CR-SoRec: BERT driven Consistency Regularization for Social Recommendation

TUSHAR PRAKASH[*] and RAKSHA JALAN[*], Sony Research India, India
BRIJRAJ SINGH, Sony Research India, India
NAOYUKI ONOE, Sony, Japan

In the real world, when we seek our friends' opinions on various items or events, we request verbal social recommendations. It has been observed that we often turn to our friends for recommendations on a daily basis. The emergence of online social platforms has enabled users to share their opinion with their social connections. Therefore, we should consider users' social connections to enhance online recommendation performance. The social recommendation aims to fuse social links with user-item interactions to offer more relevant recommendations. Several efforts have been made to develop an effective social recommendation system. However, there are two significant limitations to current methods: First, they haven't thoroughly explored the intricate relationships between the diverse influences of neighbours on users' preferences. Second, existing models are vulnerable to overfitting due to the relatively low number of user-item interaction records in the interaction space. For the aforementioned problems, this paper offers a novel framework called CR-SoRec, an effective recommendation model based on BERT and consistency regularization. This model incorporates Bidirectional Encoder Representations from Transformer(BERT) to learn bidirectional context-aware user and item embeddings with neighbourhood sampling. The neighbourhood Sampling technique samples the most influential neighbours for all the users/ items. Further, to effectively use the available user-item interaction data and social ties, we leverage diverse perspectives via consistency regularization to harness the underlying information. The main objective of our model is to predict the next item that a user would interact with based on its interaction behaviour and social connections. Experimental results show that our model defines a new state-of-the-art on various datasets and outperforms previous work by a significant margin. Extensive experiments are also conducted to analyze the proposed method.

CCS Concepts: • **Information systems→Social Recommendation**; • **Computing methodologies→Neural networks**;

Additional Key Words and Phrases: Social Recommendation, BERT , Consistency Regularization

ACM Reference Format:
Tushar Prakash, Raksha Jalan, Brijraj Singh, and Naoyuki Onoe. 2023. CR-SoRec: BERT driven Consistency Regularization for Social Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23), September 18–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3604915.3608844

1 INTRODUCTION

Numerous e-commerce websites and online platforms have evolved into popular social platforms as social media has grown in popularity to improve user engagement. Amazon's OTT platform, Prime Video has introduced "Watch Party", encouraging users to invite friends to watch content simultaneously and socialise virtually. Similarly, a popular music-streaming platform, Spotify, has a feature, "Blend", that allows users to invite friends and share their playlists. With the increasing popularity of such platforms, developing recommendation systems that embrace social interactions

[*]Both authors contributed equally to this research.

into the recommendation model is essential. Social relationships among users may reveal their diverse interest trends and can be utilised for modelling user preferences more accurately. However, the complexity of high-order social relations makes it challenging to extract the most relevant data for modelling user preferences. Existing approaches fail to anticipate the multifaceted relationship between the diverse influences of users' neighbours on their preferences. For the recommendation model to be trained efficiently, there must be an abundance of user-item interactions. However, user-item interaction data are extremely sparse in the interaction space. As a result, models trained on these data are vulnerable to over-fitting.

To address the above mentioned problems, we propose a novel framework -BERT driven Consistency Regularization for Social Recommendation (CR-SoRec). It learns efficient user-item interaction and user-user social representations by leveraging BERT along with Consistency Regularization Framework. In this paper, we have proposed an innovative way to generate robust user-item interactions representation (or social links representation) by utilizing user header with neighbourhood sampling. Neighbourhood sampling is performed to capture the multifaceted relationship between users' neighbours' diverse influences on their preferences. The proposed method also helps in eliminating insignificant signal from user-item interaction history. Next, to improve data diversity and model's robustness, we have performed data augmentation to generate different views of user-item interaction and social links from the original data. This generated data is utilised to design consistency regularization framework to learn a smooth and stable decision boundary that is robust against realistic perturbation in the input data. Since the success of Consistency Regularization(CR) framework heavily depends on the quality of sequence representation, we have utilised BERT architecture to embrace the bidirectional contexts of input sequence.

Hence, the proposed network is trained by jointly minimizing a combination of three types of losses, (a) A standard supervised loss on labeled data (b) CR penalty on augmented data for user-item interaction and (c) CR penalty for user-user social data.

The main contributions of the paper are as follows:

- Proposed a novel way to learn User/Item representations based on neighbourhood sampling in conjunction with BERT.
- To improve the generalizability of the model, we designed two novel Consistency Regularization(CR) tasks- Item CR and Social CR.
- Proposed a new way to utilize social connection and user-item interactions with CR to enhance social recommendation performance.

2 RELATED WORK

Social Recommendations have received a lot of attention since social relations give an additional perspective on user preferences along with item ratings. Earlier, various Matrix Factorization (MF) based approaches were developed where they co-factorize the rating and social link matrices by sharing a common user latent feature matrix like SocialMF[14], TrustSVD[12] and [19, 30, 42, 45]. Multiple deep learning based techniques[7, 9, 13, 31, 34] have evolved which incorporates social relationships into the recommendation .

Later, multiple deep learning techniques have been developed for recommendation systems [13, 29]. In the domain of social recommendation, methods like DSC[9], DeepSoR[34] and [7, 31] incorporate social relationships into deep learning based recommendation generation.Recently, methods like GraphRec[8], SocialGCN[38], Diffnet[37], DiffNet++[35] and [15, 18, 29, 32, 32, 39] utilized Graph Neural Network to enhance the performance of social recommendations.

Attention based networks have also demonstrated promising performance in a variety of social recommendation models like DICER[10] and [20, 33, 47]. To capture users' dynamic interests, various sequential recommendation algorithms [11, 16, 25] have emerged, which extract features from social relations and user behavior sequences. Recently, there has been rising enthusiasm for using self-attention-based models like Transformer [28] and BERT [6] in the representation learning since they have produced impressive results in text sequence modelling. Popular recommendation models like Bert4Rec[26] ,Transformers4rec[4]and SSE-PT[36] have adopted transformers for building recommendation system.

However, existing studies have limitations, mainly focusing on first-order local neighbors and neglecting helpful information from distant neighbors. Additionally, most methods for modeling user interests treat information from friends equally, disregarding the specific recommendation context and resulting in shallow context-aware aggregation of friends' information.

Consistency Regularization[1, 22, 23] is the method that helps train the model in such a way that makes it augmentation invariance. Recently [17, 43] have incorporated consistency regularization to improve sequential recommendation model's performance. However, no prior research in the domain of social recommendations has leveraged CR framework and self-attention-based networks to predict users' preferences.

3 METHODOLOGY

In this section, we first briefly outline the research problem, followed by a detailed discussion of the architecture of the proposed model CR-SoRec and its components as presented in Figure 1. At last, we discuss the training procedure of the model.

3.1 Problem Statement

In the Social recommendation, we have a user-item interaction matrix $\mathcal{R}$ for users $U = \{u_1, u_2, ...u_n\}$ and items $V = \{v_1, v_2, ...v_m\}$, along with a user-user social network N. Here, $\mathcal{R} \in \{0, 1\}$ based on the interaction of users U with items V and N 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 social recommendation is to predict the next item v_{ij} that a user u_i will interact with, given u_i's interaction history $\mathcal{H}_{uv} = \{v_{i1}, v_{i2}, ..., v_{in}\}$, and it's corresponding social network N^u. Therefore, we have addressed this problem as a next-item prediction task, referred to as a recommendation task ($\mathcal{T}_R$). To accomplish this, we propose a method based on a novel embedding generation layer and the BERT architecture. To further boost its performance, the $\mathcal{T}_R$ is supported by two Consistency Regularization (CR) tasks, i.e., Item CR task ($\mathcal{T}_{I-CR}$) and Social CR task ($\mathcal{T}_{S-CR}$).

3.2 Embedding Generation Layer

Several studies on social recommendation have found a positive correlation between users' social behaviour and their item interactions [2, 3]. In order to efficiently capture this correlation, we have proposed to enrich user and item embeddings with their most influential neighbours by performing Neighbourhood Sampling (NS) [44].

In the recommendation task $\mathcal{T}_R$, we are given a sequence of items $\mathcal{I}_v = \{v_1, v_2, v_3, ...v_m\}$ that a user u_i has interacted with. To generate a training sample, we randomly mask some items in the sequence to obtain $\mathcal{I}_v^{(m)} = \{v_1, [mask], v_3, ...v_m\}$, using the classical Cloze's task [27]. To represent each item v_i, we define an embedding E_a^v, which is the concatenation of the item-user interaction history $\mathcal{H}vu$ and item-item similarity $\mathcal{S}vv$. Here, item-user interaction history $\mathcal{H}vu$ consists a list of all users who have interacted with item v_i in the past, while item-item similarity $\mathcal{S}vv$ represents similar items, having more than 50% of common users as proposed by Sarwar et al. [24].

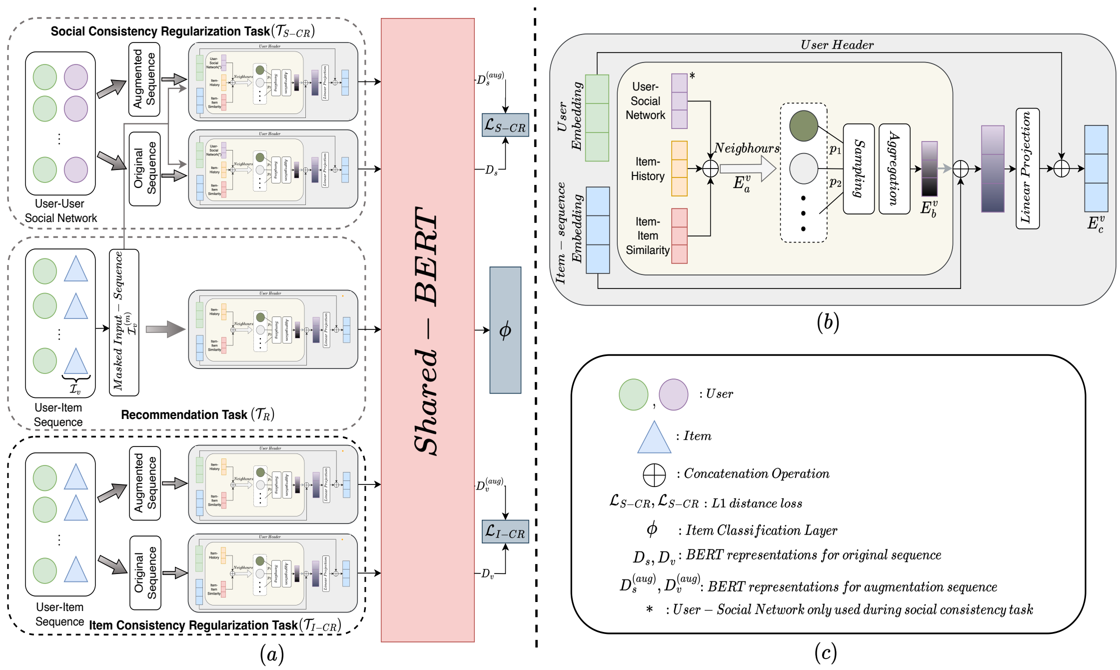


Fig. 1. CR-SoRec Framework (a) Illustrates the architecture of CR-SoRec model, (b) Details of shared Embedding Generation Layer, (c) List of Notations, where ∗ represents User-Social Network is used only for Social CR task

Next, the neighbourhood sampling has been performed on E_a^v using multinomial distribution [21] to generate a new embedding E_b^v. This step helps to incorporate the information of the most influential neighbours. To bring similar user-item pairs close in the embedding space, we introduce a user header u_i that generates the user embedding E_u. Finally, we concatenate E_b^v, E_0^v (the embedding of the items in the $I_v^{(m)}$), and E_u to generate the final embedding E_c^v. This is achieved through a linear projection with weights W followed by a sigmoid activation function, as shown below:

$$E_c^v = E_u \oplus \sigma(W^T(E_0^v \oplus E_b^v)) \tag{1}$$

This entire process of embeddings generation for task $\mathcal{T}_R$ is illustrated in figure 1(b) and mentioned in $embed\_generation()$ function of Algorithm-1.

3.3 BERT Network

The BERT model [6] is a powerful sequential model that is built on top of a multi-layer bidirectional transformer encoder. Our proposed method utilizes the strengths of BERT's multiple bidirectional transformer layers and self-attention mechanism.

To learn a deep bidirectional representation, we prepare the input sequences $I_v^{(m)}$ for the recommendation task, social CR task, and I_v for the item CR task. These sequences are used to generate E_c^v embeddings, as discussed in section 3.2, which are then fed into a shared BERT network. The BERT network is made up of transformer layers that contain a Multi-Head Self-Attention (MH) layer and a Position-wise Feed-Forward Network. The MH layer is responsible for capturing long-range dependencies between representation pairs in the sequences. This is achieved by linearly projecting the hidden representations of the input sequence for each layer and position into multiple subspaces. Then, multiple attention functions are applied in parallel to produce the output representations. The attention function uses query, key, and value matrices to compute a Scaled Dot-Product Attention. The output representations are then passed through a Position-wise Feed-Forward Network. The detailed architecture of BERT can be found in [6].

3.4 Model training with Consistency Regularization

The proposed network is trained by jointly minimizing a combination of three types of losses, (a) $\mathcal{L}_R$: Cross-entropy for task $\mathcal{T}_R$, (b) $\mathcal{L}_{I-CR}$: $L1$ distance loss for Item CR Task $\mathcal{T}_{I-CR}$ and (c) $\mathcal{L}_{S-CR}$: $L1$ distance loss for Social CR Task $\mathcal{T}_{S-CR}$. The proposed framework is shown in figure 1(a).

3.4.1 ***Recommendation Task($\mathcal{T}_R$).*** This is the main task, which predicts the masked item values v_o of input sequence $I_v^{(m)}$ as shown in line 7 of Algorithm-1. Let $\mathcal{F}_\beta(.)$ be the trainable function of the BERT network, and $\mathcal{F}_\psi(.)$ is the classification layer for predicting the masked items. For each user, E_c^v embedding is generated as mentioned in section 3.2. Then, E_c^v is fed into BERT to learn representation $\mathcal{D}$. Finally, this representation is passed into the classification layer to predict the masked values. The detailed steps are presented in the Algorithm-1.

3.4.2 ***Consistency Regularization Tasks.*** To further improve the model's performance and address over-fitting, we employed Consistency regularization(CR) on the item interaction sequence and user's social network. To increase data diversity, different views of the original sequence are generated by performing augmentation. The CR model generates its prediction for the original sequence first and then for the augmented input sequence. The model prediction is encouraged to be consistent with the augmented version of the same input sequence. This is achieved by introducing a penalty term into the model's loss function. Both Item Consistency Regularization and Social Consistency Regularization supports recommendation task $\mathcal{T}_R$.

Item Consistency Regularization Task ($\mathcal{T}_{I-CR}$) : For this task, we have the original item input-sequence $\mathcal{I}_v$, and its augmented version $\mathcal{I}_v^{(aug)}$. Then, these sequences are passed to function $embed\_generation$ (line 13-21, Algorithm-2) to create the final embeddings E_c^v and $E_c^{v(aug)}$, respectively as shown in line 2,6 in Algorithm-2. Since we only utilize item interactions in this task, we pass N^u as None while calling $embed\_generation$. Further, these embeddings are fed into BERT to learn representations $\mathcal{D}_v$ and $\mathcal{D}_v^{(aug)}$. Finally, we return penalty loss $\mathcal{L}_{I-CR}$, computed by enforcing the similarity in both representations and minimizing the L1 distance between them.

Social Consistency Regularization Task ($\mathcal{T}_{S-CR}$) : For this task, we have augmented user social network N^u to generate $N^{u(aug)}$. Here item-user history $\mathcal{H}_{vu}$ and item-item similarity matrix $\mathcal{S}_{vv}$ are generated from the same masked input sequence $I_v^{(m)}$ same as task $\mathcal{T}_R$. This CR task incorporates additional input N^u along with item-user history $\mathcal{H}_{vu}$ and item-item similarity $\mathcal{S}_{vv}$, to generate the embeddings E_a and $E_a^{v(aug)}$. These intermediate embeddings are being used to create the final embeddings E_c^v and $E_c^{v(aug)}$. These embeddings are then passed to BERT computing CR penalty loss $\mathcal{L}_{S-CR}$ similar to task $\mathcal{T}_{I-CR}$ as shown in Algorithm-3.
Please refer Algorithm-1 for detailed training steps of CR-SoRec.

4 EXPERIMENTS AND EVALUATIONS

This section aims to provide the experimental detail under subsections as experimental settings, performance evaluation, ablation study and parameter sensitivity.

4.1 Experimental Settings

4.1.1 Datasets. We have evaluated our proposed framework on Epinions[1], Ciao[2] and Yelp[3]. Epinions is a who-trusts-whom-directed online social network that offers product rating and review services. The Ciao dataset includes customer reviews of purchased items and their social connections. Since we are interested in the implicit feedback, we convert the detailed ratings into a value of 0 or 1, indicating whether the user has rated the item. The statistical details of these datasets are summarised as follows: Epinions has 22166 users, 296277 items, and 398751 social links, whereas Ciao has 7375 users, 105114 items, and 115632 social links and Yelp has 17235 users, 37378 items, and 155731 social links .

Algorithm 1 Algorithm for CR-SoRec

for i in $(1, N)$ **do** ▷ for N users
  $\mathcal{I}_v$, N^u ▷ given for user u_i
  $\mathcal{I}_v^{(m)}$ = masking($\mathcal{I}_v$)
   ▷ Making masked input item-sequence
  $\mathcal{H}_{vu}, \mathcal{S}_{vv} \Leftarrow \mathcal{I}_v^{(m)}$
  $E_c^v = embed\_generation(\mathcal{H}_{vu}, \mathcal{S}_{vv}, None)$
  $\hat{v} = \mathcal{F}_\psi(\mathcal{F}_\beta(E_c^v))$
  $\mathcal{L}_R = \mathcal{L}_{ce}(v_o, \hat{v})$
  $\mathcal{L}_{\mathcal{I}-\mathcal{CR}} = ItemSequenceConsistency(I_v)$
  $\mathcal{L}_{\mathcal{S}-\mathcal{CR}} = SocialSequenceConsistency(I_v^{(m)}, N^u)$
  $\mathcal{L} = \mathcal{L}_R + \alpha * \mathcal{L}_{\mathcal{I}-\mathcal{CR}} + \gamma * \mathcal{L}_{\mathcal{S}-\mathcal{CR}}$
end for
embed_generation$(\mathcal{H}_{vu}, \mathcal{S}_{vv}, N^u)$
  if N^u is None **then**
    $E_a^v = \mathcal{H}_{vu} \oplus \mathcal{S}_{vv}$
  else
    $E_a^v = \mathcal{H}_{vu} \oplus \mathcal{S}_{vv} \oplus N^u$
    $E_b^v = NS(E_a^v)$
    $E_c^v = E_u \oplus \sigma(W^T(E_0^v \oplus E_b^v))$
  end if
end embed_generation$(\mathcal{H}_{vu}, \mathcal{S}_{vv}, N^u)$

Algorithm 2 Item Consistency Regularization

$\mathcal{H}_{vu}, \mathcal{S}_{vv} \Leftarrow \mathcal{I}_v$
$E_c^v = embed\_generation(\mathcal{H}_{vu}, \mathcal{S}_{vv}, None)$
$\mathcal{D}_v = \mathcal{F}_\beta(E_c^v)$
$\mathcal{I}_v^{(aug)}$ = aug($\mathcal{I}_v$) ▷ Making augmented input item-sequence
$\mathcal{H}_{vu}, \mathcal{S}_{vv} \Leftarrow \mathcal{I}_v^{(aug)}$ ▷ Get item-user history and item-item similarity using $\mathcal{I}_v^{(aug)}$
$E_c^{v(aug)} = embed\_generation(\mathcal{H}_{vu}, \mathcal{S}_{vv}, None)$
$\mathcal{D}_v^{(aug)} = \mathcal{F}_\beta(E_c^{v(aug)})$
$\mathcal{L}_{\mathcal{I}-\mathcal{CR}} = \mathcal{L}1(\mathcal{D}_v, \mathcal{D}_v^{(aug)})$

Algorithm 3 Social Consistency Regularization

$\mathcal{H}_{vu}, \mathcal{S}_{vv} \Leftarrow \mathcal{I}_v^{(m)}$ ▷ Get item-user history and item-item similarity
$E_c^v = embed\_generation(\mathcal{H}_{vu}, \mathcal{S}_{vv}, N^u)$
$\mathcal{D}_s = \mathcal{F}_\beta(E_c^v)$
$\mathcal{H}_{vu}, \mathcal{S}_{vv}, N^u \Leftarrow \mathcal{I}_v$
$N^{u(aug)}$ = aug(N^u)
$E_c^{v(aug)} = embed\_generation(\mathcal{H}_{vu}, \mathcal{S}_{vv}, N^{u(aug)})$
$\mathcal{D}_s^{(aug)} = \mathcal{F}_\beta(E_c^{v(aug)})$
$\mathcal{L}_{S-CR} = \mathcal{L}1(\mathcal{D}_v, \mathcal{D}_v^{(aug)})$

4.1.2 Baselines. To demonstrate the effectiveness of our proposed model, we compare it with five strong baselines from social recommendations and our variant of CR-SoRec. TrustSVD[12] is a popular trust-based matrix factorization technique. DiffNet [37], DiffNet++ [35] considers the social influence diffusion process. DICER[10] utilizes a GNNs and attention mechanism to exploit high-order neighbour information. ConsisRec[41] simulates a GNN-based architecture to learn consistent node embeddings for rating prediction. As our objective is to anticipate the interactions, we transform

[1] https://www.cse.msu.edu/%7Etangjili/datasetcode/epinions.zip
[2] https://www.cse.msu.edu/%7Etangjili/datasetcode/ciao.zip
[3] https://www.yelp.com/dataset

Table 1. Performance evaluation of proposed model as compared to other methods on Epinion and Ciao dataset. Improvement has been shown against the best-performing SOTA method, i.e., DICER (underlined). Improvements are statistically significant with p < 0.05.

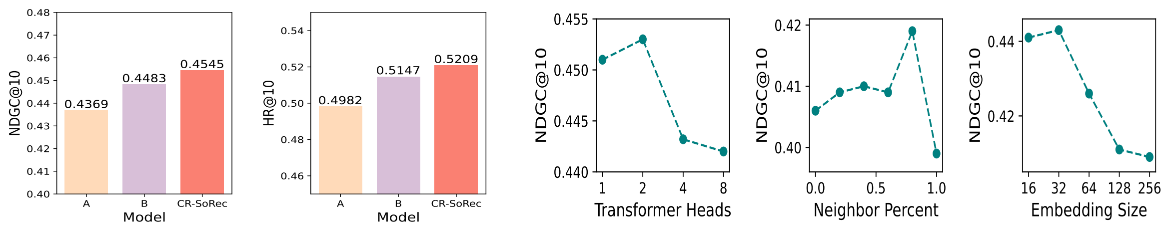
Epinion	**HR@5**	**HR@10**	**HR@15**	**NDCG@5**	**NDCG@10**	**NDCG@15**
SBPR[46]	0.2611	0.3352	0.3721	0.2310	0.2553	0.2745
SocialMF[14]	0.2656	0.3402	0.3797	0.2379	0.2601	0.2816
NCF[5]	0.2697	0.3414	0.3771	0.2398	0.2598	0.2763
NGCF[32]	0.2826	0.3571	0.3987	0.2519	0.2744	0.2931
DiffNet[40]	0.3041	0.3748	0.4222	0.2716	0.2976	0.3130
DiffNet++ [35]	0.3202	0.4049	0.4521	0.2770	0.3064	0.3221
ConsisRec [41]	0.3775	0.4658	0.5299	0.3545	0.3889	0.4114
DICER [10]	<u>0.4051</u>	<u>0.5111</u>	<u>0.5872</u>	<u>0.3541</u>	<u>0.3935</u>	<u>0.4182</u>
LSTM_SoRec	0.5261	0.6044	0.6478	0.4786	0.5067	0.5205
CR-SoRec(ours)	**0.5966**	**0.6569**	**0.6881**	**0.5566**	**0.5769**	**0.5863**
Improvement	47.27%	28.52%	17.18%	57.18%	46.60%	40.19%
Ciao	**HR@5**	**HR@10**	**HR@15**	**NDCG@5**	**NDCG@10**	**NDGC@15**
SBPR[46]	0.2018	0.2523	0.2883	0.1825	0.1993	0.2139
SocialMF[14]	0.2099	0.2609	0.2995	0.1946	0.2094	0.2202
NCF[5]	0.2045	0.2576	0.2891	0.1955	0.2037	0.2160
NGCF[32]	0.2359	0.2844	0.3160	0.2137	0.2297	0.2378
DiffNet [40]	0.2329	0.2820	0.3183	0.2198	0.2353	0.2469
DiffNet++ [35]	0.2471	0.2996	0.3403	0.2320	0.2494	0.2628
ConsisRec [41]	0.2761	0.3656	0.4331	0.2499	0.2837	0.3066
DICER [10]	<u>0.3221</u>	<u>0.4002</u>	<u>0.4678</u>	<u>0.3003</u>	<u>0.3278</u>	<u>0.3505</u>
LSTM_SoRec	0.4240	0.4942	0.5446	0.4004	0.4241	0.4410
CR-SoRec(ours)	**0.4537**	**0.5209**	**0.5664**	**0.4322**	**0.4545**	**0.4692**
Improvement	40.85%	30.15%	21.07%	43.92%	38.65%	33.86%
Yelp	**HR@5**	**HR@10**	**HR@15**	**NDCG@5**	**NDCG@10**	**NDGC@15**
SBPR[46]	0.1820	0.2797	0.3502	0.1359	0.1688	0.1901
SocialMF[14]	0.1855	0.2816	0.3586	0.1387	0.1703	0.1916
NCF[5]	0.2012	0.3041	0.3811	0.1504	0.1865	0.2107
NGCF[32]	0.2017	0.3053	0.3804	0.1522	0.1853	0.2120
DiffNet [40]	0.2215	0.3251	0.4012	0.1628	0.1983	0.2209
DiffNet++ [35]	0.2358	0.3417	0.4178	0.1789	0.2126	0.2348
ConsisRec [41]	0.3071	0.4207	0.5024	0.2175	0.2574	0.2812
DICER [10]	<u>0.4570</u>	<u>0.6344</u>	<u>0.7411</u>	<u>0.3472</u>	<u>0.4138</u>	<u>0.4473</u>
LSTM_SoRec	0.4754	0.6156	0.6930	0.3760	0.4293	0.4536
CR-SoRec(ours)	**0.5462**	**0.6907**	**0.7567**	**0.4399**	**0.4941**	**0.5150**
Improvement	19.51%	8.87%	2.10%	26.69%	19.40%	15.13%

the detailed ratings into 1 or 0, indicating whether or not the user rated the item. Finally, LSTM_SoRec is a variant of our model built by replacing the BERT module with bidirectional LSTM.

4.1.3 Evaluation Metrics. We have adopted standard evaluation metrics Normalized Discounted Cumulative Gain (NDCG) [5], [13] and Hit Ratio(HR) a[5] to evaluate the recommendation performance. For all our experiments,we have considered NDCG@10 as our primary evaluation metric. To evaluate the performance, we randomly sampled 100 non-interacted items for each user as negative samples similar to [37], [13]. Then, we combine the corresponding

Table 2. NDCG@10 and HR@10 of CR-SoRec on Ciao dataset for different augmentation techniques

Ciao	Reordering	Masking	Cropping
NDCG@10	0.4438	0.4518	0.4545
HR@10	0.5087	0.5166	0.5209



(a) Albation Study of proposed CR-SoRec

(b) Parameter sensitivity

Fig. 2. Ablation Study and Parameter Sensitivity

positive and negative samples in the test set to identify the top N possible candidates. Each experiment is performed 5 times to eliminate inconsistency in this process.

4.1.4 Experiments Details. Each dataset is randomly split into 80% training and 10% validation and 10% testing. For hyperparameters tuning, we have applied grid search that is explained in section4.3. At inference stage, we need to perform only Recommendation task of CR-SoRec to generate predictions on test dataset.

4.2 Performance Evaluation

The results of all experiments are shown in Table 1. We can observe that our proposed model CR-SoRec notably outperformed existing baselines on all the datasets. CR-SoRec performance improvement compared to attention-based model DICER and graph-based models like Diffnet, Diffnet++, and ConsisRec highlights the importance of considering the bi-directional context of user-item interaction while learning its representation. The striking improvements of CR-SoRec over the current deep learning-based Social Recommendation models demonstrate the potency of Consistency Regularization when combined with the proposed embedding layer and BERT. This helps in capturing users' dynamic interest by understanding a better representation of their interaction history. Further, to validate the significance of BERT for our task, we replaced it with LSTM in the proposed model and named as LSTM_SoRec. Here, complete architecture of CR-SoRec along with CR framework is retained, only shared BERT is replaced with shared bidirectional LSTM. As in Table1, we conclude that the BERT is more capable of creating rich representations of users' behaviour sequences to enhance recommendation performance given it's bi-directional context learning.

4.3 Ablation Study and Parameter Sensitivity

To study the impact of CR on the CR-SoRec model's performance, we present three variants of the CR-SoRec model as shown in Figure 2a: Model A is trained with just main next-item prediction task. Next, Model B is built by adding an item-CR module to Model A .While the final model CR-SoRec is built by adding a social CR module to model B. With these experiments we can see that utilizing different views of existing information in addition to original data can lead to performance improvement. We have found that the success of the CR framework is related to the quality and diversity of input perturbations. For our task, we have experimented with augmentation techniques like -cropping, masking and reordering of the input sequences and recorded their impact on the proposed model's performance. As shown in table 2, we conclude that cropping is a best choice of data augmentation for our task.

We discovered that the performance of CR-SoRec is mostly determined by hyper-parameters such as the number of transformer heads, neighbour proportion, and embedding size. Figure 2b shows NDCG@10 for various parameters for the Ciao dataset. We experimented with number of transformer heads in the range of 2 to 8. We found that 2 heads are ideal for Ciao, as one head leads to under-fitting and more heads overlook important features differentiating inputs. We looked for the best neighbour percentage in the range of 0.2 to 1.0 for the neighbourhood sampling. When the neighbour percentage increases from 0.8 to 1.0, we witness a significant drop in scores due to the noise introduced by non-influential neighbours. Ciao's optimal embedding size is 32. As we searched for *alpha* and *gamma*, the CR tasks' weights ranging from 0.1 to 1.0 , we found that *alpha* and *gamma* were the most significant at 0.5 and 0.1 respectively. To avoid over-fitting, all experiments employed early stopping based on our primary metric NDCG@10.

5 CONCLUSION

This paper proposes a novel framework called BERT driven Consistency Regularization for Social Recommendation (CR-SoRec). We have shown the significance of employing the user header and neighbourhood sampling to learn a rich representation in conjunction with BERT for Social Recommendation. In CR-SoRec, BERT offers the consideration of bidirectional contexts while predicting the next user-item interaction. To further improve the model's performance, we proposed an innovative way of incorporating multiple views of user-item interactions and users' social links in Consistency Regularization framework via Item CR task and Social CR task. Our model consistently outperformed the state-of-the-art social recommendation algorithm throughout the experiments on all the datasets.

REFERENCES

[1] Philip Bachman, Ouais Alsharif, and Doina Precup. 2014. Learning with pseudo-ensembles. *Advances in neural information processing systems* 27 (2014).

[2] 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.

[3] 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.

[4] Gabriel de Souza Pereira Moreira, Sara Rabhi, Jeong Min Lee, Ronay Ak, and Even Oldridge. 2021. Transformers4rec: Bridging the gap between nlp and sequential/session-based recommendation. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 143–153.

[5] Mukund Deshpande and George Karypis. 2004. Item-Based Top-N Recommendation Algorithms. *ACM Trans. Inf. Syst.* 22, 1 (jan 2004), 143–177. https://doi.org/10.1145/963770.963776

[6] 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).

[7] 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.

[8] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *The world wide web conference*. 417–426.

[9] Wenqi Fan, Yao Ma, Dawei Yin, Jianping Wang, Jiliang Tang, and Qing Li. 2019. Deep social collaborative filtering. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 305–313.

[10] 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.

[11] 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.

[12] 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.

[13] 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, 173–182. https://doi.org/10.1145/3038912.3052569

[14] 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.

[15] Munan Li, Kenji Tei, and Yoshiaki Fukazawa. 2020. An efficient adaptive attention neural network for social recommendation. *IEEE Access* 8 (2020), 63595–63606.

[16] Chun Liu, Yuxiang Li, Hong Lin, and Chaojie Zhang. 2022. GNNRec: Gated graph neural network for session-based social recommendation model. *Journal of Intelligent Information Systems* (2022), 1–20.

[17] Chong Liu, Xiaoyang Liu, Rongqin Zheng, Lixin Zhang, Xiaobo Liang, Juntao Li, Lijun Wu, Min Zhang, and Leyu Lin. 2021. C^2-Rec: An Effective Consistency Constraint for Sequential Recommendation. *arXiv preprint arXiv:2112.06668* (2021).

[18] Zhiwei Liu, Mengting Wan, Stephen Guo, Kannan Achan, and Philip S Yu. 2020. Basconv: Aggregating heterogeneous interactions for basket recommendation with graph convolutional neural network. In *Proceedings of the 2020 SIAM International Conference on Data Mining*. SIAM, 64–72.

[19] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*. 287–296.

[20] Nan Mu, Daren Zha, Yuanye He, and Zhihao Tang. 2019. Graph attention networks for neural social recommendation. In *2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI)*. IEEE, 1320–1327.

[21] 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.

[22] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. 2016. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems* 29 (2016).

[23] Laine Samuli and Aila Timo. 2017. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations (ICLR)*, Vol. 4. 6.

[24] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. 285–295.

[25] 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.

[26] 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.

[27] Wilson L Taylor. 1953. "Cloze procedure": A new tool for measuring readability. *Journalism quarterly* 30, 4 (1953), 415–433.

[28] 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).

[29] Chen Wang, Yueqing Liang, Zhiwei Liu, Tao Zhang, and S Yu Philip. 2021. Pre-training graph neural network for cross domain recommendation. In *2021 IEEE Third International Conference on Cognitive Machine Intelligence (CogMI)*. IEEE, 140–145.

[30] Menghan Wang, Xiaolin Zheng, Yang Yang, and Kun Zhang. 2018. Collaborative filtering with social exposure: A modular approach to social recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

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

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

[33] Chunyu Wei, Yushun Fan, and Jia Zhang. 2022. Time-aware Service Recommendation with Social-powered Graph Hierarchical Attention Network. *IEEE Transactions on Services Computing* (2022).

[34] 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.

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

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

[37] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019. 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.

[38] Le Wu, Peijie Sun, Richang Hong, Yanjie Fu, Xiting Wang, and Meng Wang. 2018. Socialgcn: An efficient graph convolutional network based model for social recommendation. *arXiv preprint arXiv:1811.02815* (2018).

[39] Le Wu, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. 2018. Collaborative neural social recommendation. *IEEE transactions on systems, man, and cybernetics: systems* 51, 1 (2018), 464–476.

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

[41] 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.

[42] Weilong Yao, Jing He, Guangyan Huang, and Yanchun Zhang. 2014. Modeling dual role preferences for trust-aware recommendation. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 975–978.

[43] Xu Yuan, Hongshen Chen, Yonghao Song, Xiaofang Zhao, Zhuoye Ding, Zhen He, and Bo Long. 2021. Improving sequential recommendation consistency with self-supervised imitation. *arXiv preprint arXiv:2106.14031* (2021).

[44] 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).

[45] Zhijun Zhang, Gongwen Xu, Pengfei Zhang, and Yongkang Wang. 2017. Personalized recommendation algorithm for social networks based on comprehensive trust. *Applied Intelligence* 47, 3 (2017), 659–669.

[46] Tong Zhao, Julian McAuley, and Irwin King. 2014. Leveraging social connections to improve personalized ranking for collaborative filtering. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*. 261–270.

[47] Jinghua Zhu, Zhichao Li, Chenbo Yue, and Yong Liu. 2019. Trust-aware group recommendation with attention Mechanism in social network. In *2019 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*. IEEE, 271–276.