Efficient Recommender Systems

Supervisor: Prof Erik Cambria
Joenathan Halim (U2340760K)

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Abstract

The rapid growth of e-commerce poses challenges for traditional collaborative filtering, as handling vast numbers of users and items demands large storage and efficient recommendations. Hashing methods for collaborative filtering help by representing data with binary codes, which reduce storage needs and speed up similarity calculations. This poster introduces a deep learning–based collaborative hashing approach that learns user and item representations close to binary codes to minimize quantization loss. The method also handles new users, items, and ratings. Experiments on real-world datasets show the effectiveness of the proposed framework.

Introduction

Recommender system, which focuses on recommending products to customers that may be interested in, have become an integral part of e-commerce, media, and social platforms. Out of a number of recommendation systems, neural collaborative filtering has achieved great success due to its superior performance [1]. Despite the success of collaborative filtering, the fast growth of online shopping platforms brings new challenges to recommender system to scale efficiently to a massive user and products. This poster presents a work that address this problem to improve computational efficiency via deep learning to learn binary codes for collaborative filtering.

Results & Discussion

In experiments on MovieLens, Amazon, and Yelp, DCH's Top-20 Hit Rate and NDCG exceeded or matched state-of-the-art baselines (e.g. 10% NDCG improvement on Amazon and Yelp). Performance improves with longer code lengths up to a point (after 64 bits, gains level off), showing a trade-off between efficiency and accuracy. Crucially, DCH outperformed two-stage hashing and even some full-precision models in ranking metrics. It also maintained robust performance when handling new user/item scenarios, with only slight drops in hit rate for cold-start cases. Overall, the study demonstrated that efficient binary representations can significantly reduce memory and computation (millions of similarity checks in milliseconds) with minimal loss in recommendation quality

	Cases	MovieLens	Amazon	Yelp
EUEI	*DCF*	37.74%	19.29%	30.26%
	CH	20.87%	23.43%	39.14%
	EUEI1	37.94%	49.05%	51.98%
	EUEI2	44.54%	49.19%	59.49%
EUNI	*DCF*	30.00%	2.17%	10.71%
	CH	50.00%	–	–
	EUNI1	50.00%	63.24%	47.40%
	EUNI2	60.00%	64.61%	52.67%
NUEI	*DCF*	14.91%	15.31%	15.38%
	CH	19.77%	–	–
	NUEI1	40.44%	21.63%	34.91%
	NUEI2	42.66%	29.63%	35.50%

Table 1. Results of HIT@20 in the out of samples when the binary code length is 64.

	Cases	MovieLens	Amazon	Yelp
EUEI	*DCF*	10.08%	4.68%	2.83%
	CH	5.43%	5.95%	10.0%
	EUEI1	14.59%	22.01%	23.35%
	EUEI2	18.87%	19.54%	28.98%
EUNI	*DCF*	5.94%	0.5%	2.45%
	CH	11.47%	–	–
	EUNI1	12.02%	28.84%	22.64%
	EUNI2	20.47%	29.93%	20.40%
NUEI	*DCF*	3.60%	3.80%	3.82%
	CH	4.98%	–	–
	NUEI1	14.84%	7.63%	13.10%
	NUEI2	14.78%	10.48%	14.02%

Table 2. Results of NDCG@20 in the out of samples when the binary code length is 64.

Methodology

The Deep Collaborative Hashing (DCH) framework jointly learns compact, binary-friendly user and item representations for efficient recommendation. Starting from a sparse user–item interaction matrix, one-hot encoded IDs are fed into two parallel multilayer perceptrons—one for users and one for items—that map inputs into a shared latent space with dimensionality equal to the desired hash length. The final layers use a binarization-friendly activation (e.g., soft-sign) to push outputs toward ± 1. Training minimizes the loss function written below using SGD:

$$
\begin{aligned}
J_{(\theta_u,\theta_v)} = {} & \sum_{ij:S_{ij}>0} (S_{ij} - \mathbf{b}_i^T \mathbf{d}_j)^2 + \frac{\mu}{2}\|\mathbf{B} - (\mathbf{P} + \frac{1}{\mu}\mathbf{E})\|_F^2 \\
& + \frac{\mu}{2}\|\mathbf{D} - (\mathbf{Q} + \frac{1}{\mu}\mathbf{F})\|_F^2 + \lambda_1\|\theta_u\|_F^2 + \lambda_2\|\theta_v\|_F^2 \\
s.t. \quad & \mathbf{b}_i = softsgn(MLP_u(\mathcal{N}_{ui})), \mathbf{d}_j = softsgn(MLP_v(\mathcal{N}_{vj}))
\end{aligned}
$$

Equation 1. Loss Function

Once trained, embeddings are converted into fixed-length binary codes via the sign function, enabling compact storage and fast retrieval through Hamming distance. DCH also supports out-of-sample generation of codes for new users or items using the trained towers, avoiding costly full retraining. This integrated design ensures that DCH produces binary codes that are both efficient to store and compute with, and directly optimized for recommendation performance [2].

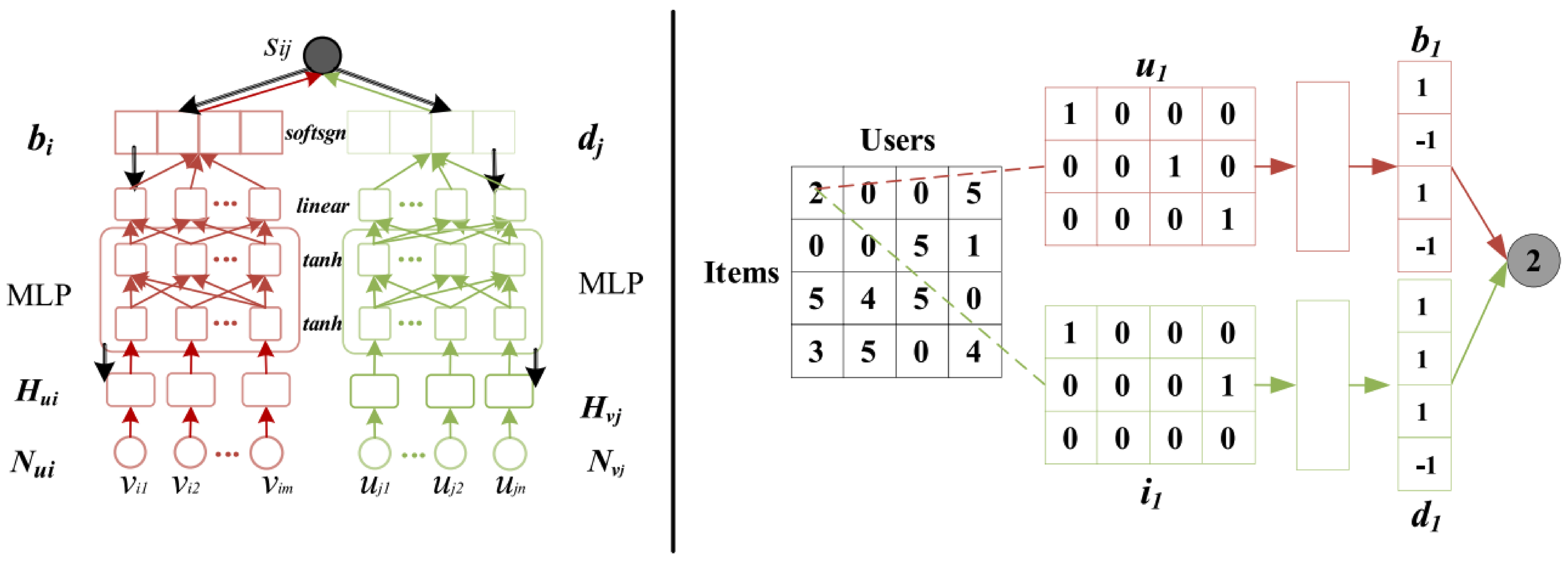

Figure 1. DCH Framework

Conclusion

Efficient Recommendation Systems (DCH) demonstrate that we can push the boundaries of scale – making personalization feasible even with web-scale data – through clever use of neural networks and hashing to compress information. Going forward, recommender system research is likely to fuse these directions: scalable architectures that can handle multi-objective optimization (accuracy, diversity, fairness, transparency simultaneously).

References

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

[2] Yang Li, Suhang Wang, Quan Pan, Haiyun Peng, Tao Yang, and Erik Cambria. Learning binary codes with neural collaborative filtering for efficient recommendation systems. *Knowledge-Based Systems*, 172:64–75, 2019.