Key Laboratory of Big Data and Artificial Intelligence in Transportation, Ministry of Education

The 38th Annual AAAI Conference on Artificial Intelligence

Optimizing Recall in Deep Graph Hashing Framework for Item Retrieval (Student Abstract)

Fangyuan Luo and Jun Wu
School of Computer and Information Technology
Beijing Jiaotong University

1 MOTIVATIONS

- **Hashing-based recommendation (HR) methods** plays a key role to improve item retrieval efficiency, but it is challenged by the **inconsistency between optimization objective and evaluation metric**.
- Existing optimization strategies **incur large information loss**.

2 CONTRIBUTIONS

- **Proposing a smooth recall loss to mitigate the inconsistency.**
- **Proposing an approximation-adjustable gradient estimator to solve the NP-Hard optimization problem.**

3 ALGORITHM

The overall framework of our method is shown in the Figure, where the architecture of user encoder and item encoder are the same as LightGCN.

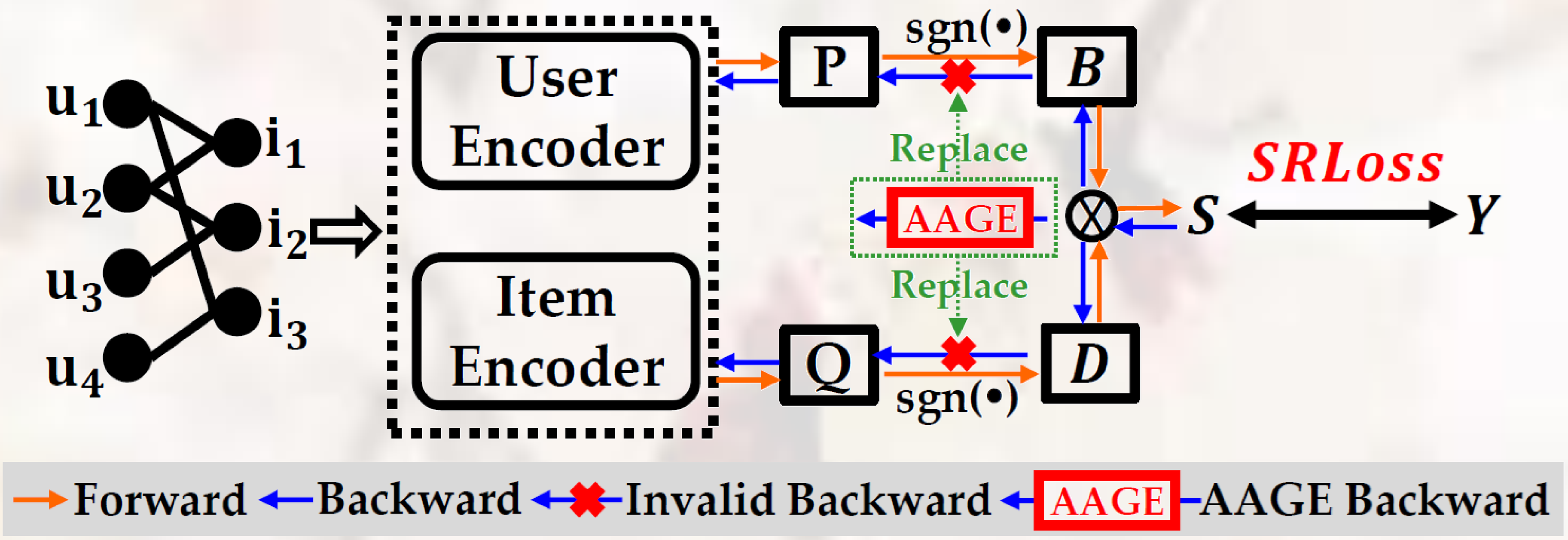

Figure: Illustration of the proposed framework, where P(B) and Q (D) are real-valued embeddings (hash codes) for users and items; S and Y denote the predicted scores and groundtruth; ⊗ denotes the inner product.

- Optimization Objective: targets Recall as the optimization objective

$$\mathcal{L} = -\frac{1}{|\Omega|} \sum_{(u,i)\in\Omega} \mathbb{I}\Big(\big(1 + \sum_{j=1\backslash i}^{n} \mathbb{I}(\mathbf{b}_u^T\mathbf{d}_i > \mathbf{b}_u^T\mathbf{d}_j)\big) \le N\Big)$$

$$s.t.\ \mathbf{B} \in \{-1,1\}^{f\times m}, \mathbf{D} \in \{-1,1\}^{f\times n}$$

Smoothing

$$\mathcal{L} = -\frac{1}{|\Omega|} \sum_{(u,i)\in\Omega} \mathcal{K}_{\phi}\Big(\big(1 + \sum_{j=1\backslash i}^{n} \mathcal{K}_{\psi}(\mathbf{b}_u^T\mathbf{d}_i > \mathbf{b}_u^T\mathbf{d}_j)\big) - N\Big)$$

$$s.t.\ \mathbf{B} \in \{-1,1\}^{f\times m}, \mathbf{D} \in \{-1,1\}^{f\times n}$$

- Optimization Strategy: takes signswish function with parameter β to approximate hash function in the backpropagation

$$\mathrm{sgn}(\mathrm{x}) = \lim_{\beta\to\infty} 2\sigma(\beta \mathrm{x})(1 + \beta \mathrm{x}(1 - \sigma(\beta \mathrm{x}))) - 1$$

$$\frac{\partial \mathrm{sgn}(\mathrm{x})}{\partial x} = \frac{2\cdot[(\beta^2 x + 2\beta)e^{-2\beta x} - (\beta^2 x - 2\beta)e^{-\beta x}]}{(1 + e^{-\beta x})^3}$$

4 EXPERIMENTS

	Gowalla				Yelp2018			
	R@50	R@100	N@50	N@100	R@50	R@100	N@50	N@100
Proposed	**0.23082**	**0.31396**	**0.15109**	**0.17424**	**0.10140**	**0.16348**	**0.06016**	**0.08035**
BGCH	<u>0.19160</u>	<u>0.26590</u>	<u>0.12740</u>	<u>0.14840</u>	<u>0.08350</u>	<u>0.13450</u>	<u>0.05000</u>	<u>0.06700</u>
HashGNN	0.09481	0.15110	0.05112	0.06649	0.04692	0.08250	0.02661	0.03811
HashRec	0.12060	0.18930	0.06160	0.08100	0.06307	0.10995	0.03590	0.05134

Recommendation performance on Gowalla and Yelp2018 dataset, where “R” and “N” denote the Recall and NDCG. The best performing method in each column is boldfaced, and the second best method is each column is underlined.

1. Overall, our proposed method, BGCH and HashRec show superior performance to HashGNN. Such an observation illustrates the effectiveness of providing accurate gradient estimation.
2. Among Proposed, BGCH, and HashRec, our proposed method demonstrates significant improvements. The performance improvements are attributed to the benefits of joint effect of the proposed SR loss and the AAGE.