Enhancing Social Recommendation with Multi-View BERT Network

Tushar Prakash*
Sony Research India, India
tushar121prakash@gmail.com

Raksha Jalan*
Sony Research India, India
rakshajalan9@gmail.com

Naoyuki Onoe
Sony, Japan
Naoyuki.Onoe@sony.com

***Abstract*—With the emergence of online social platforms enabling users to share their opinions with others, there has been a critical need for developing a recommendation system incorporating users' social connections to learn their preferences. Social relations between users offer potential information about users' preferences, which alleviates the data sparsity issue and boosts recommendation performance. Several efforts have been made to develop an efficient social recommendation system. Nevertheless, existing approaches have two significant drawbacks: First, they haven't efficiently explored the complex correlations between the diverse influence of neighbours on users' item preferences. Second, contemporary systems rely on the unidirectional context of items and fail to embrace bidirectional contexts and long range dependencies while predicting the next user-item interaction. To address the above issues, we propose a novel framework for Social Recommendation called Multi-View Bert Network (MVBN). This model incorporated Bidirectional Encoder Representations from Transformer(BERT) with Multi-Task learning to learn bidirectional context-aware user and item embeddings with neighbourhood sampling. Neighbourhood Sampling technique samples the most influential neighbours for all the items, and our proposed Sequence Header correlates sampled neighbours with users. The main objective of our model is to predict the next item that a user would interact with based on its interaction behaviour. Experiments on three real-world datasets demonstrate that the proposed MVBN model outperforms the state-of-the-art recommendation methods consistently and significantly. Finally, an extensive ablation study is carried out to validate the importance of each component in the system.**

***Index Terms*—Recommendation Systems, Social Recommendations, Transformers, BERT, Multi-Task learning**

I. Introduction

With the rapid growth of E-commerce and online entertainment platforms, users expect personalised items and information to be delivered for a better experience. Success of these platforms rely heavily on recommendation systems that are designed to filter irrelevant information and offer personalised recommendations. As social media has grown in popularity, numerous e-commerce websites and review platforms have evolved into popular social platforms to enhance users' engagement. For instance, Disney+ offers "GroupWatch" and Amazon Prime Video has "Watch Party", both of which enable people to watch synchronized content together and share their reactions. With the rising popularity of such platforms, building recommendation systems incorporating social relationships into the recommendation model is essential.

* Both authors contributed equally

These systems offer significant potential to alleviate data sparsity and enhance recommendation performance. Social relations among users can provide connections' interests for more accurate modelling of user preferences and introduce new opportunities to recommend items to the target users. However, it is difficult to extract the most relevant information for modelling user preferences due to the complexity of the high-order social relations and similarity linkages.

Existing Social Recommendation methods made numerous attempts to incorporate social relations into the recommendation generation. Earlier Matrix-factorization have been utilized to build social recommendation models [8], [10] by incorporating social relations into existing models. In the last few years, various deep learning-based recommendation models like NGCF [16] as well as social recommendation models such as [6], [18] have been proposed. Recently, there has been rising enthusiasm for using Transformer [14] and BERT [4] in the recommendation systems, models like Bert4Rec [13] and SSE-PT [19] are a couple of famous transformer-based models in the recommendation system domain.

Existing social recommendation methods have shown promise, but they have two major limitations. First, they assume that users with social connections share common interests, without effectively considering the complex influences of a user's neighbors on their preferences. Second, these methods don't capture comprehensive context-aware representations of user preferences and long-term dependencies because they only focus on one-sided interaction contexts.

Therefore, to tackle the above problems in social recommendations, we propose a novel framework called Multi-View BERT Network (MVBN). MVBN consists of an embedding layer using neighborhood sampling and a sequence header, both serving as input to Bidirectional Encoder Representations from Transformer (BERT) with Multi-Task learning. To address the first issue, the model employs neighborhood sampling to identify influential neighbors for a given user and introduces a concept called the "sequence header" to establish correlations between users and interacted items, enhancing representation learning.To tackle the second issue, BERT is used to capture bidirectional interaction contexts and learn user preferences. Additionally, two different views, user-centered and item-centered, are generated from the user-item interaction history to enhance the model's performance.

We utilize these multi-view information using Multi-Task learning, which not only enables the model to learn better user representations for the interacted items, but also allows the model to learn item representations for interacted users along with their social network.Here, incorporating bidirectional contexts and social relations in multi-tasking allows our model to learn the different preferences of users with similar item interaction histories.

To the best of our knowledge, we are the first to propose a solution utilizing BERT in conjunction with multi-task learning for a social recommendation system.

The key contributions of this work are summarized as follows:

- Introduced a novel approach to learn user/item representation by performing neighborhood sampling to identify influential neighbors and using sequence headers to correlate the sampled neighbor with the user space.
- Proposed a BERT-based framework to predict user-item interactions by considering bidirectional contexts and long-term dependencies.
- To enhance model's performance, we employed Multi-task learning to generate two views: user-centered view and item-centered view, based on user-item interactions and social network.

II. Related Work

Earlier, various Matrix Factorization(MF) based approaches like SocialMF [10], TrustSVD [8] and [27] were developed. Later, multiple deep learning techniques have been proposed for recommendation systems [9]. Additionally, models like LightGCN [29], NGCF [16] utilizes GNNs and Graph Convolution Networks(GCNs). In the domain of social recommendation, methods like DeepSoR [17] and [5], [15] incorporate social relationships into deep learning based recommendation generation. Models like Diffnet [20], DiffNet++ [18], and DICER [6] have demonstrated effectiveness of graphs and attention based networks for better capturing the user-item interaction and social relations. To capture users' dynamic interests, various sequential recommendation algorithms [7] have emerged, which extract features from social relations and user behaviour sequences. To further improve sequential modelling, various self-attention based models like Transformer [14] and BERT [4] have been utilised to build recommendation systems. Bert4rec [13], SASRec [28] are a couple of well-known recommendation systems that use transformers. Prior literature on recommendation systems has explored sequential behavior and self-attention mechanisms, yet the domain of social recommendations has not harnessed self-attention and transformers to capture users' high-order relationships.

Existing studies suffer from two common limitations: i) Most methods, apart from DICER [6], only consider the immediate local neighbors' information, neglecting potentially valuable insights from distant connections. ii) User interest modeling often treats information from friends uniformly, ignoring the context of the recommendation, which may involve long-range dependencies. Our proposed model adopts a unique method, sending sequence header and embeddings to BERT instead of the direct user-item interaction sequence. This approach extracts relevant information from the interaction history and models high-order relations between items and users.

In order to further enhance the social recommendation model's performance, various models have been incorporating multi-task learning techniques. Recent work like SoNeuMF [25] and Trust-aware Multi-task Knowledge Graph (TMKG) [26] leveraged multi-tasking to improve recommendation's performance. Motivated by the proven effectiveness of multi-tasking in above recommendation models, we present a novel way of utilizing multi-tasking framework for social recommendation.

III. Methodology

In this section, we first briefly outline the research problem, followed by a detailed discussion of the architecture of the proposed model MVBN and its components. Finally, we discuss the training procedure of the model.

A. Problem Statement

In Social Recommendation(SR) problem, we have a user-item interaction matrix R of users $U = \{u_1, u_2, u_3, ...u_M\}$ and items $V = \{v_1, v_2, v_3, ...v_N\}$, along with a user-user social network N_{uu}. Here, $R \in \{0, 1\}$ based on the interaction of users U with items V and N_{uu} represents the social links $L = \{l_1, l_2, ...l_q\}$, where $l_p = \{u_i, u_j\}$, implying that user u_i trusts user u_j.

The objective of SR is to predict the next item v_{ij} that a user u_i will interact with, given u_i's interaction history and it's corresponding social network $N_{u_i u}$. Items in the interaction history are ordered based on user-item interaction timestamps. Therefore, we have viewed this problem as a next-item prediction task and denoted as $\mathcal{T}_1$. To support $\mathcal{T}_1$, a auxiliary task $\mathcal{T}_2$ of next-user prediction is also used.

B. Embedding Layer

Several studies have shown a positive correlation between users' social behavior and their interactions with items [1], [2]. To capture this correlation effectively, we propose enriching user and item embeddings with Neighbourhood Sampling ($\mathcal{NS}$). This method allows us to sample the most influential neighbors, as considering all neighbors can introduce noise and degrade recommendation performance [23].

In the task $\mathcal{T}_1$, we have a item input-sequence as $\mathcal{I}_v(u_i) = \{v_1, v_2, v_3, ...v_m\}$ for user u_i and we randomly mask some items to generate $\mathcal{I}_v(u_i)^{(m)} = \{v_1, [mask], v_3, ...v_m\}$ using Cloze task [**?**]. Now, for each item v_i in $\mathcal{I}_v(u_i)^{(m)}$, we define an intermediate embedding, $\mathbb{E}_I^{v_i}$ as the concatenation of the item-user history $\mathcal{H}_{v_i u}$ and item-item similarity $\mathcal{S}_{v_i v}$. Here, $\mathcal{H}_{v_i u}$ consist of all user that have interacted with v_i. While, $\mathcal{S}_{v_i v}$ contain similar items to v_i by considering 50% common users. Next, the neighbourhood sampling has been performed on $\mathbb{E}_I^{v_i}$ using multinomial distribution as described in [11], which generates the embedding $\mathbb{E}_\alpha^{v_i}$. In this way, we learn v_i's correlation with the user and item's latent spaces by prioritising the most influential neighbours.

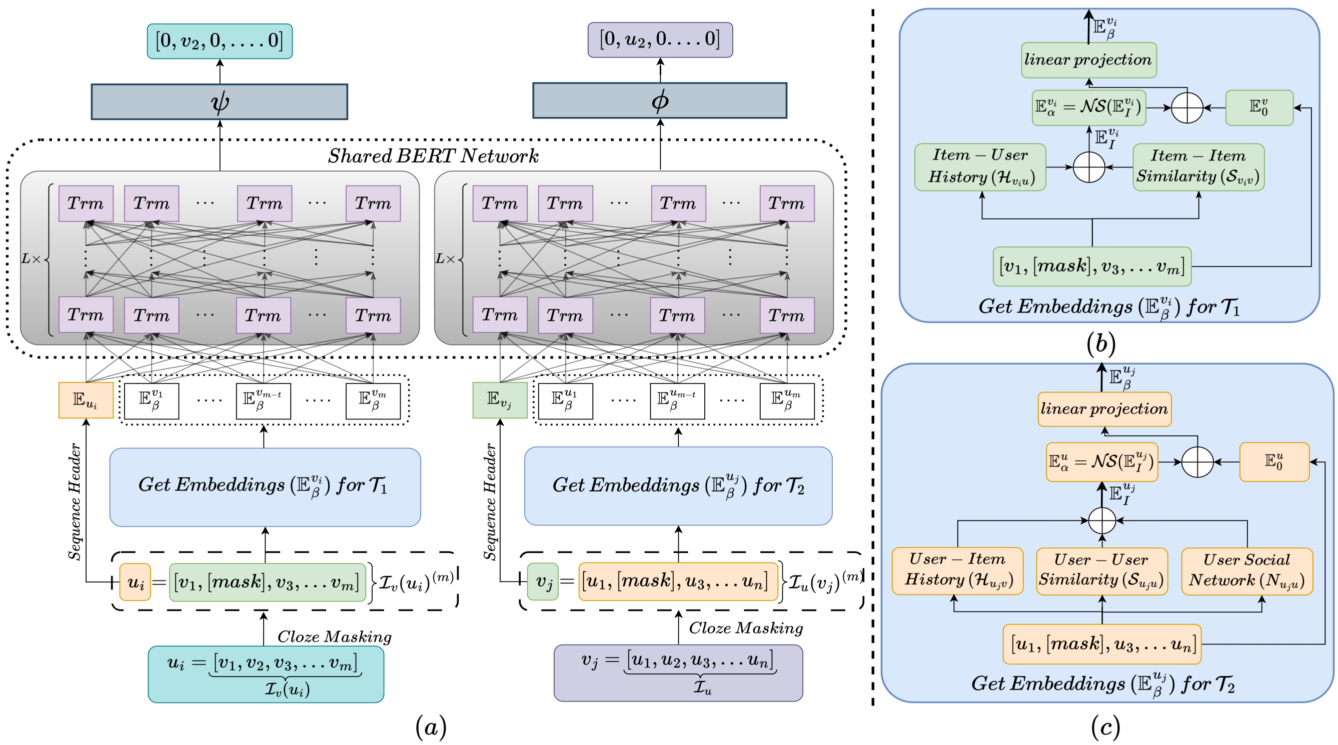

Fig. 1: (a) Illustrates the The Multi-View BERT Network MVBN architecture. (b) Module to generate embedding $\mathbb{E}_\beta^{v_i}$ for $\mathcal{T}_1$. (c) Module to generate embedding $\mathbb{E}_\beta^{u_j}$ for $\mathcal{T}_2$

Finally, To place the similar user-item pairs closely in vector space, we proposed a sequence header (u_i) to generate embedding $\mathbb{E}_{u_i}$. It improves correlation learning between user and item sequences within BERT. Then, we concatenate the $\mathbb{E}_\alpha^{v_i}$, $\mathbb{E}_0^v$ and $\mathbb{E}_{u_i}$ to generate the final embedding, $\mathbb{E}_\beta^{v_i}$ as shown below:

$$\mathbb{E}_\beta^{v_i} = \mathbb{E}_{u_i} \oplus \sigma(W^T(\mathbb{E}_0^v \oplus \mathbb{E}_\alpha^{v_i})) \tag{1}$$

Where, $\mathbb{E}_0^v$ is the embedding of $\mathcal{I}_v(u_i)^{(m)}$ and W are the weights for the linear projection followed by the sigmoid activation function. This whole process of embedding generation for $\mathcal{T}_1$ is presented in figure 1(b) and mentioned in function $get\_embeddings()$ of Algorithm 1.

Similarly, for the task $\mathcal{T}_2$, final embedding of user u_j is shown below:

$$\mathbb{E}_\beta^{u_j} = \mathbb{E}_{v_j} \oplus \sigma(W^T(\mathbb{E}_0^u \oplus \mathbb{E}_\alpha^{u_j})) \tag{2}$$

Note that for $\mathcal{T}_2$, In addition to user-item history $(\mathcal{H}_{u_j v})$ and user-user similarity $(\mathcal{S}_{u_j u})$, user's social network $(N_{u_j u})$ is also used.Here, $N_{u_j u}$ allows us to capture the hidden correlation between the user's preferences and its social network. Since $\mathcal{S}_{u_j u}$ only considers similarity in user's preferences based on common purchase history and does not utilise its explicit social connection.

C. Shared BERT Network

BERT [4] is an effective sequential model that employs a multi-layer bidirectional Transformer encoder as its foundation. Our Multi-View BERT Network leverages the BERT's multiple bidirectional Transformer layer and self-attention mechanism.

To learn a deep bidirectional representation, the final embeddings $\mathbb{E}_\beta^{v_i}$ and $\mathbb{E}_\beta^{u_j}$ are passed to BERT, which is shared between both tasks. The shared BERT network contains Transformer layers (referred to as Trm in figure 1), which further consist of a Multi-Head Self-Attention (MH) layer and a Position-wise Feed-Forward Network. MH allows to capture the long-range dependencies between representation pairs in the sequences.The detailed architecture of the Transformer can be referred from [14].

D. Model Training with Multi-Task Learning

Multi-task learning can be leveraged when we have several interrelated tasks with shared embedding. To improve the performance of our model, by availing the benefits of social links and item's view, we have employed multi-task learning by designing an auxiliary task $\mathcal{T}_2$, to learn the rich user embeddings.

Let the θ be the trainable function of the BERT network and ψ, ϕ are the classification layers for the $\mathcal{T}1$ and the $\mathcal{T}2$ respectively. For each iteration, final embeddings $\mathbb{E}_\beta^{v_i}$ and $\mathbb{E}_\beta^{u_j}$ are generated as described in section III-B and fed into BERT to learn the final representations. These representations are then passed through classification layers ψ and ϕ to predict the masked values of the input sequence as mention in line 10, 11 of Algorithm 1 for $\mathcal{T}_1$ and $\mathcal{T}_2$ respectively.

The objective function $\mathcal{L}$ for the training of our MVBN is described below:

$$\mathcal{L}(\mathbb{E}_{\beta}^{v_i}, \mathbb{E}_{\beta}^{u_j}; \theta, \psi, \phi) = \mathcal{L}_{\mathcal{T}_1}(\psi(\theta(\mathbb{E}_{\beta}^{v_i})), v_o) + \alpha(\mathcal{L}_{\mathcal{T}_2}(\phi(\theta(\mathbb{E}_{\beta}^{u_j})), u_o) \tag{3}$$

Where, $\mathcal{L}_{\mathcal{T}_1}$ and $\mathcal{L}_{\mathcal{T}_2}$ are the cross entropy loss functions for $\mathcal{T}_1$ and $\mathcal{T}_2$ respectively. v_o and u_o are the masked value, and α is the weight given to auxiliary task($\mathcal{T}_2$).

$$\underset{(\theta,\psi,\phi)}{\arg\min}[\mathcal{L}(\mathbb{E}_{\beta}^{v_i}, \mathbb{E}_{\beta}^{u_j}; \theta, \psi, \phi)] \tag{4}$$

The Final objective function of MVBN is to minimise the loss $\mathcal{L}$ given the model parameters as mentioned in equation 4. The training process with multi-task learning is presented in figure 1(a) and mentioned in Algorithm 1.

Algorithm 1 Algorithm for MVBN

```
for each user i, item j in N Users and M Items respectively do
    I_v(u_i)                       ▷ given item input-sequence for user u_i
    I_u(v_j)                       ▷ given user input-sequence for item v_j
    I_v(u_i)^(m) = Cloze-Masking(I_v(u_i))
    I_u(v_j)^(m) = Cloze-Masking(I_u(v_j))
    H_{v_i u}, S_{v_i v} ⇐ I_v(u_i)^(m)
    H_{u_j v}, S_{u_j u}, N_{u_j u} ⇐ I_u(v_j)^(m)
    E_β^{v_i} = get_embeddings(H_{v_i u}, S_{v_i v}, None)
    E_β^{u_j} = get_embeddings(H_{u_j v}, S_{u_j u}, N_{u_j u})
    v̂ = ψ(θ(E_β^{v_i})), û = φ(θ(E_β^{u_j}))
    L_{T_1} = L_ce(v_o, v̂), L_{T_2} = L_ce(u_o, û)
    L = L_{T_1} + α * L_{T_2}
end for
get_embeddings(H, S, N_uu)
    if N_uu is None then
            E_I = H ⊕ S
    else
        E_I = H ⊕ S ⊕ N_uu, E_α = NS(E_I)
        E_β = E_u ⊕ σ(W^T(E_0 ⊕ E_α))
    end if
end get_embeddings(H, S, N_uu)
```

IV. Experiments and Evaluations

This section aims to answer following research questions:

- RQ1: How does MVBN perform compared to baselines?
- RQ2: How do different components and Hyper-parameter settings impact the MVBN framework's performance?

A. Experimental Settings

1) Datasets: We have evaluated our proposed framework on three real-world Social Recommendation datasets, Epinions, Ciao and Yelp.

The statistical details of these datasets are summarized in Table I. For users without social links, we randomly sampled 12 users from the top 50 most trusted users across the dataset.

TABLE I: Dataset details

Datasets	**# users**	**# items**	**# social links**
Epinions	22166	296277	398751
Ciao	7375	105114	115632
Yelp	17235	37378	155731

2) Baselines: To demonstrate the effectiveness of our proposed model, we compare MVBN with three strong baselines from recommendations and seven baselines from social recommendations. NGCF [16] integrated neural architectures with matrix factorization technique. SASRec [28] and BERT4Rec [13] captures users' sequential behaviours using transformers and performs state-of-the-art sequential recommendation. In social recommendation, SBPR [27] is popular Bayesian personalised ranking-based recommendation model. Social MF [10], TrustSVD [8] are wellknown trust-based matrix factorization techniques. DiffNet [20] , DiffNet++ [18] utilizes GNNs and neural influence based diffusion network for the social recommendation. ConsisRec [22] utilizes GNN-based architecture to resolve inconsistent neighbours and learn consistent node embeddings for rating prediction. As our primary objective is to anticipate the interactions, We transform the detailed ratings into a value of 1 or 0, indicating whether or not the user rated the item. DICER [6] utilizes a relation-aware GNNs to exploit multi-relationship and high-order neighbour information.

3) Evaluation Metrics: For the evaluation of our model, we have used Hit Ratio (HR) [3] and Normalized Discounted Cumulative Gain (NDCG) [3]. These are widely used top-N ranking evaluation matrics for recommendation systems. HR specifically measures the percentage of hit items in the top-N list, whereas NDCG focuses on the highest-ranked items. To evaluate the performance, we randomly sampled 100 negative items for each user as we focused on the top-N ranking performance with a large itemset, similar to many other works [20], [9].

4) Experiment Details: Each dataset is randomly split into 80% training and 10% validation, and 10% testing. For hyperparameters tuning, we have applied grid search. We experimented with the number of transformer heads in {2,4,6,8} and looked for the best neighbour percentage in {0.2,0.4,0.6,0.8,1.0} for neighbourhood sampling.We searched for α, weight assigned to auxiliary task in {0.2, 0.3, 0.5, 0.8} and to construct the similarity matrices, we find optimal value of η is 0.1. Lastly, we search for embedding sizes in {32,64,128,256}.To address the over-fitting issue, all experiments used early stopping based on NDCG@10. The proposed method is implemented in Pytorch and trained on Geforce RTX 2070 GPU.

B. Performance Evaluation

We can clearly observe that our proposed model MVBN performed significantly better on all datasets as compared to baselines. From Table II, we have the following findings: (1) We have also compared model's performance with few popular sequential recommendation baselines like NGCF,BERT4Rec

TABLE II: Performance evaluation of proposed MVBN as compared to existing recommendation methods on Epinion and Ciao. Improvement has been shown against the best-performing SOTA methods in general recommendation i.e BERT4Rec and Social recommendation model i.e., DICER (underlined). Due to space constraint we have listed results of yelp on **https://github.com/theTushar-dot/MVBN_results**. Improvements are statistically significant with $p < 0.05$.

Epinion	**HR@5**	**HR@10**	**HR@15**	**NDCG@5**	**NDCG@10**	**NDCG@15**
NGCF [16]	0.2826	0.3571	0.3987	0.2519	0.2744	0.2931
SASRec [28]	0.3988	0.5089	0.5814	0.3431	0.3867	0.4151
BERT4Rec [13]	<u>0.4359</u>	<u>0.5206</u>	<u>0.5927</u>	<u>0.3845</u>	<u>0.4153</u>	<u>0.4294</u>
SBPR [27]	0.2611	0.3352	0.3721	0.2310	0.2553	0.2745
SocialMF [10]	0.2656	0.3402	0.3797	0.2379	0.2601	0.2816
TrustSVD [8]	0.2718	0.3428	0.3872	0.2403	0.2609	0.2827
DiffNet [21]	0.3041	0.3748	0.4222	0.2716	0.2976	0.3130
DiffNet++ [18]	0.3202	0.4049	0.4521	0.2770	0.3064	0.3221
ConsisRec [22]	0.3775	0.4658	0.5299	0.3545	0.3889	0.4114
DICER [6]	<u>0.4051</u>	<u>0.5111</u>	<u>0.5872</u>	<u>0.3541</u>	<u>0.3935</u>	<u>0.4182</u>
MVBN	**0.5908**	**0.6567**	**0.6912**	**0.5507**	**0.5735**	**0.5841**
Improvement	45.84%	28.48%	17.71%	55.52%	45.74%	39.67%

Ciao	**HR@5**	**HR@10**	**HR@15**	**NDCG@5**	**NDCG@10**	**NDGC@15**
NGCF [16]	0.2359	0.2844	0.3160	0.2137	0.2297	0.2378
SASRec [28]	0.3101	0.3896	0.4605	0.2816	0.3148	0.3463
BERT4Rec [13]	<u>0.3475</u>	<u>0.4213</u>	<u>0.4703</u>	<u>0.3223</u>	<u>0.3397</u>	<u>0.3584</u>
SBPR [27]	0.2018	0.2523	0.2883	0.1825	0.1993	0.2139
SocialMF [10]	0.2099	0.2609	0.2995	0.1946	0.2094	0.2202
TrustSVD [8]	0.2194	0.2719	0.3172	0.2063	0.2197	0.2289
DiffNet [21]	0.2379	0.2860	0.3209	0.2198	0.2353	0.2469
DiffNet++ [18]	0.2471	0.2996	0.3403	0.2320	0.2494	0.2628
ConsisRec [22]	0.2761	0.3656	0.4331	0.2499	0.2837	0.3066
DICER [6]	<u>0.3221</u>	<u>0.4002</u>	<u>0.4678</u>	<u>0.3003</u>	<u>0.3278</u>	<u>0.3505</u>
MVBN	**0.4502**	**0.5206**	**0.5680**	**0.4298**	**0.4534**	**0.4687**
Improvement	39.77%	30.08%	21.41%	43.12%	38.31%	33.72%

and SASRec recommendation systems. Since these model's don't originally incorporate social information, to make fair comparison of models along with social information, we have replaced user embeddings with concatenation of user embeddings and its friend's embeddings. (2) Compared with the existing deep learning-based Social Recommendation models, the impressive improvements of MVBN prove the effectiveness of the custom embedding layer, BERT and sequence header to capture users' dynamic interest by learning better representation of their interaction history. Improvement in MVBN's performance compared to graph-based models like Diffnet, Diffnet++, and ConsisRec highlights the importance of considering the bi-directional context of user-item interaction. Especially, MVBN achieves substantial improvement compared to the best-performing social SOTA model (DICER), which also uses an attention mechanism. The overall performance of MVBN highlights the improvement gained by embedding layer and BERT in conjunction with Multi-tasking.

V. Ablation Study

An extensive ablation study is carried out to validate the importance of each component in the Multi-View BERT Network (MVBN). For this, we have created three different variants of MVBN named A, B and C. A is modelled by replacing E_β by initial embedding E^0. B is built by adding sequence header embeddings to model A. Both the models A and B do not utilize the Neighbourhood Sampling. Finally, C is modelled by removing the sequence header embedding from

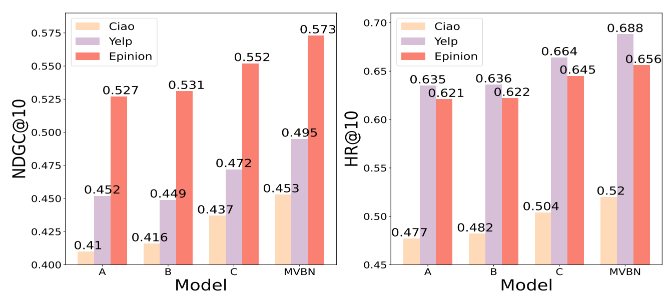

Fig. 2: Study of vaious components of MVBN

the MVBN. Performance comparison of the above variants is demonstrated in Figure 2.

The poor performance of variants A and C compared to MVBN highlights the importance of the sequence header. While We found that omitting neighbour sampling, variant B, significantly degrades performance, emphasising the importance of Neighbourhood sampling which consider only influential neighbours. As compared to model C, MVBN gives significant improvement. This proves that the sequence header with neighbourhood sampling performs well as only the information most about relevant/influential users are being used along with sequence header.

A. Parameter Sensitivity

In MVBN, important hyper-parameters include the number of transformer heads, neighbour percentage, and embedding size. For Ciao and Yelp, two and four respectively, are the suitable number of transformer heads as a single head causes under-fitting, and more heads result in losing critical information differentiating inputs. For both the datasets, MVBN performed best on 0.8 neighbour percentage and if it increases from 0.8 to 1.0, we witnessed a significant drop in scores due to the noise added by non-influential neighbours. Ciao and Yelp have an optimal embedding size of 32. Lower embedding sizes are inadequate for representing node information, whereas larger embedding sizes result in over-fitting.

VI. Conclusion and Future Work

This paper introduces a new framework, MVBN (Multi-View BERT Network) framework for Social Recommendation, emphasizing the importance of sequence headers and neighborhood sampling with BERT to enhance user-item interaction representation. MVBN effectively identifies influential neighbors and utilizes bidirectional contexts from BERT for predicting user-item interactions. Incorporating users' social links in auxiliary tasks for Multi-Task learning further boosts model performance, consistently surpassing state-of-the-art social recommendation algorithms on all three datasets. Future research will delve into session-based user-item interactions and the use of transformers for social recommendation.

References

[1] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. 2019. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 257–266.

[2] Weilin Cong, Rana Forsati, Mahmut Kandemir, and Mehrdad Mahdavi. 2020. Minimal variance sampling with provable guarantees for fast training of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1393–1403.

[3] Mukund Deshpande and George Karypis. 2004. Item-Based Top-N Recommendation Algorithms. *ACM Trans. Inf. Syst.* 22, 1 (jan 2004), 143Äì177. https://doi.org/10.1145/963770.963776

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[5] Wenqi Fan, Qing Li, and Min Cheng. 2018. Deep modeling of social relations for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[6] Bairan Fu, Wenming Zhang, Guangneng Hu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2021. Dual side deep context-aware modulation for social recommendation. In *Proceedings of the web conference 2021*. 2524–2534.

[7] Pan Gu, Yuqiang Han, Wei Gao, Guandong Xu, and Jian Wu. 2021. Enhancing session-based social recommendation through item graph embedding and contextual friendship modeling. *Neurocomputing* 419 (2021), 190–202.

[8] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2015. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.

[9] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 173Äì182. https://doi.org/10.1145/3038912.3052569

[10] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*. 135–142.

[11] Marc Najork, Sreenivas Gollapudi, and Rina Panigrahy. 2009. Less is more: sampling the neighborhood graph makes salsa better and faster. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*. 242–251.

[12] Weiping Song, Zhiping Xiao, Yifan Wang, Laurent Charlin, Ming Zhang, and Jian Tang. 2019. Session-based social recommendation via dynamic graph attention networks. In *Proceedings of the Twelfth ACM international conference on web search and data mining*. 555–563.

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

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[15] Qiang Wang, Yonghong Yu, Haiyan Gao, Li Zhang, Yang Cao, Lin Mao, Kaiqi Dou, and Wenye Ni. 2019b. Network representation learning enhanced recommendation algorithm. *IEEE Access* 7 (2019), 61388–61399.

[16] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019a. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.

[17] Yufei Wen, Lei Guo, Zhumin Chen, and Jun Ma. 2018. Network embedding based recommendation method in social networks. In *Companion Proceedings of the The Web Conference 2018*. 11–12.

[18] Le Wu, Junwei Li, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. 2020b. Diffnet++: A neural influence and interest diffusion network for social recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 10 (2020), 4753–4766.

[19] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020a. SSE-PT: Sequential recommendation via personalized transformer. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 328–337.

[20] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019a. A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. 235–244.

[21] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. 2019b. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *The world wide web conference*. 2091–2102.

[22] Liangwei Yang, Zhiwei Liu, Yingtong Dou, Jing Ma, and Philip S Yu. 2021. Consisrec: Enhancing gnn for social recommendation via consistent neighbor aggregation. In *Proceedings of the 44th international ACM SIGIR conference on Research and development in information retrieval*. 2141–2145.

[23] Hanqing Zeng, Hongkuan Zhou, Ajitesh Srivastava, Rajgopal Kannan, and Viktor Prasanna. 2019. Graphsaint: Graph sampling based inductive learning method. *arXiv preprint arXiv:1907.04931* (2019).

[25] Feng, X., Liu, Z., Wu, W. & Zuo, W. Social recommendation via deep neural network-based multi-task learning. *Expert Systems With Applications*. **206** pp. 117755 (2022)

[26] Wu, Q., Jiang, L., Gao, X., Yang, X. & Chen, G. Feature Evolution Based Multi-Task Learning for Collaborative Filtering with Social Trust.. *IJCAI*. pp. 3877-3883 (2019)

[27] Zhao, T., McAuley, J. & King, I. Leveraging social connections to improve personalized ranking for collaborative filtering. *Proceedings Of The 23rd ACM International Conference On Conference On Information And Knowledge Management*. pp. 261-270 (2014)

[28] Kang, W. & McAuley, J. Self-attentive sequential recommendation. *2018 IEEE International Conference On Data Mining (ICDM)*. pp. 197-206 (2018)

[29] He, X., Deng, K., Wang, X., Li, Y., Zhang, Y. & Wang, M. Lightgcn: Simplifying and powering graph convolution network for recommendation. *Proceedings Of The 43rd International ACM SIGIR Conference On Research And Development In Information Retrieval*. pp. 639-648 (2020)