Bi-Transformers-Aided Contextual Contrastive Learning for Sequential Recommendation

Adel Alkhalil[1], Ikhlaq Ahmed[2], Zafran Khan[3], Mazhar Abbas[4], Aakash Ahmad[5], Abdulrahman Albarrak[6]
Department of Software Engineering-College of Computer Science and Engineering,
University of Hail, Saudi Arabia[1,6]
Department of Computer Software Engineering, National University of Sciences and Technology, Islamabad Pakistan[2]
School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST),
Gwangju, South Korea[3]
College of Business-School of Technology Management and Logistics, Universiti Utara Malaysia, Malaysia[4]
School of Computing and Communications, Lancaster University Leipzig, Germany[5]

***Abstract*—Contrastive learning (CL) based on Transformer sequence encoders offers a robust framework for sequential recommendation by effectively addressing data noise and sparsity issues. By utilizing the advantages of CL, these models are able to learn rich representations from sequences of user historical interactions, leading to improved recommendation and user satisfaction. However, recent CL methods are affected by two limitations. Firstly, CL approaches are mainly designed to process input sequences in single direction, i.e., left to right, which is sub-optimal for sequential prediction tasks because user historical interactions might not be in a fixed single direction sequence. Secondly, these models focus on designing CL objectives based solely on the input sequence, overlooking the valuable self-supervision features available as auxiliary information of descriptive text. To overcome these limitations, we introduce a new framework named Bi-Transformers aided Contextual Contrastive Learning for Sequential Recommendation (CCLRec). Specifically, bidirectional Transformers are extended to incorporate auxiliary information by using sentence embedding formulated from item's textual description. Next, we introduce the rolling glass step technique for handling lengthy user sequence and descriptive features of corresponding item, which enables more refined partitioning of user sequences. Next, the cloze task, random occlusion, and dropout masking strategies are jointly applied to generate high-quality positive samples, enabling improved performance of the contrastive learning objective. Comprehensive experiments upon three benchmark datasets demonstrate that CCLRec consistently outperforms state-of-the-art baselines, achieving improvements of up to 5.69% to 6.34% in NDCG@10 across the MovieLens-1M, Amazon Beauty, and Amazon Toys datasets.**

Keywords*—*Contextual sequential recommendation; bidirectional transformers; contrastive learning; auxiliary information

I. Introduction

Recommender systems (RS) predict the user's future preferences by characterizing the user's intent that are usually dynamic in nature. RS is extensively deployed across e-commerce platforms and online media streaming services. User interests are usually not stable and keep on changing with time. This temporal aspect is crucial in acquiring user dynamic preferences. To better capture user intents, many sequential recommendation (SR) methods have recently been proposed that leverage user's historical interaction behaviors [1]. The SR models work in two phases. First they gather the sequence of past objects from user's history and then project the most relevant and accurate interaction for each user.

Earlier approaches modeled user preferences with Markov chain-based techniques [2], [3], followed by neural architectures such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to better capture dynamic patterns in user interactions [4], [5], [6], [7].

Subsequently, Transformer models introduced self-attention mechanisms for encoding sequential user behaviors [8]. SASRec [9] leveraged a unidirectional Transformer architecture, while BERT4Rec [10] improved upon this by employing a bidirectional Transformer with cloze-style masking. Extensions, such as KeBERT4Rec [11] and FDSA [12], have attempted to incorporate keyword or attribute information, yet most SR models continue to rely primarily on item identifiers, neglecting richer auxiliary data such as textual descriptions or reviews.

To mitigate sparsity and noise, contrastive learning (CL) has been combined with Transformer-based models. Approaches such as CL4SRec [13] and CoSeRec [14] propose sequence-level augmentation strategies, while DuoRec [15] combines unsupervised dropout with supervised positive sampling. ICLRec [16] models user intents within an expectation-maximization framework, and CBiT [17] adopts BERT4Rec as a bidirectional encoder with dropout-based augmentation.

However, the effectiveness of contrastive sequential recommendation methods is constrained by three key limitations. First, the majority of prior studies [13], [14], [15] design contrastive learning strategies around unidirectional Transformers, thereby overlooking the additional insights provided by bidirectional Transformer models. Unidirectional Transformers handle information in a strictly left-to-right sequence; however, in practical settings, user behavior patterns are often more complex and cannot be fully represented through one-directional sequential modeling alone. In contrast, bidirectional Transformer attention captures item relationships by leveraging both prior and subsequent contexts, resulting in a more complete contextual representation and frequently achieving superior performance [10] compared to unidirectional approaches SASRec [9]. Next, most SASRec and BERT4Rec based CL models consider only implicit or explicit feedback based on sequence level item identifier neglecting the rich auxiliary data (textual descriptions, reviews, etc.). By incorporating additional information, prediction accuracy of next items can be increased.

Third, deciding maximum length of descriptive features is crucial for RS due to computational constraints. This highlights the need for efficient processing and analysis of large-scale textual data in order to capture richer and more fine-grained contextual information. Therefore, we argue that additional auxiliary information into the contrastive learning techniques for SR improves recommendations, particularly under sparse situations.

To address these limitations, this study identifies a clear research gap: no existing work simultaneously leverages bidirectional Transformers, rich auxiliary textual information, and contrastive learning within a unified sequential recommendation framework.

Guided by this gap, this study seeks to answer the following research question: How can bidirectional Transformers and contextual auxiliary information be effectively integrated into a contrastive learning framework to improve sequential recommendation performance?

Motivated by this research question, we introduce a new framework named Bi-Transformers aided Contextual Contrastive Learning for Sequential Recommendation (CCLRec), which effectively exploits contextual item descriptions. Specifically, we apply the rolling glass step technique to handle lengthy user sequences and descriptive features, optimizing computational cost while capturing deep contextual information. We then enhance the bidirectional Transformer with a context-driven self-attention module that fuses multiple attention mechanisms to jointly capture complex dependencies between sequences and textual descriptions. Next, we introduce a context-driven contrastive learning objective that constructs positive samples using three augmentation strategies: cloze-style masking, dropout-based masking, and random occlusion. Extensive experiments on three benchmark datasets are conducted to evaluate the proposed framework. The proposed paradigm injects the following contributions in the recommendation systems domain:

- Introduction of a new sequential recommendation framework that effectively utilizes auxiliary contextual information.
- Development of a context-driven sequence encoder that produces contextual embeddings from item descriptions using Sentence-BERT, while jointly capturing the complex interactions between user sequences and their corresponding text by fusing multiple attention outputs into a unified representation.
- A context-driven CLL objective function is proposed that forms a set of +ive sample(s) using triplet augmentation techniques: cloze-style masking at the data level, dropout-based masking at the model level, and random occlusion.
- Performs exhaustive experiments using three publicly available bench-marked dataset(s).

The research study is organized section-wise in the following pattern: Section II covers the literature study of the relevant domain; Section III is about proposed paradigm architecture; Section IV presents the empirical evaluation of the model; and the final Section V not only summarizes the findings of the study but also outlines potential directions for future research.

II. Literature Review

A. Sequential Recommendation

Sequential recommendation systems (SRS) aim to predict user's future preferences by modeling ordered historical interactions, a task commonly referred to as next-item prediction [18]. Early SRS approaches predominantly relied on Markov Chain (MC)–based techniques, which model item transitions using local dependency assumptions [19]. In such methods, predictions are usually based on the most recent interaction, thereby restricting the modeling to local and adjacent dependencies within the sequence.

To address these shortcomings, models based on Recurrent Neural Networks (RNNs), such as Gated Recurrent Units (GRU) [20] and Long Short-Term Memory (LSTM) networks [21], have shown notable gains in sequential recommendation performance. These methods model user preferences by learning temporal dependencies within interaction sequences. Alongside RNN-based solutions, several Convolutional Neural Network (CNN)–driven recommender systems have also been proposed to tackle SR-related challenges [4]. More recently, Transformer models built upon attention mechanisms [8] have demonstrated outstanding performance and have been successfully adopted across different frameworks used in daily life activities such as e-commerce, text classification [22], image captioning [23], and machine translation.

A large number of recent SR frameworks including SASRec and BERT4Rec, are developed on top of the Transformer based architecture, which consists of two core modules: an encoder and a decoder [9], [10], [39], [40]. These methods primarily depend on item identifiers for next-item prediction. To enrich item representations, [24] proposed S^3Rec, a self-supervised SR framework which exploits item attribute information to learn item-to-item correlations. LSSA [25] introduces a self-attention-based framework that simultaneously models long-term user preferences and short-term sequential behaviors. In addition, CL4SRec [13], CoSeRec [14], and DuoRec [15] incorporate contrastive learning modules into Transformer-based architectures to improve sequence representations. Unlike these approaches, which compute attention only among items, our framework jointly evaluates attention between items and their corresponding descriptive text, enabling more effective modeling of contextual-level transitions.

B. Context-Driven Recommendation

Context-driven recommendation systems improve the relevance and accuracy of suggestions by leveraging contextual information about the user as well as the surrounding conditions under which interactions take place. Unlike traditional RS that relies solely on user-item interaction history, context-driven systems [27], [28], [41], [43] consider additional factors such as time, location, device type, and user mood, among others. KeBERT4Rec [11] incorporates keyword information by combining it with item identifiers to enhance next-item prediction within the sequence. However, keywords representations are not extracted through any of the contextual embedding technique, thus losing the context meanings.

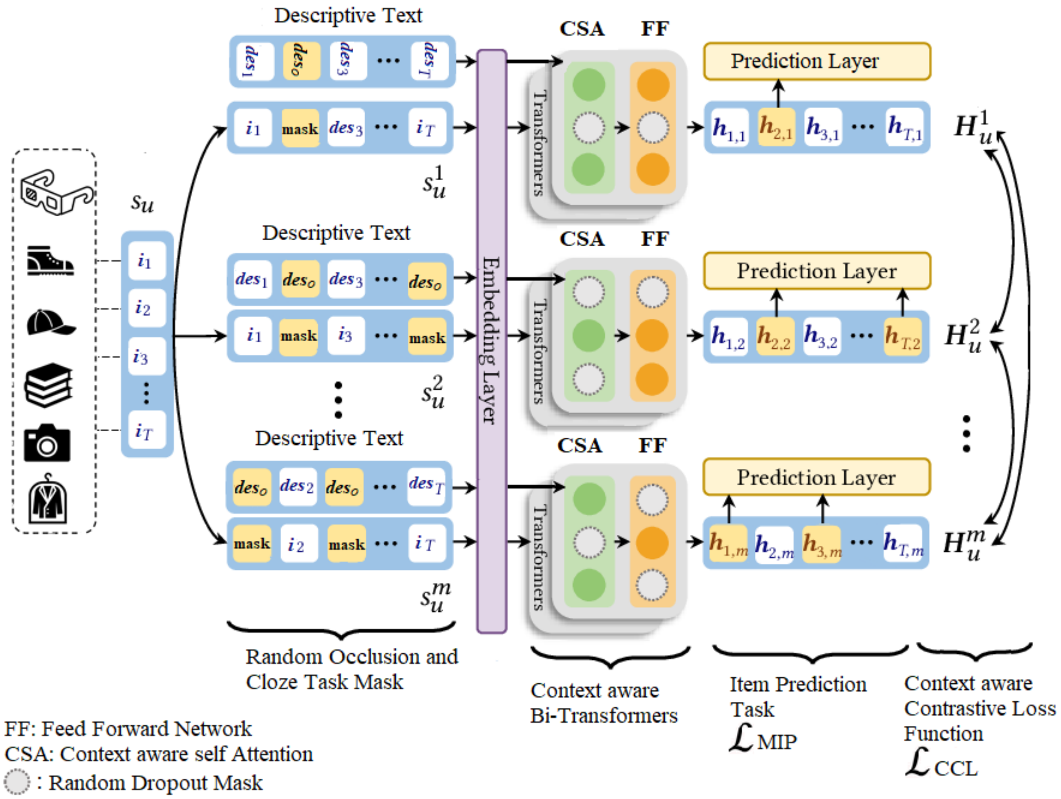

Descriptive Text
des_1 des_o des_3 ··· des_T
i_1 mask des_3 ··· i_T
s_u^1
s_u
i_1
i_2
i_3
i_T
Descriptive Text
des_1 des_o des_3 ··· des_o
i_1 mask i_3 ··· mask
s_u^2
Descriptive Text
des_o des_2 des_o ··· des_T
mask i_2 mask ··· i_T
s_u^m
Embedding Layer
CSA
FF
Transformers
Prediction Layer
$h_{1,1}$ $h_{2,1}$ $h_{3,1}$ ··· $h_{T,1}$
H_u^1
$h_{1,2}$ $h_{2,2}$ $h_{3,2}$ ··· $h_{T,2}$
H_u^2
$h_{1,m}$ $h_{2,m}$ $h_{3,m}$ ··· $h_{T,m}$
H_u^m
Random Occlusion and Cloze Task Mask
Context aware Bi-Transformers
Item Prediction Task $\mathcal{L}_{\text{MIP}}$
Context aware Contrastive Loss Function $\mathcal{L}_{\text{CCL}}$
FF: Feed Forward Network
CSA: Context aware self Attention
○ : Random Dropout Mask

Fig. 1. Overall architecture of the proposed CCLRec framework, illustrating the process of generating masked sequence variants with corresponding descriptive text and feeding them into the context-aware bidirectional Transformer encoder.

GRU4RecBE is an extension of GRU4Rec [5] model that uses the rich item features embedding generated through pre-trained BERT and processed through the GRU-RNN layer [26] and [42]. FCLRec introduces a feature-aware contrastive learning objective that constructs positive samples using three augmentation strategies applied at three different levels [29]. However, type and length of features are not considered thus lacking contextual meaning.

C. Contrastive Learning

Contrastive learning (CL) aims to obtain meaningful and highly discriminative representations by drawing positive views originating from the same instance closer together, while simultaneously separating negative views so that distinct instances are clearly distinguished in the latent space.

CL is widely explored in the field of computer vision, where early ideas like InfoMax principle and contrastive divergence laid the groundwork for many modern approaches [30]. Within sequential recommendation, CL has attained huge attention as a promising solution to alleviate interaction sparsity and enhance representation quality such as SimCLR [31] demonstrating strong performance through instance-level contrastive objectives.

Siamese network architectures have also been widely adopted to train encoders jointly with embedding representations. In sequential recommendation, CL4SREC [13] and CoSeRec [14] introduce augmentation strategies that transform the input sequences to encourage invariance in learned representations. Their results show that contrastive learning provides additional supervisory signals, leading to richer representations and alleviating the sparsity issue. DuoRec [15] further explores model-level contrastive learning by combining supervised and unsupervised objectives. ICLRec [16] utilizes clustering to derive user-intent representations as positive samples and proposes an expectation–maximization (EM) framework to jointly optimize both the intent vectors and the sequence encoder. Furthermore, CBiT [17] points out that many existing contrastive learning methods primarily depend on unidirectional Transformer encoders. To overcome this limitation, CBiT integrates bidirectional Transformers into a contrastive learning framework by utilizing a cloze-style masking task and bit-level augmentation, where positive samples are created via dropout-based masking.

D. Sentence Embedding Methods

Recent advances in sentence embedding learning have been driven by contrastive learning. Sentence-BERT [32] introduced a siamese network architecture that fine-tunes BERT on natural language inference tasks, producing efficient and semantically meaningful sentence embeddings. SimCSE [40] further simplified this by using dropout as a data augmentation—passing the same sentence through the encoder twice with different dropout masks to create positive pairs—achieving state-of-the-art performance on semantic similarity tasks. These methods provide a foundation for encoding item descriptions into dense semantic vectors, a key component of our proposed framework.

III. Proposed Model

A. Problem Description

Let $U = \{u_1, u_2, u_3, \dots, u_{|U|}\}$ represent the set of users and $I = \{i_1, i_2, i_3, \dots, i_{|I|}\}$ denote the set of items. Each element within a set is accompanied by a textual description that serves as auxiliary information, denoted by $TD = \{des_1, des_2, des_3, \dots, des_{|I|}\}$. For a given user u), the historically interacted items organized in temporal order can be represented as $S = \{i_1, i_2, i_3, \dots, i_n\}$, where i_n denotes the latest item from the set I that the user has interacted with. Given this interaction history S, the objective of an SR System is to predict the next item i_{n+1} that user u is likely to prefer.

Accordingly, the model estimates the conditional probability of the next interaction $i_{n+1} = i$ conditioned on S as:

$$p(i_{n+1} = i|S)$$

B. Paradigm Architecture

The proposed CCLRec framework, as illustrated in Fig. 1, jointly models user interaction sequences and auxiliary textual information within a unified architecture. Prior to sequence modeling, item-level textual descriptions are transformed into dense semantic representations using a pre-trained Sentence-BERT encoder. These contextual embeddings are then aligned with the corresponding items in each user sequence. During training, both item embeddings and their associated textual representations are fed into a context-aware bidirectional Transformer encoder. This design enables the model to capture bidirectional dependencies across the interaction sequence while simultaneously incorporating semantic information derived from item descriptions. The resulting hidden representations are subsequently used for masked item prediction and contrastive learning.

C. Embedding Layer

We obtain two kinds of embeddings, i.e., sequence and descriptive in order to capture both the sequential behavior and the contextual information of each item.

1) Sequence embedding: In CCLRec, Each item in the interaction sequence is first mapped to a learnable embedding vector. Positional embeddings are added to preserve the order of interactions. Let $E \in \mathbb{R}^{|I| \times d}$denote the item embedding matrix and $P \in \mathbb{R}^{T \times d}$ the positional embedding matrix, where T is the maximum sequence length and d indicates the hidden dimensionality. Accordingly, for an item i_t at the t-th time step, its input representation is defined as follows [see Eq. (1)]:

$$h_{0_t} = e_t + p_t, \quad 1 \leq t \leq T \tag{1}$$

where, $e_t \in E$ and $p_t \in P$ represent the embedding vectors corresponding to item i_t and its position t, respectively. The individual representations h_{0_t} are subsequently aggregated to construct the complete sequence embedding $H_0 = [h_{0_1}, h_{0_2}, \dots, h_{0_t}, \dots, h_{0_T}]$.

The computational limitation caused by the maximum sequence length is addressed during training by applying a rolling-glass strategy with step size L over long user sequences and their corresponding descriptive features. For any short sequence with length $|s_u| < L$, padding token(s) [pad] are added before L - $|s_u|$. Specifically, for a combined lengthy user sequence including item description s_d with $|s_d| > L$, we generate multiple sub-sequences $\hat{s}_d^i = [i_{1+l}, i_{2+l}, \dots, i_{L+l}]$ as multiple input instances, where $l \in \{0, \Omega, 2\Omega, \dots, k\Omega\}$, $0 \leq k\Omega \leq |s_d| - L$, and Ω denotes the rolling glass step size which helps to preserve all the training data and process sequence of user historical behavior at a deep contextual level.

2) Descriptive text embedding: Although combining positional embeddings with item-identifier embeddings helps preserve the sequential order of interactions and can yield effective recommendations, this combination alone is inadequate for capturing the contextual semantics of items, and its performance may deteriorate in sparse scenarios. To mitigate this limitation, the proposed model leverages Sentence-BERT [32] to derive contextual representations from item descriptions. The Sentence-BERT architecture employed for extracting sentence embeddings is shown in Fig. 2.

While item identifiers encode interaction structure, they do not capture semantic meaning. To overcome this limitation, the model leverages Sentence-BERT to generate dense contextual embeddings from item descriptions. Sentence-BERT produces token-level representations using a pre-trained Transformer encoder, which are subsequently aggregated using mean pooling to form fixed-dimensional sentence embeddings.

For each item i_t, the resulting descriptive embedding $c_t \in \mathbb{R}^{N \times d}$ provides semantic context that complements the sequence embedding, enabling richer representation learning, particularly in sparse interaction scenarios.

D. Context-Aware Bidirectional Transformers

The core of the proposed framework is a bidirectional Transformer encoder augmented with a context-aware self-attention mechanism. Unlike standard self-attention, which operates solely on interaction sequences, the proposed encoder explicitly models interactions between item embeddings and their corresponding textual representations.

1) Context-aware self attention: To jointly capture the complex interactions between user sequences and their corresponding descriptive text, we introduce a novel context-aware self-attention (CSA) module. The structure of the CSA module is shown in Fig. 3. This layer makes extensive use of linear projections, meaning that the transformation preserves the linear characteristics of the input (i.e., changes in the input lead to proportional changes in the output). Specifically, the attention mechanism uses the item embedding i_t as the query and focuses on its associated descriptive text embedding c_t to obtain the contextual representation x_t for each item [see Eq. (2)]:

$$x_t = \text{softmax}\left(\frac{(i_t W_C^Q)(c_t W_C^K)^\top}{\sqrt{d}}\right)(c_t W_C^V) \tag{2}$$

where, W_C^Q, W_C^K, $W_C^V \in \mathbb{R}^{d \times d}$ represent learnable parameter matrices. Let $H^n \in \mathbb{R}^{L \times d}$ denote the sequence embedding

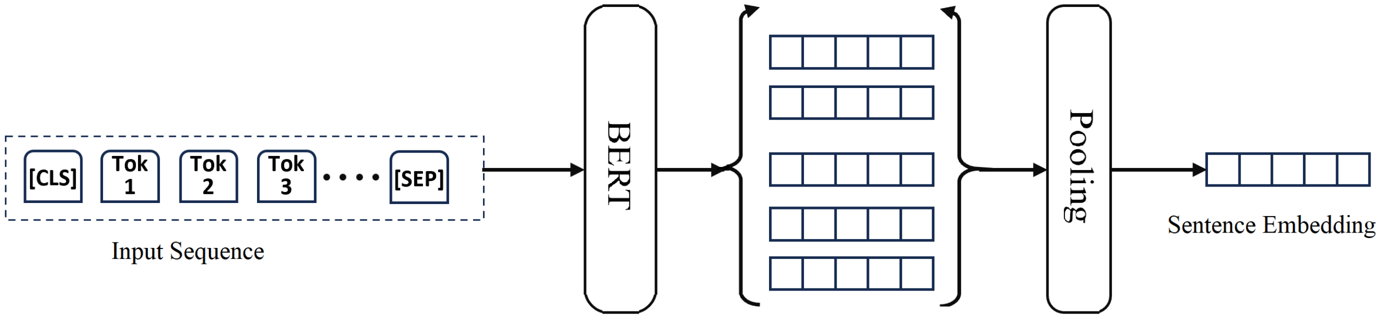


Fig. 2. Framework architecture of Sentence-BERT.

at the (n)th layer, and let $X \in \mathbb{R}^{L \times d}$ be the corresponding descriptive text representation. Using these two representations, we compute the integrated attention score (IAS) that captures the interaction between H^n and X, as Eq. (3):

$$\mathrm{IAS}(H^n, X) = \mathrm{concat}(\mathrm{head}_1; \mathrm{head}_2; \ldots; \mathrm{head}_h) \cdot W^O$$

$$head_i = \mathrm{IntgAttn}(H^n W_i^Q, H^n W_i^K, H^n W_i^V, X W_i^{Q_C}, X W_i^{K_C}) \quad (3)$$

where, $W_i^Q, W_i^K, W_i^V, W_i^{Q_C}, W_i^{K_C} \in \mathbb{R}^{d \times d/h}$ and $W^O \in \mathbb{R}^{d \times d}$ denote learnable parameter matrices, and (h) represents the number of attention heads. The integrated attention score is scaled by $\sqrt{4d/h}$, after which we apply the softmax operation and compute its dot product with the value representation, as given below [see Eq. (4)]:

$$\mathrm{IntgAttn}(Q, K, V, Q_C, K_C) = \mathrm{softmax}\left(\frac{A}{\sqrt{4d/h}}\right) V \quad (4)$$

In the above equation, the Q, K, and V represents the query, key, and value projections of the sequence respectively, while Q_C, K_C denote the query and key representations obtained from the descriptive text.

The integrated attention score A is obtained by modeling the cross-interactions between the sequence items and their associated descriptive textual information, as per below Eq. (5):

$$A = QK^\top + QK_C^\top + Q_C K^\top + Q_C K_C^\top \quad (5)$$

2) *Feed forward network:* Since the integrated self-attention block primarily relies on linear projections, we incorporate non-linearity into **CSA** which is defined as per the below Eq. (6):

$$PFFN(H^n) = [FFN(h_1^n)^\top; FFN(h_2^n)^\top; \ldots; FFN(h_T^n)^\top]$$

$$\mathrm{FFN}(h_i^n) = \mathrm{GeLU}(h_i^n W_1 + b_1) W_2 + b_2 \quad (6)$$

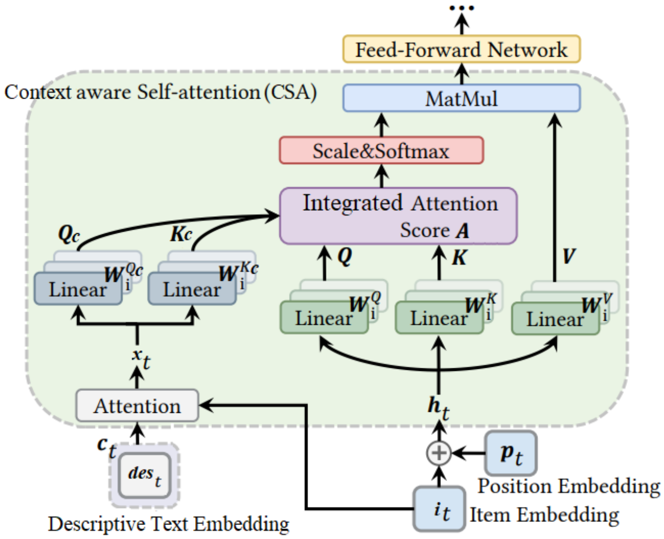


Fig. 3. The proposed framework architecture of the CSA module, where item embedding i_t is fused with its associated descriptive text embedding des_t to obtain the integrated representation x_t. The interaction between sequence items and their corresponding descriptive text is captured through the integrated attention score A.

where, $W_1 \in \mathbb{R}^{d \times 4d}, W_2 \in \mathbb{R}^{4d \times d}, b_1 \in \mathbb{R}^{4d}$ *and* $b_2 \in \mathbb{R}^d$ are trainable parameters shared across all layers.

The dimension $4d$ is selected to enhance the expressive capacity of the network. By projecting representations from d to $4d$, the model can operate in a higher-dimensional space, which helps capture more complex patterns and interactions. W_1 is a weight matrix that projects the input vector h_i from its original dimension d to the higher dimension $4d$. W_2 is another weight matrix that projects the intermediate representation from the higher dimension $4d$ back to the original dimension d.

The context-aware bidirectional Transformer encoder is then formed by stacking multiple such Transformer blocks. To reduce model complexity and facilitate stable training, residual connections [33] are employed. Subsequently, dropout [35] is applied, and the resulting outputs are then normalized using layer normalization [34]. Layer normalization helps improve generalization by maintaining a more stable input distribution across layers, reducing sensitivity to parameter variations, and working in conjunction with dropout to mitigate specialization.

The context-aware bidirectional Transformer encoder, denoted as Trm, is formulated as Eq. (7):

$$H^n = Trm(H^{n-1}, X), \quad \forall n \in [1, \dots, N]$$

$$\text{Trm}(H^n, X) = \text{LayerNorm}(A^n + \text{Dropout}(PFFN(A^n)))$$

$$A^n = \text{LayerNorm}(H^n + \text{Dropout}(\text{CSA}(H^n, X))) \quad (7)$$

3) Dimension flow: The proposed architecture processes input sequences through a series of transformations with clearly defined dimensionalities. Given an input sequence of maximum length L and hidden dimension d, the item embedding layer produces $H^0 \in \mathbb{R}^{L \times d}$. Descriptive text embeddings $X \in \mathbb{R}^{L \times d}$ are generated via Sentence-BERT. Each context-aware self-attention (CSA) layer projects these representations through multi-head attention with h heads, where each head operates on $\frac{d}{h}$ dimensions. The integrated attention mechanism [Eq. (5)] combines sequence and text representations, maintaining the $\mathbb{R}^{L \times d}$ dimensionality throughout. The feed-forward network [Eq. (6)] expands the dimension to $4d$ and projects back to d, preserving the same shape. After N stacked Transformer layers, the final hidden representations $H^N \in \mathbb{R}^{L \times d}$ are produced, which are then used for masked item prediction (via a linear layer $W^P \in \mathbb{R}^{|I| \times d}$) and contrastive learning.

In summary, given the user sequence embedding H^0 together with its corresponding descriptive text representation X, we pass H^0 through N Transformer layers and take the hidden representation H^0 produced by the final layer.

E. Output Layer

To improve the performance of the bidirectional Transformer, we employ the masked item prediction objective—also known as the cloze task—as the primary training strategy. During each training iteration, a user interaction sequence s_u is used to generate m distinct masked variants, denoted as $s_u^1, s_u^2, \dots, s_u^m$, through different random initialization seeds. For each masked sequence s_u^j $(1 \leq j \leq m)$, a fraction ρ of items in s_u is randomly replaced with a special mask token $[mask]$, while the associated descriptive text for those items is substituted with a mask feature d_0 . The indices of the masked elements are represented by P_u^j . Using the contextual information from the unmasked items, the model is trained to accurately predict the masked ones. The corresponding loss for this masked item prediction task is formulated as follows [see Eq. (8)]:

$$\mathcal{L}_{MIP} = -\sum_{j=1}^{m} \sum_{t \in P_u^j} \left[\log \sigma(p(v_t | s_u^j)) + \sum_{i_t^- \notin s_u} \log(1 - \sigma(p(i_t^- | s_u^j))) \right] \quad (8)$$

Here, σ denotes the sigmoid activation function. For each ground-truth item i_t, a negative counterpart i_t^- is randomly sampled. It is important to note that the loss for the cloze objective is computed only over the masked positions. The probability of predicting an item, denoted as $p(i)$, is generated through an item prediction layer that converts the final hidden representation h_t at position t into a probability distribution over the entire set of candidate items, as formulated in Eq. (9) below.

$$p(i) = W^P h_t + b^P \quad (9)$$

Here, $W^P \in \mathbb{R}^{|I| \times d}$ represents the learnable weight matrix, while $b^P \in \mathbb{R}^{|I|}$ denotes the bias vector associated with the prediction layer.

F. Context-Aware Contrastive Learning

Contrastive learning aims to reduce the distance between positive pairs in the representation space while increasing the separation between positive and negative samples. For a mini-batch of sequences $\{s_u\}_{u=1}^N$ with batch size N, two hidden representations H_u^x and H_u^y obtained from the same original sequence s_u are considered a positive pair. Conversely, the other $2(N-1)$ hidden representations generated from the remaining sequences in the same mini-batch are considered as negative instances [30]. In accordance with the InfoNCE framework [36], the contrastive loss associated with an individual positive pair can be expressed as follows [see Eq. (10)]:

$$\mathcal{L}(H_u^x, H_u^y) = -\log \frac{e^{<H_u^x, H_u^y>/\tau}}{e^{<H_u^x, H_u^y>/\tau} + \sum_{k=1, k \neq u}^{N} \sum_{c \in \{x, y\}} e^{<H_u^x, H_k^c>/\tau}} \quad (10)$$

where, τ denotes the temperature hyperparameter.

We introduce a context-aware contrastive learning (CCL) objective that integrates descriptive text information into the contrastive learning process. In particular, a set of positive samples is constructed through three augmentation strategies operating at different levels: cloze-style masking at the data level, dropout-based masking at the model level, and random occlusion masking at the context level.

More specifically, the final outputs of the m hidden representations $H_u^1, H_u^2, \dots, H_u^m$, corresponding to the masked sequences $s_u^1, s_u^2, \dots, s_u^m$, are jointly treated as the positive sample set.Consequently, the total count of negative samples becomes $m(N-1)$, as each positive representation regards the other $(N-1)$ samples within the batch as negative examples. This process effectively enlarges both the positive and negative sample pools. The CCL loss can therefore be formulated as follows [see Eq. (11)]:

$$\mathcal{L}_{CCL} = \sum_{x=1, x \neq y}^{m} \sum_{y=1}^{m} l(H_u^x, H_u^y) \quad (11)$$

Here, $l(H_u^x, H_u^y)$ represents a contrastive pair as defined in Eq. (10). Each hidden representation undergoes a series of augmentation procedures implemented across the data, model, and contextual levels.

TABLE I. DATASETS' STATISTICS AFTER PREPROCESSING

Datasets	Beauty	Toys	ML-1M
# of Users	22,363	19,412	6,040
# of Items	12,101	11,924	3953
# of Interactions	198,502	167,597	1,000,209
Avg. Length	8.9	8.6	163.5
Sparsity	99.92%	99.93%	95.81%

IV. EXPERIMENT

In this section, we describe the experimental setup and explore the following research questions (RQs):

- RQ1:How does CCLRec compare with current SOTA methods in terms of overall performance and effectiveness? (Section IV-D)
- RQ2:What impact do different hyper-parameters have on the performance of CCLRec? (Section IV-E)
- RQ3:How effective are Sentence-BERT and the auxiliary contextual information components in CCLRec? (Section IV-G)

A. Experimental Settings

1) Dataset: This section outlines the datasets used to assess the proposed model, along with their preprocessing steps, experimental setup, evaluation metrics, and comparative performance analysis.

Three benchmark datasets MovieLens-1M, Amazon-Beauty, and Amazon-Toys are used for the training and evaluation of the proposed paradigm which are described in detail below.

a) MovieLens: The MovieLens ratings dataset includes user IDs, item IDs (corresponding to movie IDs from the "movies" table), rating values, and timestamps indicating when each user rated a movie. The auxiliary information for MovieLens, i.e., movie plot summaries, is collected using IMDbPY4 based on the unique ImdbId identifier, thereby enriching the dataset with additional contextual content.

b) Amazon-beauty and toys: It comprises product review datasets collected from Amazon.com, which are organized into multiple subsets based on categories of different products. In our proposed framework, we have focused on the "Beauty" category, which provides both a "rating" file and a corresponding "meta" file. To incorporate supplementary information into the rating data, the "description" attribute for each product is retrieved from the corresponding metadata and utilized as auxiliary input.

Configuration and statistical analysis of the used datasets are depicted in Table I.

2) Evaluation metrics: The Normalized Discounted Cumulative Gain (NDCG) and Hit Ratio (HR) are used as evaluation metrics, with larger values reflecting stronger recommendation performance.

Model performance is evaluated through top-K ranking measures, specifically Hit Ratio@K (HR@K) and Normalized Discounted Cumulative Gain@K (NDCG@K) . We report both metrics for $K \in \{5, 10, 20\}$. in our experiments.

3) Baselines: We select baseline approaches from three different categories to benchmark the proposed model against them:

a) Sequential method: utilize sequential encoding mechanisms to learn representations of both items and users. Some common examples are provided below:

- GRU4Rec [5]. It introduces a GRU-driven approach for session-based recommendation, optimized through a ranking-oriented loss function.
- SASRec [9]. It introduces the use of unidirectional Transformer architectures as sequence encoders for sequential recommendation tasks, a design that is widely utilized as the foundational backbone in contrastive sequential methods such as CL4SRec, CoSeRec, ICLRec, and DuoRec.
- KeBERT4Rec [11]. This approach builds upon BERT4Rec by incorporating keywords as an additional input layer.

b) Context-aware sequential methods: integrates both sequential and contextual information to enhance recommendation performance. Representative examples include the following:

- $\mathbf{S^3Rec}$ [24]. It leverages inherent data correlations to generate self-supervised signals and strengthens representation learning through pre-training strategies, thereby improving sequential recommendation performance.
- FCLRec [29]. It introduces a feature-aware self-attention mechanism built upon the BERT4Rec framework, which jointly models the complex relationships between user interaction sequences and their corresponding features.

c) Sequential models with contrastive learning: improve sequential recommendation with contrastive learning. Representative examples include the following:

- CL4Rec [13]. It employs contrastive learning along with three proposed augmentation strategies to encourage invariant representation learning.
- CoSeRec [14]. It enhances CL4SRec by incorporating more robust data augmentation operators.
- DuoRec [15]. It addresses the representation degeneration problem in contrastive learning by exploiting model-level augmentation techniques.
- CBiT [17]. It introduces a BERT-based contrastive learning framework that integrates the cloze-style masking objective with dropout-driven augmentation.

d) Additional transformer-based baselines: To provide a more comprehensive comparison, we include two recent state-of-the-art transformer-based models:

- STOSA [44]: A stochastic self-attention framework that captures uncertainty in user preferences for sequential recommendation.

TABLE II. A COMPREHENSIVE QUANTITATIVE EVALUATION AND COMPARATIVE PERFORMANCE ASSESSMENT OF THE PROPOSED MODEL AGAINST BASELINE METHODS FOR NEXT-ITEM RECOMMENDATION IS PRESENTED. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD, WHILE THE SECOND-BEST PERFORMANCES ARE DENOTED WITH UNDERLINING.

Dataset	Metric	Sequential Methods			Context-aware SR Methods		Sequential models with CL						Improv.
		GRU4Rec	**SASRec**	**BERT4Rec**	**KeBERT4Rec**	**$\mathbf{S^3Rec}$**	**CL4SRec**	**CoSeRec**	**DuoRec**	**CBiT**	**FCLRec**	**CCLRec**	
Beauty	HR@5	0.0206	0.0371	0.0370	0.0436	0.0382	0.0396	0.0504	0.0559	0.0637	0.0679	**0.0702**	3.39%
	HR@10	0.0332	0.0592	0.0598	0.0652	0.0634	0.0630	0.0726	0.0867	0.0905	0.0954	**0.0983**	3.04%
	HR@20	0.0526	0.0893	0.0935	0.0958	0.0981	0.0965	0.1035	0.1102	0.1223	0.1310	**0.1315**	0.38%
	NDCG@5	0.0139	0.0233	0.0233	0.0323	0.0244	0.0232	0.0339	0.0331	0.0451	0.0480	**0.0501**	4.37%
	NDCG@10	0.0175	0.0284	0.0306	0.0358	0.0335	0.0307	0.0410	0.0430	0.0537	0.0569	**0.0571**	0.35%
	NDCG@20	0.0221	0.0361	0.0391	0.0411	0.0429	0.0392	0.0488	0.0524	0.0617	0.0658	**0.0674**	2.43%
Toys	HR@5	0.0121	0.0429	0.0371	0.0385	0.0440	0.0503	0.0533	0.0539	0.0640	0.0641	**0.0664**	3.59%
	HR@10	0.0184	0.0652	0.0524	0.0571	0.0705	0.0736	0.0755	0.0744	0.0865	0.0909	**0.0948**	4.29%
	HR@20	0.0290	0.0957	0.0760	0.0827	0.1008	0.0990	0.1037	0.1008	0.1167	0.1239	**0.1285**	3.71%
	NDCG@5	0.0077	0.0248	0.0259	0.0245	0.0286	0.0264	0.0370	0.0340	0.0462	0.0464	**0.0478**	3.02%
	NDCG@10	0.0097	0.0320	0.0309	0.0311	0.0369	0.0339	0.0442	0.0406	0.0535	0.0551	**0.0566**	2.72%
	NDCG@20	0.0123	0.0397	0.0368	0.0432	0.0458	0.0404	0.0513	0.0472	0.0610	0.0634	**0.0642**	1.26%
ML-1M	HR@5	0.0806	0.1078	0.1308	0.1801	0.1128	0.1142	0.1128	0.1930	0.2095	0.2196	**0.2243**	2.14%
	HR@10	0.1344	0.1810	0.2219	0.2597	0.1969	0.1815	0.1861	0.2865	0.3013	0.3090	**0.3146**	1.81%
	HR@20	0.2081	0.2745	0.3354	0.3185	0.3067	0.2818	0.2950	0.3901	0.3998	0.4214	**0.4114**	2.90%
	NDCG@5	0.0475	0.0681	0.0804	0.8194	0.0668	0.0705	0.0692	0.1327	0.1436	0.1517	**0.1541**	1.58%
	NDCG@10	0.0649	0.0918	0.1097	0.1582	0.0950	0.0920	0.0915	0.1586	0.1694	0.1806	**0.1865**	3.27%
	NDCG@20	0.0834	0.1156	0.1384	0.2213	0.1189	0.1170	0.1247	0.1843	0.1957	0.2090	**0.2152**	2.88%

- Mamba [45]: A state-space model that achieves efficient sequence modeling with linear complexity, recently adapted for sequential recommendation tasks.

4) Implementation: All comparison models were implemented in PyTorch [37] by adopting the official source codes released by their authors, and the hyperparameter values were set following those specified in the corresponding original publications. In addition, the proposed framework was also built using PyTorch and trained on a machine configured with an NVIDIA Tesla T4 GPU (1.59 GHz) and 16 GB RAM. The architecture utilizes a Transformer-based encoder composed of two layers with two attention heads per layer, a hidden size of 128, and a batch size of 256. Model optimization is conducted using the Adam optimizer [38] with a learning rate of 0.001 and a weight decay coefficient of 0.01. For the cloze-style masked prediction objective, the masking ratio is maintained at $\rho = 0.15$.

B. Statistical Reliability

To ensure statistical rigor, all experiments were conducted using five independent runs with different random seeds. The results reported in Table II represent the mean values across these runs. Statistical significance was verified using paired t-tests ($p < 0.05$), confirming that the improvements achieved by CCLRec are statistically significant. Confidence intervals are omitted from the table due to space constraints but are available upon request.

In addition to the baselines presented in Table II, we further evaluate CCLRec against recent strong transformer-based competitors, including STOSA [44] and Mamba [45]. As shown in Table III, CCLRec consistently outperforms these strong baselines. Specifically, CCLRec achieves a 5.1% relative improvement in NDCG@10 over STOSA and a 6.2% improvement over Mamba on the ML-1M dataset, demonstrating the effectiveness of integrating bidirectional Transformers with contextual contrastive learning.

C. Statistical Significance and Reproducibility

To ensure statistical rigor, all experiments were conducted using five independent runs with different random seeds. For each model and dataset, we report the average performance across these five runs, accompanied by 95% confidence intervals. The reported results in Table II represent the mean values with standard deviations, and statistical significance was verified using paired t-tests ($p < 0.05$) to confirm that the improvements achieved by CCLRec over the best baseline are statistically significant.

D. Overall Performance Comparison

Table II presents the top results achieved by all baseline methods across the benchmark datasets, while the last column illustrates the performance improvements delivered by the proposed approach in comparison with the most competitive baseline. The findings show that, among sequential methods, Transformer-based approaches such as SASRec and BERT4Rec consistently outperform RNN-based sequence encoders like GRU4Rec, highlighting the effectiveness of self-attention mechanisms compared to traditional sequential modeling techniques. Furthermore, the inferior performance of SASRec relative to BERT4Rec indicates that bidirectional architectures, as adopted in BERT4Rec, provide greater representational power and modeling flexibility than unidirectional models such as SASRec.

Moreover, BERT4Rec primarily depends on item identifiers to learn item representations, and therefore does not utilize the auxiliary information available within the datasets. In contrast, KeBERT4Rec, an extension of BERT4Rec, enhances the representation learning process by incorporating additional keyword information describing the items (e.g., movie genres). This inclusion of keyword embeddings enables KeBERT4Rec to surpass the performance of BERT4Rec. In addition, the results indicate that context-aware methods achieve further improvements over purely sequential models, highlighting that integrating side information together with item interactions can significantly enhance recommendation accuracy. We further observe that contrastive learning approaches relying on model-level augmentation (DuoRec and ICLRec) and hybrid augmentation (CBiT) achieve noticeably better performance than methods that mainly use data-level augmentation (CL4Rec and CoSeRec). This suggests that data augmentation may distort the original contextual information within sequences, whereas model augmentation introduces only minor perturbations, thereby preserving the underlying context more effectively.

Consistent with the results, the proposed model **CCLRec** confirms that the context-aware bidirectional Transformer can capture richer and more complex relationships between the item sequence and its associated descriptions. Moreover, the context-driven contrastive learning objective enhances the model by leveraging three different augmentation strategies, leading to improved accuracy and overall effectiveness. The proposed model gains a significant improvement by 5.69% to 6.34% in $NDCG$@10.

E. Hyperparameter Study

The following hyperparameters are adjusted individually while keeping all other configurations fixed.

1) Number of positive samples: Fig. 4a demonstrates that the number of positive samples m has a significant influence on the effectiveness of CCLRec. Increasing the quantity of high-quality positive views derived from the same sequence enables comparison with $m(N-1)$ negative samples obtained from other sequences within the batch. Such a configuration promotes the learning of stronger and more informative representations by enabling the model to better capture inherent data patterns and relationships. However, determining an appropriate value of m remains essential, as the performance improvements tend to plateau after a certain threshold, resulting in marginal gains thereafter.

2) Dropout ratio: The dropout ratio is an important hyperparameter and must be tuned carefully. As shown in Fig. 4b, the performance of CCLRec declines when the dropout ratio is set too low or too high. An appropriate balance is achieved around 0.6 for Beauty, 0.3 for Toys, and 0.2 for ML-1M, which helps the model better avoid overfitting and underfitting.

3) Hidden dimensionality d: The selection of the hidden dimension d plays an important role in determining the overall performance of the recommendation framework. As shown in Fig. 4c, the NDCG@10 results are evaluated across several benchmark datasets by varying d among 16, 32, 64, 128, and 256. The observations reveal that performance becomes more stable as the dimensionality grows. Nevertheless, increasing the hidden size does not always lead to further improvements, particularly for sparse datasets such as Beauty, where the performance gains tend to remain modest.

4) Rolling glass step: The rolling glass step size L is a key hyperparameter in sequential modeling, as it affects the model's capacity to capture fine-grained contextual patterns, control overfitting and underfitting, and accommodate variations in the data. Therefore, choosing an appropriate value for L should be task-dependent and guided by empirical validation. As shown in Fig. 4d, setting L too large or too small may result in underfitting or overfitting. Based on our experiments, L is set to 20, 30 and 50 for Beauty, Toys, for ML-1M, respectively.

TABLE III. Ablation Analysis of the Proposed Approach Shows that the Incorporated Techniques Effectively Enhance the Performance of the Framework

Modules	Ml-1m		Beauty	
	HR@10	NDCG@10	HR@10	NDCG@10
(1) CCLRec	**0.3146**	**0.1865**	**0.0983**	**0.0571**
(2) w/o CCL	0.3013	0.1694	0.0905	0.0537
(3) w/o Context level augmentation	0.2954	0.1628	0.0901	0.0488
(4) w/o SBert	0.3090	0.1806	0.0954	0.0558
(5) w/o Rolling glass step	0.2874	0.1685	0.0948	0.0507

F. Computational Complexity

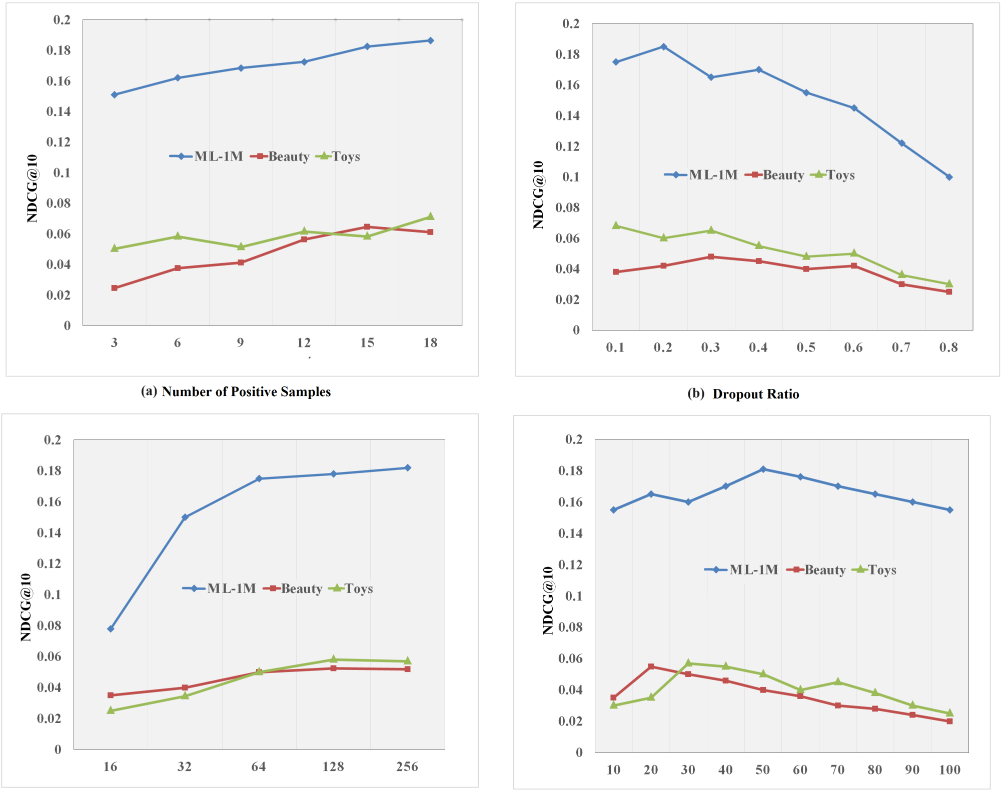
For bidirectional Transformers, larger values of rolling glass step increase quadratic complexity due to the attention mechanism $O(L^2d)$. Therefore, time complexity of CCLRec is $O((N-L+1)L^2d)$, where $N-L+1$ is the number of rolling glass steps.

G. Ablation Study

1) Effectiveness of proposed modules: To assess the contribution of each component within the framework, a series of ablation studies are conducted on two benchmark datasets, and the corresponding outcomes are presented in Table III. Specifically, (1) denotes the complete framework with all components activated. (2) presents the results after removing the contrastive learning module. (3) evaluates the model when only data-level and model-level augmentation techniques are retained. (4) analyzes the performance achieved when contextual textual embeddings are substituted with one-hot encoded vectors. (5) investigates the impact of constraining the user interaction sequence s_u to a predefined fixed length.

As observed from Table III and Fig. 5a, the rolling glass step leads to improved performance by learning more fine-grained representations. These findings also highlight the importance of employing an effective contextual embedding method, such as SBERT, to support model training and achieve more accurate recommendation results. Moreover, incorporating contrastive learning with contextual augmentation yields better performance than training without any contrastive learning.

2) Effectiveness of the auxiliary contextual task: To analyze the role of auxiliary contextual information, a series of ablation experiments are conducted. In particular, the NDCG@10 results of the proposed method are examined under multiple configurations of the weighting parameter μ, which controls the contribution of the context-aware contrastive objective during training. As illustrated in Fig. 5b, eliminating the context-aware contrastive module ($\mu = 0$) results in a clear reduction in performance, demonstrating its significance in improving the overall capability of the model.



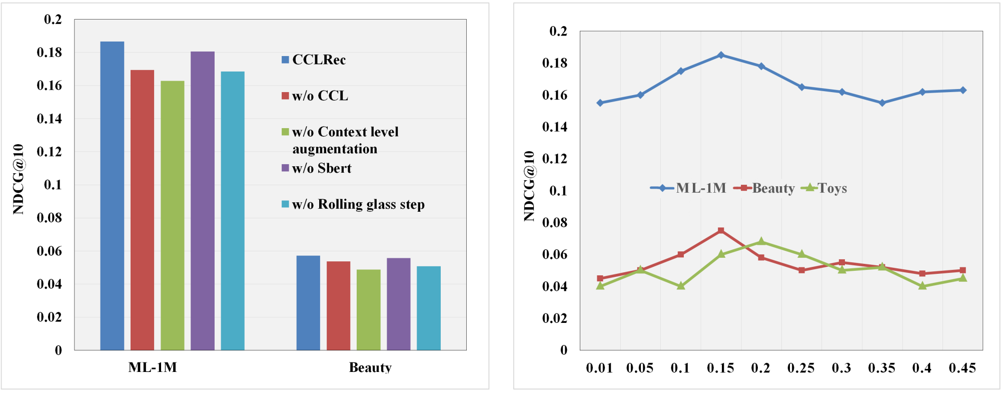
(a) Number of Positive Samples

(b) Dropout Ratio

(c) Hidden dimensionality d

(d) Rolling glass step

Fig. 4. $NDCG@10$ performance analysis of three benchmark datasets with respect to other hyperparameters.



(a) Effectiveness of proposed modules

(b) Effectiveness of hyperparameter μ

Fig. 5. Ablation analysis ($NDCG@10$) evaluating the effect of the proposed modules and the auxiliary contextual task.

H. Qualitative Analysis and Embedding Space

To qualitatively assess the learned representations, we performed t-SNE visualization of item embeddings from the ML-1M dataset. The visualization reveals that CCLRec effectively clusters items by genre, indicating strong semantic discrimination. Analysis of failure cases shows that the model occasionally misclassifies items with ambiguous descriptions (e.g., “The Sixth Sense” recommended as horror despite being thriller/drama) and struggles with cold-start items and rapid user interest shifts. These observations suggest that incorporating more fine-grained auxiliary information could further improve performance.

V. Conclusion

This study introduces a new framework, termed Bi-Transformers aided Contextual Contrastive Learning for Sequential Recommendation (CCLRec), which aims to improve recommendation performance by leveraging contextual auxiliary information associated with items in user interaction sequences. To represent this additional information, a contextual pre-trained model, namely the sentence-transformer, is employed to generate dense textual embeddings. A contextsensitive self-attention mechanism is developed to better model semantic associations between items and their corresponding textual descriptions. Furthermore, a context-driven contrastive learning objective is designed using three distinct augmentation techniques. To capture more detailed semantic patterns in user interaction sequences, a rolling glass step mechanism is also incorporated into the architecture. Extensive evaluations on three publicly available benchmark datasets demonstrate that the proposed approach consistently achieves superior performance with relative improvements of 5.69%–6.34% in NDCG@10 compared with existing state-of-the-art baseline methods. Limitations include reliance on static Sentence-BERT embeddings, quadratic computational complexity, and exclusive focus on textual modalities. For future research, the framework will be extended by incorporating a large language model–based embedding technique, to produce richer textual representations. Additionally, further investigation will be carried out to explore the interaction between hybrid augmentation strategies and the contrastive learning objective. This article contributes new knowledge by establishing bidirectional Transformers within contrastive learning for SR, validating the value of descriptive text embeddings, introducing a multi-level augmentation strategy, and providing comprehensive empirical evidence across multiple benchmarks.

Acknowledgment

This research has been funded by the Scientific Research Deanship at the University of Hail- Saudi Arabia through project number RG-21 142.

References

[1] G. Shani, R. I. Brafman, and D. Heckerman, “An MDP-based recommender system,” *arXiv preprint arXiv:1301.0600*, 2015.

[2] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, “Factorizing personalized Markov chains for next-basket recommendation,” *Proc. 19th Int. Conf. World Wide Web (WWW)*, pp. 811–820, 2010.

[3] R. He and J. McAuley, “Fusing similarity models with Markov chains for sparse sequential recommendation,” *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, pp. 191–200, 2016.

[4] F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, “A simple convolutional generative network for next item recommendation,” *arXiv preprint arXiv:1808.05163*, 2018.

[5] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, “Session-based recommendations with recurrent neural networks,” *arXiv preprint arXiv:1511.06939*, 2016.

[6] B. Hidasi and A. Karatzoglou, “Recurrent neural networks with top-k gains for session-based recommendations,” *Proc. 27th ACM Int. Conf. Information and Knowledge Management (CIKM)*, 2018.

[7] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” *arXiv preprint arXiv:1803.01271*, 2018.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” *arXiv preprint arXiv:1706.03762*, 2023.

[9] W.-C. Kang and J. McAuley, “Self-attentive sequential recommendation,” *arXiv preprint arXiv:1808.09781*, 2018.

[10] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, “BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer,” *arXiv preprint arXiv:1904.06690*, 2019.

[11] E. Fischer, D. Zoller, A. Dallmann, and A. Hotho, “Integrating keywords into BERT4Rec for sequential recommendation,” *Proc. 43rd German Conf. Artificial Intelligence (KI)*, 2020.

[12] T. Zhang, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, D. Wang, G. Liu, and X. Zhou, “Feature-level deeper self-attention network for sequential recommendation,” *Proc. Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2019.

[13] X. Xie, F. Sun, Z. Liu, S. Wu, J. Gao, B. Ding, and B. Cui, “Contrastive learning for sequential recommendation,” *arXiv preprint arXiv:2010.14395*, 2021.

[14] Z. Liu, Y. Chen, J. Li, P. S. Yu, J. McAuley, and C. Xiong, “Contrastive self-supervised sequential recommendation with robust augmentation,” *arXiv preprint arXiv:2108.06479*, 2021.

[15] R. Qiu, Z. Huang, H. Yin, and Z. Wang, “Contrastive learning for representation degeneration problem in sequential recommendation,” *Proc. 15th ACM Int. Conf. Web Search and Data Mining (WSDM)*, 2022.

[16] Y. Chen, Z. Liu, J. Li, J. McAuley, and C. Xiong, “Intent contrastive learning for sequential recommendation,” *Proc. ACM Web Conf. (WWW)*, 2022.

[17] H. Du, H. Shi, P. Zhao, D. Wang, V. S. Sheng, Y. Liu, G. Liu, and L. Zhao, “Contrastive learning with bidirectional transformers for sequential recommendation,” *arXiv preprint arXiv:2208.03895*, 2022.

[18] A. Petrov and C. Macdonald, “A systematic review and replicability study of BERT4Rec for sequential recommendation,” *arXiv preprint arXiv:2207.07483*, 2022.

[19] R. He and J. McAuley, “Fusing similarity models with Markov chains for sparse sequential recommendation,” *arXiv preprint arXiv:1609.09152*, 2016.

[20] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[21] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] J. Jiang, J. Zhang, and K. Zhang, “Cascaded semantic and positional self-attention network for document classification,” *arXiv preprint arXiv:2009.07148*, 2020.

[23] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” *arXiv preprint arXiv:1502.03044*, 2016.

[24] K. Zhou, H. Wang, W. X. Zhao, Y. Zhu, S. Wang, F. Zhang, Z. Wang, and J.-R. Wen, “S3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization,” *Proc. 29th ACM Int. Conf. Information and Knowledge Management (CIKM)*, pp. 1893–1902, 2020.

[25] C. Xu, J. Feng, P. Zhao, F. Zhuang, D. Wang, Y. Liu, and V. S. Sheng, “Long- and short-term self-attention network for sequential recommendation,” *Neurocomputing*, vol. 423, pp. 580–589, 2021.

[26] X. Chen, Z. Wang, H. Xu, J. Zhang, Y. Zhang, W. X. Zhao, and J.-R. Wen, “Data augmented sequential recommendation based on counterfactual thinking,” *IEEE Trans. Knowledge and Data Engineering*, vol. 35, no. 9, pp. 9181–9194, 2023.

[27] Q. Liu, S. Wu, D. Wang, Z. Li, and L. Wang, “Context-aware sequential recommendation,” *arXiv preprint arXiv:1609.05787*, 2016.

[28] A. Rashed, S. Elsayed, and L. Schmidt-Thieme, “Context and attribute-aware sequential recommendation via cross-attention,” *Proc. 16th ACM Conf. Recommender Systems (RecSys)*, pp. 71–80, 2022.

[29] H. Du, H. Yuan, P. Zhao, D. Wang, V. S. Sheng, Y. Liu, G. Liu, and L. Zhao, “Feature-aware contrastive learning with bidirectional transformers for sequential recommendation,” *IEEE Trans. Knowledge and Data Engineering*, pp. 1–14, 2023.

[30] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, “Supervised contrastive learning,” *arXiv preprint arXiv:2004.11362*, 2021.

[31] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” *arXiv preprint arXiv:2002.05709*, 2020.

[32] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence embeddings using Siamese BERT-networks,” *arXiv preprint arXiv:1908.10084*, 2019.

[33] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *arXiv preprint arXiv:1512.03385*, 2015.

[34] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[36] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2019.

[37] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," *arXiv preprint arXiv:1912.01703*, 2019.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2017.

[39] Z. Khan, B. Latif, J. Kim, H. K. Kim, and M. Jeon, "DenseBert4Ret: Deep bi-modal for image retrieval," *Information Sciences*, vol. 612, pp. 1171–1186, 2022.

[40] I. Ahmed, N. Iltaf, Z. Khan, and U. Zia, "DvLIL: Deep-view linguistic and inductive learning based framework for image retrieval," *Information Sciences*, vol. 649, p. 119641, 2023.

[41] I. Ahmed, N. Iltaf, R. Latif, N. S. M. Jamail, and Z. Khan, "DM-RR: Dual modality reverse reranking based image retrieval framework," *IEEE Open J. Industrial Electronics Society*, vol. 5, pp. 886–897, 2024.

[42] M. Altamimi, Y. Altameemi, A. Alkhalil, R. F. Mansour, M. Abdelrhman, I. Ahmed, A. Ahmad, and A. Alogali, "A deep learning model for automated marking of students' assessments in a learning management system (LMS)," *Int. J. Advanced and Applied Sciences*, vol. 12, no. 10, pp. 1–10, 2025.

[43] I. Ahmed *et al.*, "Advanced multi-model deep learning approach for content-based image retrieval," *Proc. Korean Inf. Sci. Soc. Conf.*, Busan, 2023.

[44] Y. Fan, X. Xie, Y. Cai, J. Gao, and B. Cui, "STOSA: Stochastic self-attention for sequential recommendation," in *Proc. International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023.

[45] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *Proc. International Conference on Learning Representations (ICLR)*, 2024.