25] proposes a dynamic hypergraph neural network, which dynamically updates the hypergraph structure on each layer. With these effective hypergraph representation learning methods, some recent works propose employing HGNNs to advance recommendations' performance. HyRec [43] regards users as hyperedges to aggregate information from the interacted items. MHCN [48] constructs multi-channel hypergraphs to model high-order relationships among users. DHCF [24] is a hypergraph collaborative filtering model to learn the hybrid high-order correlations. These works empirically demonstrate the powerful expressivity and flexibility of hypergraphs. Due to the high-order relations in the historical behaviors (e.g., session, order, etc.), we also utilize the flexible hypergraph to model this complex information. Furthermore, we develop a novel heterogeneous textual hypergraph transformer to generate desirable node representations by fully fusing the fine-grained textual semantics and designed hypergraph topology.

C OPTIMAL TRANSPORT SOLVER

The exact minimization over $\mathbf{T}$ in Formula (5) is computational intractable [1, 38]. Here we employ the Inexact Proximal point method for Optimal Transport (IPOT) algorithm [45] to generate the matrix $\mathbf{T}$ approximately, which iteratively solves the following optimization problem with Algorithm 2:

$$\begin{aligned} \mathbf{T}^{(t+1)} &= \underset{\mathbf{T}\in\mathbf{U}(\mathcal{S},\mathcal{Z}))}{\arg\min} \left\{ \langle \mathbf{T}, \mathbf{C} \rangle + \beta \cdot \mathcal{B}\left(\mathbf{T}, \mathbf{T}^{(t)}\right) \right\} \\ \mathcal{B}\left(\mathbf{T}, \mathbf{T}^{(t)}\right) &= \sum_{i,j} \mathbf{T}_{i,j} \log \frac{\mathbf{T}_{i,j}}{\mathbf{T}^{(t)}_{i,j}} - \sum_{i,j} \mathbf{T}_{i,j} + \sum_{i,j} \mathbf{T}^{(t)}_{i,j}. \end{aligned} \tag{15}$$

Different from the conventional Sinkhorn [13] method, IPOT provides a feasible solution that is closest to the optimal solution set in each proximal iteration until the optimal solution is reached with nested iterative loops, which can address the unstable numerical issue. Furthermore, IPOT can efficiently converge with a single inner iteration, leading to similar computational complexity (near-$O(Lp)$ [1]) to Sinkhorn. Following previous successful applications [9, 10], the stepsize β is set to 0.5, and the number of outer iteration and inner iteration is set to 10 and 1, respectively.

Algorithm 2: OTsolver

Input: token embeddings set $\mathcal{S} = \{\mathbf{s}_i\}_{i=1}^{L}$, reference embeddings set $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^{p}$, cost matrix $\mathbf{C} \in \mathbb{R}^{L\times p}$ and generalized stepsize $1/\beta$.
Output: optimal transport plan $\mathbf{T} \in \mathbb{R}^{L\times p}$.

$\theta \leftarrow \frac{1}{L}\mathbf{1}_L, \mathbf{T}^{(1)} \leftarrow \mathbf{1}_L\mathbf{1}_p^\top$
$\mathbf{A}_{(i,j)} \leftarrow \exp(-\frac{\mathbf{C}_{(i,j)}}{\beta})$
for $t \leftarrow 1, 2, 3, .., T$ **do**
  $\mathbf{Q} \leftarrow \mathbf{A} \cdot \mathbf{T}^{(t)}$
  for $k = 1, ...K$ (Usually set K = 1) **do**
    $\delta \leftarrow \frac{1}{p\mathbf{Q}\theta}, \theta \leftarrow \frac{1}{L\mathbf{Q}^\top\delta}$
  end for
  $\mathbf{T}^{(t+1)} \leftarrow diag(\delta)\mathbf{Q}diag(\theta)$
end for
$\mathbf{T} \leftarrow \mathbf{T}^{(T+1)}$
return $\mathbf{T}$

Table 9: Connected relation between nodes and hyperedges.

Node	Connected hyperedge	Contained nodes
Query	Query Click hyperedge	Queries, Keywords
	Local Session hyperedge	Queries
Keyword	Keyword Click hyperedge	Keywords, Queries
	Local Order hyperedge	Keywords
User	User Click hyperedge	User, Query, Keyword
	Global Search hyperedge	User, Queries
Advertiser	Advertiser Click hyperedge	Advertiser, Keyword, Query
	Global Bid hyperedge	Advertiser, Keywords

Table 10: Hyper-parameter search space.

Hyper-parameter	Search Space	Type
bs	[64, 128, 256, 512]	Choice
β	[0.1, 0.9]	Range
lr	[0.0005, 0.05]	Range
p	[1, 10]	Range
δ	[0.3, 0.7]	Range

D IMPLEMENTATION DETAILS

D.1 Hyperedge Details

Intuitively, the connection relations between different types of nodes and hyperedges in sampled hypergraphs are presented in Table 9. Take the query node as an example. Each query node is linked by two types of hyperedges (i.e. query click hyperedge and local session hyperedge). And the query click hyperedge may connect a various number of query nodes and keyword nodes, while the local session hyperedge only contains multiple query nodes.

D.2 Hyper-parameter Search Space

The hyper-parameters are tuned by the random search [4] for each dataset, including learning rate lr, size of reference set p, shortcut connection weight β, training batch size bs and dropout rate δ. The hyper-parameter search space is shown in Table 10.

E DISTILLATION FOR ONLINE SERVING

Compared to the traditional NLU models, our proposal can incorporate rich behaviors as complementary. However, PASS is more complicated than the doublet-based models due to the designed heterogeneous hypergraph representation learning, which may aggravate the serving latency. Therefore, we adopt a knowledge distillation strategy to learn a lightweight online serving model. Assume PASS is fully trained on the training set, which is viewed as the teacher model. Then, a larger number of (u, q) pairs and (a, k) pairs are randomly sampled from the whole search logs and are used to form up the (u, q, k, a) quadruples. We employ the learned PASS model to generate pseudo-labels for the sampled quadruples, which are viewed as the training signals to learn the student model. The student model is implemented as a BERT model with only three layers of transformers. q and k in input quadruple are fed into the student model to achieve the corresponding embeddings, which are further input into a scoring layer along with pre-learned user and advertiser representations to make final decisions. Note that the student model is trained to fit the soft pseudo-labels instead of the hard labels, namely the probabilities of the quadruples being relevant. The pseudo-labels can be viewed as the intermediates to transport the learned high-order relation enhanced knowledge to the small student model for online serving.

A slightly simplified version of our distilled PASS model has already been successfully deployed in recall stage of Bing Ads and demonstrated significant performance gains. Revenue Per Mile (RPM) and Defect Rate are selected as measurements to estimate the revenue gained for every thousand search requests and the ratio of irrelevant ad impressions, respectively. The defected impressions are labeled by human experts. Compared with the original online severing model (distilled AdsGNN), the online A/B testing results show that PASS significantly increases RPM by 2.11% and reduces advertising defect rates by 2.73%, which demonstrates that our proposal is capable of improving the user experience and driving revenue for the advertisers simultaneously.