Proceedings of the 4th International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2026.38758

Sequence Recommendation Algorithm Based on Contrastive Learning

Farun Zhan*, Yang Qian, Yanji Mou

Stony Brook Institute at Anhui University, Hefei, China
**Corresponding Author. Email: 3052536144@qq.com*

Abstract. Weibo become a important social and service platform in people's daily life, as social network grow fast and popular, it create lots of valuable informations. Users can through this platform efficient get many dimensions news about social hot topics and industry trends, this can good satisfy personal information consumption needs. As a communication and sharing medium in internet resources, Weibo's obvious and hidden value informations is deeply affect modern life now, and it integration into people's daily routine make information get more easier than before, also the platform let user share and exchange ideas quickly, this create a new way for people connect with world and stay inform about what happening around them.

Keywords: Weibo recommendation, Contrastive learning, Graph collaborative filtering, Dynamic routing, Multi-interest recall

1. Introduction

This paper focus on two big problems in current Weibo recommendation algorithms, user behavior data is often sparse and have noise, and using single vector to show user interest cannot catch complex relations between interests. We propose two models to solve this, one is graph collaborative filtering with similarity contrastive learning, another is MIND Weibo recommendation based on dual attention mechanism. Through experiment evaluation and performance analysis, we verify that these two models is effective and can improve recommendation result, the result show our method work well in real application scenario and solve the data sparse problem better than before [1, 2].

2. Graph neural collaborative filtering

For user-item pairs that have interactions (u,i) , we define the message pass from item i to user u as below:

$$m_{u\leftarrow i}=f(e_i,e_u,p_{ui}) \tag{1}$$

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.38758

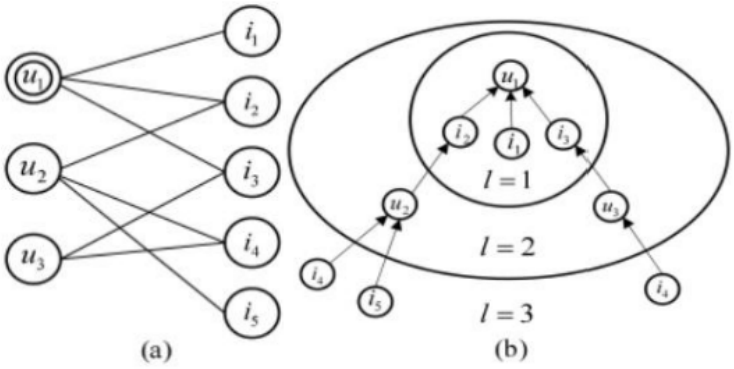


Figure 1. Interaction schematic

Aggregate messages propagated from u's neighborhood to improve u's embedding representation. Define the aggregation function as:

$$e_u^{(1)} = \text{LeakyReLU}(m_{u\leftarrow u} + \sum_{i\in N_u} m_{i\leftarrow u}) \tag{2}$$

where, the propagated message is defined as follows:

$$m_{u\leftarrow u}^{(l)} = W_1^{(l)} e_u^{(l-1)} \tag{3}$$

Figure 1 show user-item interaction graph and high-order connectivity. Figure (a) is the user-item interaction graph, and Figure 1(b) display a tree structure expand from target user u_1 who need recommendation. High-order connectivity means the path from nodes with path length bigger than 1 to user u_1, this kind of high-order connection contain rich collaborative signals that can help improve recommendation accuracy and capture complex user preference patterns in the graph structure.These high-order connections can good find user hidden interests, it gather information from multi-hop neighbors and solve data sparse problem [3]. The tree structure's layered show help tell which path is more important, this lay foundation for graph neural network's message passing, and make the model learn better from complex connection patterns in the graph, also improve recommendation performance by use these rich connectivity information effectively.

3. Dynamic routing

Dynamic routing mechanism can achieve message passing between high-level capsules and low-level capsules, high-level capsules dynamically adjust connection strength based on activation states of received low-level capsules [4]. So it can effectively select low-level features that most relevant to current task, this selective information transmission help network focus on important features and ignore useless ones, which make the model work better in complex recommendation scenarios where user interests is diverse and need careful feature selection.As shown in Figure 2.

DOI: 10.54254/2755-2721/2026.38758

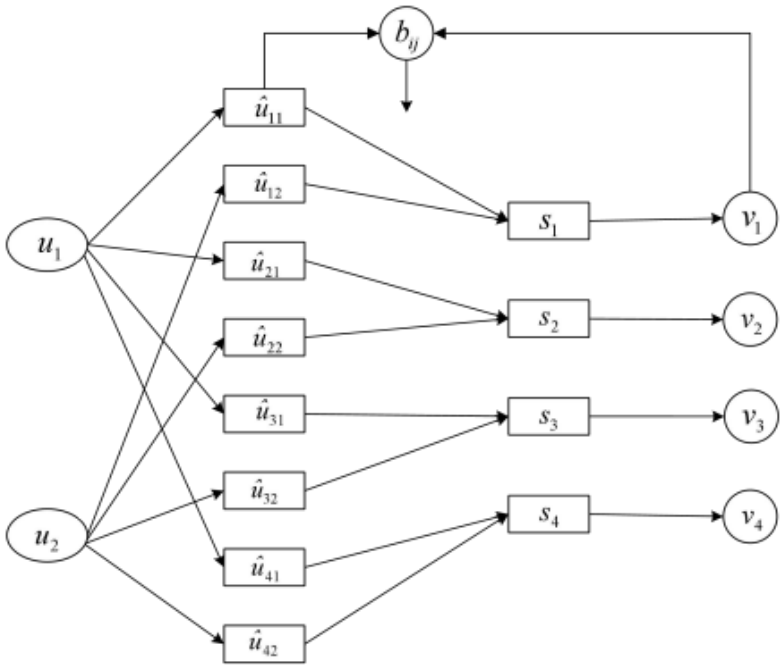


Figure 2. Filter model styles

Suppose there are two capsule layers, we call first layer low-level capsules and second layer high-level capsules. Dynamic routing aims to calculate high-level capsule values iteratively based on given low-level capsule values [5]. In each iteration, given low-level capsules, the information from low-level capsules is aggregated into candidate vectors of high-level capsules, the formula is shown below:

$$s_i = \Sigma_i c_{ij} u_{ji} \tag{4}$$

Where $u_{j|i}$ is the product of previous layer capsule and weight matrix W_{ij}.c_{ij} is coupling coefficient,its value is adjusted through dynamic routing iteration process.The calculation formulas for $u_{j|i}$ and coupling coefficient cij is shown below respectively,and the softmax function is used to normalize the coupling coefficients so that they sum to one and canproperly represent the agreement between low-level and high-level capsulesduring the routing process, Thus derived:

$$u_{ji} = c_{ij} u_i \tag{5}$$

$$c_{ij} = \frac{\exp b_{ij}}{\Sigma exp b_{ik}} \tag{6}$$

where, b_{ij} represent routing between low-level capsule u_i and high-level capsule v_j, it need initialization when use,this value is updated during iteration process and finally determine how much information pass between capsules.

Finally, update all b_{ij} through the following formula:

DOI: 10.54254/2755-2721/2026.38758

$$b_{ij}=b_{ij}+u_{j|i}^{T}v_i \tag{7}$$

4. Attention mechanism

Compute attention scores. In attention mechanism, core calculation include three parts: Query, Key, Value, all of them are get from input data through linear transformation. Methods to measure similarity between Query and Key include vector dot product, cosine similarity and multi-layer perceptron, the calculation formulas of these three methods is shown below, and different methods suit different scenarios, dot product is simple and fast, cosine similarity focus on direction, while multi-layer perceptron can learn complex matching patterns, Thus derived:

$$\text{Similarity}(\text{Query},\text{Key}_i)=\text{Query}.\text{Key}_i \tag{8}$$

$$\text{Similarity}(\text{Query},\text{Key}_i)=\frac{\text{Query},\text{Key}_i}{||\text{Query}||,||\text{Key}_i||} \tag{9}$$

$$\text{Similarity}(\text{Query},\text{Key}_i)=\text{MLP}(\text{Query},\text{Key}_i) \tag{10}$$

weighted sum:

$$Attention = \Sigma_{i=1}^{l_x} a_i \text{Vaule}_i \tag{10}$$

Apply attention scores to Value, get final output through weighted sum. The weighted sum formula is shown below, this process combine different values based on their importance weights, so the output can focus on most relevant information and ignore less important parts, which make the model's representation more accurate and useful for downstream tasks like recommendation, Thus derived:

$$\text{Output}=\text{Attention}.\text{V} \tag{12}$$

5. Evaluation metrics for recommendation algorithms

Confusion matrix is a tool for evaluate classification model performance, it show model's prediction results and actual labels' relationship in matrix form. The matrix contain four main indicators: true positive (TP), false positive (FP), true negative (TN) and false negative (FN). These indicators can calculate precision, recall and other evaluation metrics [6, 7]. So we can deeply analyze the model's effectiveness and reliability, this help us understand where the model make mistakes and how to improve it.As shown in Table 1:

DOI: 10.54254/2755-2721/2026.38758

Table 1. efficiency analysis

confusion matrix		true value	
		1	0
predicted value	1	TP	FP
	0	FN	TN

Recall is used to measure recommendation system's coverage ability, specifically, recall represent the proportion of successfully recommended items among all items that users actually like, this metric show how good the system can find relevant items from large candidate pool and recommend them to users.As shown in formula (11):

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{EN}} \tag{13}$$

Normalized Discounted Cumulative Gain evaluate ranking accuracy by computing the gap between model's output ranking list and real user interaction list. The calculation formula is shown below:

$$\frac{1}{\text{N}}\sum_{\text{i}=1}^{\text{N}}\frac{1}{\log_2(\text{p}_\text{i}+1)} \tag{14}$$

6. Conclusion

This paper explore personalized Weibo recommendation algorithms, though two models get good results, there still have improvement space. Future work can improve from three aspects: combine likes, comments, follows and other behavior data; explore more methods to find similar users and posts based on SCLGCF; mix MINDDouAtt with similarity contrastive learning, this can reduce data sparsity and noise, mine user interests from many sides, and make recommendations more diverse and fresh, so users will have better experience when using the recommendation system.

References

[1] Niu, J., Zhou, J., & Dai, Y. (2026). Financial risk prediction and control optimization of listed seed companies based on machine learning algorithms: An empirical analysis using time-series data. *Proceedings of Business and Economic Studies, 9*(2), 103–117. https://doi.org/10.26689/PBES.V9I2.13640

[2] Chi, M., Pan, S., Feng, J., et al. (2026). A two-stage algorithm for time series compression: ARIMA-based pre-compression and reinforcement learning optimized chunking. *Mathematics, 14*(5), 841. https://doi.org/10.3390/MATH14050841

[3] Zhang, Z., Liu, H., Shi, T., et al. (2026). Measuring and forecasting global digital economy efficiency: An integrated approach using the super-efficiency sequential SBM model and machine learning algorithms. *Information Processing and Management, 63*(4), 104638. https://doi.org/10.1016/J.IPM.2026.104638

[4] Malek, G. N., & Peng, Q. (2025). Correction: Reinforcement learning for self-adaptive genetic algorithm in assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 141*(9–10), 1. https://doi.org/10.1007/S00170-025-17034-0

[5] Amoli, G. M., Hasanlou, M., Samadzadegan, F., et al. (2025). Estimating soil organic carbon using time series Band 11 (SWIR) of multispectral Sentinel-2 satellite images and machine learning algorithms. *Remote Sensing*

DOI: 10.54254/2755-2721/2026.38758

Applications: Society and Environment, 40, 101736–101736. https://doi.org/10.1016/J.RSASE.2025.101736

[6] Tabar, E. M., & Sisman, Y. (2025). Deep learning-based modeling and prediction of GNSS time series: A comparative analysis of adaptive optimization algorithms. *Advances in Space Research, 76*(4), 2086–2103. https://doi.org/10.1016/J.ASR.2025.06.018

[7] Jovic, A., Frid, N., Brkic, K., et al. (2025). Interpretability and accuracy of machine learning algorithms for biomedical time series analysis – a scoping review. *Biomedical Signal Processing and Control, 110*(PB), 108153–108153. https://doi.org/10.1016/J.BSPC.2025.108153