TRANSFORMER BASED SEQUENTIAL RECOMMENDER SYSTEM

By
Major Nadia Farooq

Supervisor
Assoc Professor Dr. Naima Iltaf

A thesis submitted to the Department, Computer Software Engineering Department, Military College of Signals (MCS), National University of Sciences and Technology, Islamabad, Pakistan, in partial fulfillment of the requirements for the degree of MS in Software Engineering

January 2023

ABSTRACT

Recommender systems (RS) aids in helping endusers by providing suggestions and predicting items of their interest in e-commerce and social media platforms. Sequence of user's historical preferences are used by Sequential Recommendation system (SRS) to predict next user-item interaction. In recent literature, various deep learning methods like CNN and RNN have shown significant improvements in finding recommendations, however, anticipating future item pertaining to user's past record history is still challenging. With the introduction of transformer architecture, SRS have gained major performance boost in generating precise recommendations. Recently proposed models based on transformer architecture predict next user-item by exploiting item identifiers only. Regardless of the efficacy of these models, we believe that performance of recommendation models can be improved by adding some additional descriptive item features along with the item identifiers. This paper proposes a transformer based SRS that models user behavior sequences, by incorporating auxiliary information along with item identifiers for producing more accurate recommendations. The proposed model extends the BERT4Rec model to incorporate auxiliary information by exploiting the *"Sentence Transformer model"* to produce the sentence representations from the textual features of items. This dense vector representation is then merged with the item representations of user. Comprehensive experiments upon various benchmark datasets shows remarkable improvements when corelating with other similar state-of-the-art models.

DECLARATION

I, Maj Nadia Farooq declare that this thesis titled ***"Transformer Based Sequential Recommender System"*** *has not been submitted before for any degree application at NUST or any other educational Institutes. This synopsis is presented as a result of my original research.*

Maj Nadia Farooq

(00000359480 / MSSE-27)

DEDICATION

This thesis is dedicated to

MY FAMILY, FRIENDS AND TEACHERS

for their love, endless support and encouragement

ACKNOWLEDGEMENTS

I am grateful to God Almighty, the Beneficiant and the most Merciful, who has bestowed me with the strength and the passion to accomplish this thesis.

I would like to convey my gratitude to my guide and support, Dr Naima Iltaf, CSE Department at MCS, who made me able to withstand the difficulties faced in this thesis. I would also extend my gratitude to my Thesis Committee Members for their support and guidance regarding the topic.

As a final words, I am extremely grateful to my husband, parents, and other relatives. They all have always stood by my dreams and have been a great source of inspiration for me. I would like to thank for all the care, love and support through my times of stress and excitement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

Recommender Systems	RS
Sequential Recommender System	SRS
Bidirection Encoder Representation for Transformer	BERT
Collaborative Filtering Recommender Systems	CFRS
Content Based Filtering	CBF
Deep Learning	DL
Recurrent Neural Network	RNN
Gated Recurrent Unit	GRU
Convulutional Neural Network	CNN
Long Short-teerm Memory	LSTM
Markov Chain	MC
Positional Embedding	PE
Learning rate	lr
Content Based Recommender Systems	CBRS
Hit Ratio	HR
Normalized Discounted Cumulative Gain	NDCG

Chapter 1

INTRODUCTION

A powerful recommender system (RS) aims at predicting the user preferences by characterizing the user intent that are usually dynamic in nature [27, 22, 34]. RS are widely being used in various online domain like e-commerce (Taobao, AliExpress and Amazon) and online media streaming websites like YouTube, NetFlix and facebook, to mitigate the efforts by the user in this information overload world [15]. User's likings are usually not stable and and keeps on changing with time. This temporal aspect is crucial in acquiring user dynamic preferences. For the purpose of identifying user intents more precisely, numerous sequential recommendation techniques have been introduced in recent past that uses users previous history [49,47,19,53,54] .

The intention of sequential recommendation (SR) models are basically to first gather the sequence of past objects in user's history and then projecting the most relevant and accurate interaction for each user. Traditionally, to model the user preferences in SR, researches exploits Markov Chain model to anticipate the future item in the sequence [28, 47]. The emergence of deep learning has resulted in massive number of work has been proposed using neural networks models like RNNs [41,36] and CNN [45]. The introduction of attention based Transformers [20], have motivated many researchers to practice and implement the technique to solve SR related problems. J. McAuley *et al* first introduced attention based transformer model(SASRec) to infer the user preferences in SR [53] . Since the model uses uni-directional attention mechanism for modeling the user sequence, it lacks in learning the optimal hidden representations of sequential user behavior. F.Sun *et al* proposed BERT4Rec which introduces a bi-directional architecture thus learning context from both directions [54]. Most SR models including SASRec and BERT4Rec consider only implicit or explicit feedback based on item identifier for next item recommendation thus ignoring auxiliary data (textual descriptions, keywords, reviews etc). By incorporating additional information, prediction accuracy of next items can be increased. KeBERT4Rec [62] integrates keywords along with the item identifier in BERT4Rec model by concatenating the keyword representation with item and its positional representations. However, this model uses one-hot encoding technique to generate the keyword vector, thus neglecting the contextual meaning of keywords. Another model, FDSA [61] utilizes the attribute information by applying a separate self-attention block for item in the user history as well as for the features. Although, these sequential recommendation models show significant performance gain, however, they do not exploit contextual features to generate meaningful representations.

We anticipate that incorporating auxiliary information in RS model will boost the recommendations especially under sparse conditions with low user-item interaction records. Therefore, with this aim of integrating auxiliary information, we present a modification of BERT4Rec model with added auxiliary information. Since the auxiliary information about the items are in sentences form, to generate the contextualized sentence embedding of that, we use Sentence-BERT [44] model and adding with the item-identifiers.

1.1 Problem Statement

Predicting the next item based on the user's interaction is very defying for SRS. With the introduction of transformer and BERT, SRS gained a major advancement because of the much precise recommendations. Several models have been proposed in the literature with the intent of precise and effective predictions to reduce the issues of sparsity and feature extraction. However, most of the proposed model lacks the auxilliary feature extraction. In this thesis, a Transformer based SRS is proposed to anticipate the upcoming item in the user interaction by using the contextual data information. The addition of some additional descriptive item features alongwith the item identifiers can enhance the performance of recommendations models.

1.2 Objectives

Objectives of the thesis work are:

- To build an efficient and accurate transformer based sequential recommendation system embedded with item rich features.
- Compare the suggested model with baseline models and recently developed state-of-the-art technique.
- To improve predictions accuracy in sequential recommender systems.

1.3 Research Contributions

The main contribution of this paper is to proposes a transformer based SRS that models user behavior sequences, by incorporating auxilliary information along with item identifiers for producing more accurate recommendations. These are summarized as below.

- introduce a model to incorporate the auxiliary information into the user behavior sequences using Masked Language Model (Close Objective Task) and bidirectional transformers.
- preparation of datasets to include the textual descriptions of items as auxiliary information along with the item identifier.
- generation of dense vector representations of item description by employing pre-trained sentence-BERT.
- model evaluation and performance comparison with existing state-of-the-art.

1.4 Areas of Application

The use of feature rich recommendation system will be beneficial for many online organizations. It will also improve the recommendations provided to their customers. A user-friendly sequential RS suggest the next course of action to the user based upon the previous interaction. In e-commerce industry, the customer will be recommended the next most relevant item. Thus, enhancing the organization's business by introducing feature integrated sequential recommendation systems. The majors application areas are as follows.

- Online Shopping e.g. Amazon, Alibaba
- Movies and TV shows recommendations e.g. Youtube, NetFlix
- Music recommendation e.g. Spotify, SoundCloud
- Finance Domain e.g. next investment recommendations
- Health care Domain e.g. next diagnosis/ follow up recommendations

A brief overview of the model is explained below. Detailed description is explained in later chapters. The proposed model is a deep learning based technique for sequential recommendations that incorporates the auxilliary information for predicting the next item in the user sequence. The model is developed based upon Transformer architecture. Before passing the sequence of items to the proposed model, the auxilliary information e.g. description, overview, movieplots of these items are gathered and contextualized embeddings of these features are extracted through Sentence-BERT, pretrained model. These auxilliary information is the textual description of the items in the form of sentences that are processed to extract the contextual dense feature representation which are then passed to the embedding layer where they are contcatenated with the item's embeddings and positional embeddings, to cater for the sequential behavior of the items. The concatenated embedding of sequence of items is then process through stack of Transformer layer from [20]. After processing through all layers, a final learned hidden representation is projected at output layer that contemplated the future item recommendation for a user.

1.5 Thesis Outline

This thesis is divided into five chapters:

- Chapter 1: This chapter contains introduction, objectives and the contributions made in this thesis. It also contains brief overview of the proposed model.
- Chapter 2: In this chapter, review of literature and background is given along with brief description of existing technique and quantitative measures used in this report.
- Chapter 3: In this chapter, our proposed Transformer based Sequential Recommender System is presented along with the introduction of the embedding technique being used in the proposed model is also explained.
- Chapter 4: This chapter discusses the experiment detail and analysis of the results by comparing with baseline models along with the brief explaination of the evaluation mer=trics being used to evaluate the model are also highlighted.
- Chapter 5: This chapter conclude the report and proposed the future work.

Chapter 2

PRELIMINARIES

2.1 Recommendation System

Recommender systems or Recommendation system (RS) is a tool and techniques that helps the users by providing suggestions and predicting items of their interest [1]. These days, RS are playing an important role in the very field especially in the field of education, e-commerce, social media, news, the roles of RS in both academia and industries cannot be neglected. RS are also being used by numerous media industries for the purpose of promotions and recommending movies and videos that are of more user interests [2].

The exponential growth of data in this time period aka big data era is one of the biggest reason of giving recommender system that much of importance. Since the internet is inundated with lots of unwanted information, recommendation systems is the only way of getting the right and meaningful data efficiently. Hence, there is an immense need to understand and consider the importance of recommendation system [3].

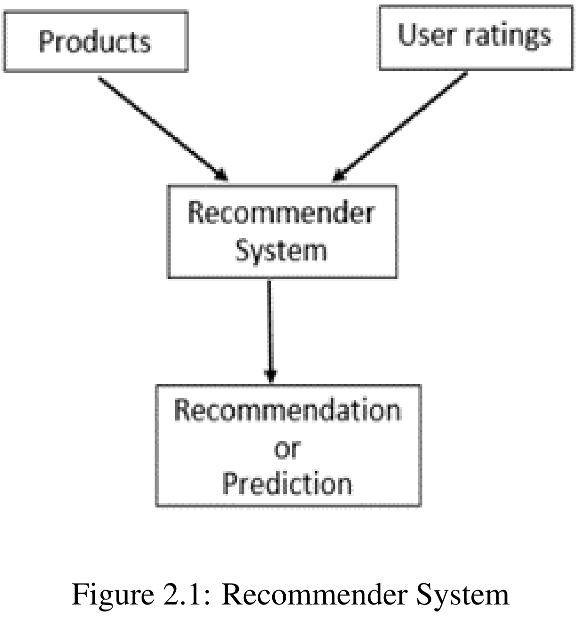
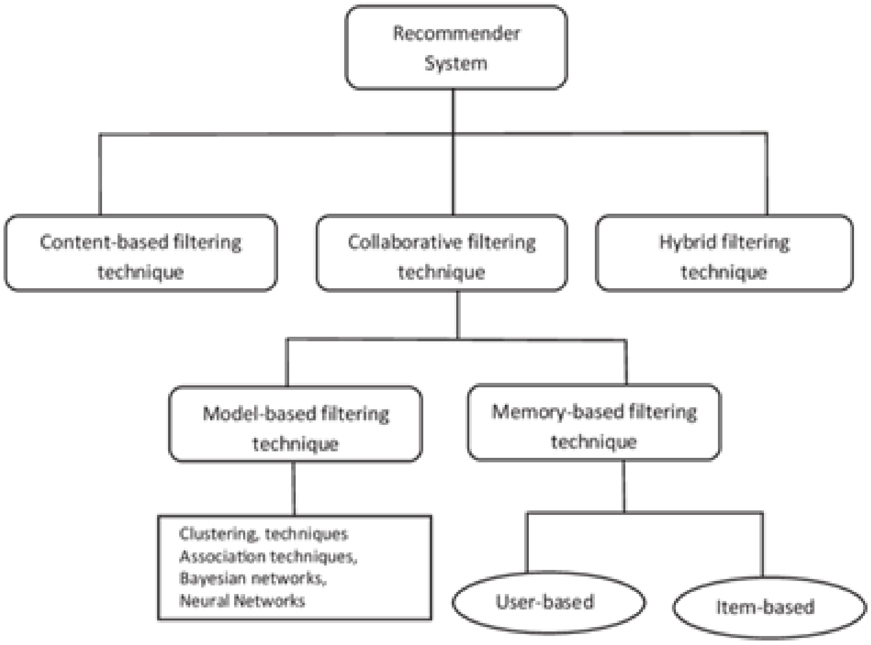


Figure 2.1: Recommender System

Deep learning (DL), field of machine learning, has appeared to be a most viable solution in which features in users and items are fetched and modelled automatically from a large amount of information/ data in RS [4]. Doing so, improves the quality of recommendation and also reduces the manual work. Moreover, in DL, we have the flexibility to model DL based RS by combining its different DL techniques, thus hybrid RS can also be modelled to enhance the performance and prediction quality. Such DL category of systems are very intelligent as predicting preference of an item that a user would like to give it as shown in Fig 2.1

2.2 Traditional Recommender System

The RSs are basically the software methods or approaches that evaluate the user preferences and the predicts the next or most likely user item of their interest [5]. These decision making systems are widely being used in the real world in the field of different decision making processes in the real world applications such as entertainment, health, e-commerce and social networks. RS can also be considered as a information filtering system that uses information filtering strategies to handle user mapping with item [7]. Numerous approaches that differentiate RS techniques have been developed depending on the knowledge that they utilize and arrange according to the recommendation made by those techniques. Three basic techniques which are being recommended and classified by the many authors collaborative filtering CF, content based CB and Hybrid recommender. Broader structure and classification of traditional recommendation system is depicted in Fig 2.2



Recommender System
Content-based filtering technique
Collaborative filtering technique
Hybrid filtering technique
Model-based filtering technique
Memory-based filtering technique
Clustering, techniques
Association techniques,
Bayesian networks,
Neural Networks
User-based
Item-based

Figure 2.2: Classification of Traditional Recommender System

2.2.1 Collaborating Filtering

Collaborative Filtering, CF makes the recommendation relying on the previous interactions of user-item with the user having similar preferences [8]. CF is the most extensively used methods in the RS [9]. CF is also known as social filtering, as it filter the information through analysing recommendations of peoples in a community. It is working on concept that a users, previously agreed in certain evaluation of their items would also agree again in the future as well. Thus CF is working on the past experiences of user-user, item-item and user-item relationship. CF is classified into two major types, memory based CF and model based CF.

Memory based CF methods learns the full user-item matrix and then use this learned matrix to find the similarity [10]. It is further grouped into user-based CF and item-based CF. Model Based CF methods constructs a model that observes the user-item interactions and predicts the similar item.

2.2.2 Content-Based Filtering

Content-based Filtering (CBF) is an algorithm that generates predictions by focusing mainly on the user characteristics [11]. Content-based filtering technique is used by many recommender system which exploits content of the item to generate features to calculate users profiles of each user and content of new product is compared with to content of items that a user has liked in the past and top matched products are recommended to users as shown in Figure 2.3. We can say it a domain-dependent technique. The CBF methods is mainly used for recommending news and other documents like magazines. web mining, and journals. In CBF method, recommendations are made based on the user profiles that are generated depending upon the features and attributes i.e. contents of the item which the user has acted

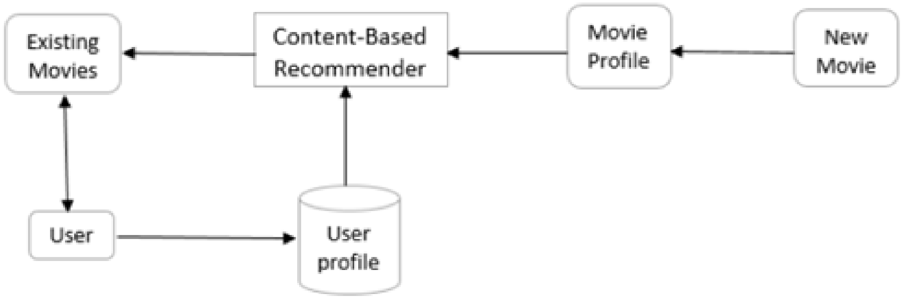


Figure 2.3: Content Based Recommender System

(likes/ dislikes, clicks/ views etc) upon in past [12]. To generate a significant recommendation, different types of algorithms are used to find the similarity among the documents or the items such as TF/IDF , decision tree and Naive Bayesian Classifier are the most popular.

2.2.3 Hybrid Recommendation System

In hybrid RS, both the approaches i.e. CF and CBF are integrated with each other to build a strong RS. The crux of the hybrid Rs is that we can improve the quality of recommendation by combining both CB and CBF techniques. Moreover, integrating different methods can also overcome the disadvantages of single method. Authors have proposed different methods for hybrid RS like Aslanian et al. has proposed feature augmentation and feature combination method [12] . Another paper has suggested the cascading and switching method [13]. The taxonomy of hybrid filtering is shown in Figure 2.4

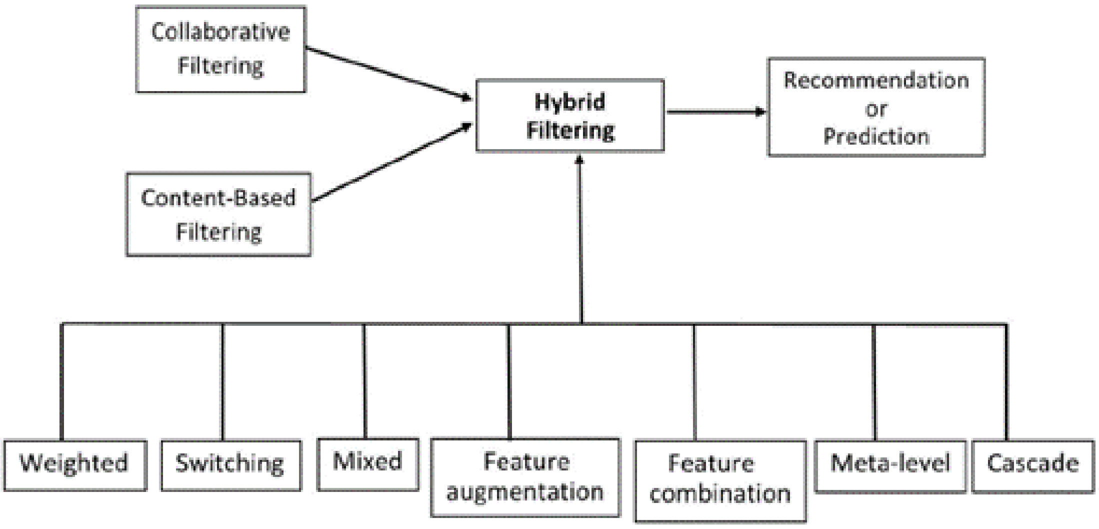


Figure 2.4: Hybrid Recommender System

2.3 Sequential Recommender

Sequential recommender is a kind of RS exploiting the user interaction sequences to infer the successive item [16]. The aim of SR is to recommend future product by considering historical behavior of users, also known as *next item prediction*. Earlier, the SRS were introduced using Markov Chains (MC) models for capturing sequential patterns from the user historical preferences [29, 28, 32]. The next item preferred by the users are predicted depending upon the last item, thus interpreting only the adjoining sequential behavior.

Recurrent Neural Network (RNN) based models exploiting GRU [48, 41] along with LSTM [23] have showed substantial performance gain for SR [43, 25, 26, 36, 41, 57, 42]. RNNs enforce rigid sequential patterns for encoding user preferences for making predictions. Besides RNN, a number of Convulational Neural Network (CNN) [58,59] based RS have also been introduced that also target problems related to the sequential recommendation. For example, Tang and Wang *et al* [45] exploit CNN for capturing local sequential features using more recent behaviors.

Recently, attention mechanism [20] based sequential recommendation models have shown extraordinary performance in the domain of text classification [55], image captioning [38] and machine translation [46]. Some other attention based model [60,53,54] have also shown exceptional results in modelling sequetial information. These sequential recommendation techniques exploit the item identifiers for next item recommendation. Models proposed by [61, 62, 35, 33] incorporate additional information for the prediction of successive item. A feature level deeper self attentive model [61] introduced by T. Zhang *et al* exploits segregated attention blocks for items and their associated features for the purpose of next item prediction. [35] proposed S$\hat{3}$Rec, a self supervised sequential recommendation model that utilized the attribute data of item to learn the correlation among them. KeBERT4Rec [62] leverages the keyword by integrating them with item identifier for the prediction of next item in sequence. However, keywords representations are not extracted through any of the contextual embedding technique, thus losing the context meanings. GRU4RecBE, an extension of GRU4Rec [41] model uses the rich item features embedding generated through pre-trained BERT and processed through the GRU-RNN layer [33].

2.4 Transformer - Attention Block

Transformers, primarily modeled for natural language processing, have shown revolutionary impact in the field of sequential recommendations [16]. For modeling the sequential data, only the encoder part of the Transformer is used that aims at mapping the sequence of items that represents the user interaction history into the sequence of vector representations [20]. Using Transformer in SR, a sequence of items are passed as input that is encoded through embedding layer followed by concatenating with the positional embedding (vector representations that learns the item's placement in the sequential order) and processed via Attention block layer, Transformer. A single Transformer attention block comprising of two seperate and independent sublayers. One is ”multi-head self attention” layer in which the input representation is interacted and computed within itself. The output of this layer is feeded as in input to the ”position-wise feed forward” layer of Trasformer that utilizaes weight during the training of model.

SASRec and BERT4Rec exploits the item identifiers for modelling the user interaction. KeBERT4Rec uses keywords along with item identifiers for next item prediction. However, keywords representations are not extracted through any of the contextual embedding technique, thus losing the context meanings. Proposed model incorporates the auxiliary information along with the item identifiers and constructs the embedding using Sentence BERT [44] embedding technique to capture the contextualized representations, thus enhancing the item recommendation and prediction accuracy.

Summary

In this chapter, a broad background to recommender systems is explained. Different types of recommender systems along with existing research is also presented. Furthermore, a brief introduction to sequential recommendation and its relevant studies are explored. What challenges a sequential recommender system has to cater for is also included. in the end, Tranformer that is attention based architecture is also intrduced and relevant literature review incorporating transformer based Recommendation system is also highlighted in this chapter.

Chapter 3

CONTEXTUAL SEQUENTIAL RECOMMENDATION SYSTEM

3.1 Contextual Sequential Recommendation System

In this section, the proposed framework ”Contextual Sequential Recommendation System” is illustrated. Proposed model is developed based upon Transformer architecture that adapted the deep bidirectional BERT model for SR prediction task. Proposed model architecture is illustrated in Figure 3.1. Before passing the sequence of items to the model proposed, the auxiliary features of these items are taken as input to the Sentence-BERT. This auxiliary information is the textual description of the items in the form of sentences that are processed to extract the contextual dense feature representation and stored in a matrix *PE*. These dense embedding are extracted prior to training phase to reduce the model training time.

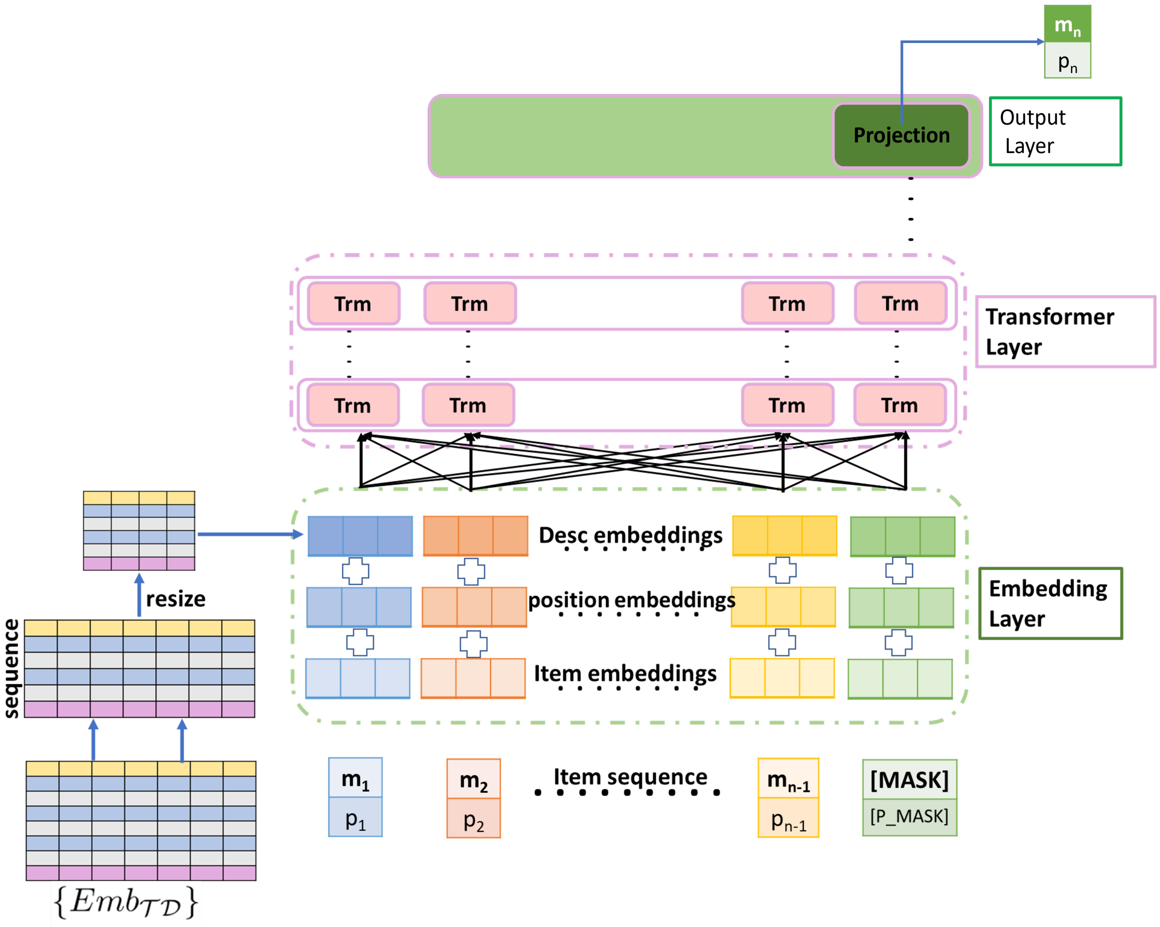


Figure 3.1: Model Architecture of Contextual Sequential RS

Before passing the sequence of items to the model proposed, the auxiliary features of these items are taken as input to the Sentence-BERT. This auxiliary information is the textual description of the items in the form of sentences that are processed to extract the contextual dense feature representation and stored in a matrix *PE*. These dense embedding are extracted prior to training phase to reduce the model training time.

Subsequently, during the training of a sequence of items, the auxiliary information's embedding of items in the sequence are extracted from *PE* as shown in Figure 3.1 and resized to match the dimensionality according to the given *batch_size*. These embedding are then passed to the embedding layer where they are concatenated with the item's embedding and positional embedding presenting the sequential behavior of the items. This resultant concatenated item's representations of sequence are then processed through stack of Transformer layer from [20] where hidden features for each item are calculated simultaneously at each layer. Only the encoder part of Transformer is used to compute the hidden representation using self attention mechanism for each item. These layers share information bidirectionally across each position in hierarchical manner. After processing through all layers, a final learned hidden representation is projected at output layer that contemplated the future item recommendation for a user. Numerous Experiments performed using three benchmark datasets including movielens-1m, movielens-20m and Amazon Beauty to prove the effectiveness of the proposed model. The layers of proposed model are assembled using embedding layer, transformer layer and the output layer.

3.1.1 Mathematical Formulation of Proposed Model

Let set of users be shown mathematically as $\mathcal{U}$ = $\{u_1$, u_2, u_3, ..., $u_{|\mathcal{U}|}\}$ $\mathcal{M}$ = $\{m_1$, m_2, m_3, ..., $m_{|\mathcal{M}|}\}$ be the set of items. For each item, there is some item description (auxiliary information) that is in textual form denoted as $\mathcal{TD}$ = $\{des_1$, des_2, des_3, ..., $des_{|\mathcal{M}|}\}$. The items of user interacted by in the sequence be $\mathcal{S}$ in historical order of a user u is denoted as $\mathcal{S}$ = $\{m_1$, m_2, m_3, ..., $m_n\}$ where m_n is the sequence of items from $\mathcal{M}$, the user has acted upon previously. Given the sequence history $\mathcal{S}$, the objective of the SRS is to anticipate the future item m_n+1 a user will reach out to as

$$\mathcal{P}(m_n+1 = m|\mathcal{S})$$

3.1.2 Embedding Layer

The recommendation model in [54] make use of the positional embeddings along with the item identifier embedding to maintain the sequence of the items, thus memorizing the sequential order of the input. However, the pair alone does not describe the contextual representation of the input and hence dose not recommend contextually especially under sparse conditions with low user-item interaction records. To overcome this limitation, the propose model incorporates additional auxiliary information based on contextualized description of items. The proposed model utilizes the Sentence-BERT [44] for capturing contextual representation of the item descriptions. The architecture of Sentence-BERT for extracting sentence embedding is depicted in Figure 3.2. Sentence-BERT works in two layer. First, it utilizes BERT to generate word/ token embedding. Input in the form of sentences or text of various length is injected to the selected SBERT model, that generates contextualized word embedding for all input tokens in the sentence. Secondly, these word embedding are passed through a pooling layer to generate a fixed-sized vector representation. Among various pooling options available, sentence-BERT utilizes the mean pooling in which mean of all contextualized token embedding is calculated to produce a fixed dimensional output embedding vector. Given the item descriptions of various length of all items, $\{\mathcal{TD}\}$ as input, the model produces 384 dimensional densed vector representation $\{Emb_{\mathcal{TD}}\}$ as in equation 3.1 These 384 dimensional embedding are stored in a matrix and used along with the item identifier and position embedding to produce information rich vector representations as

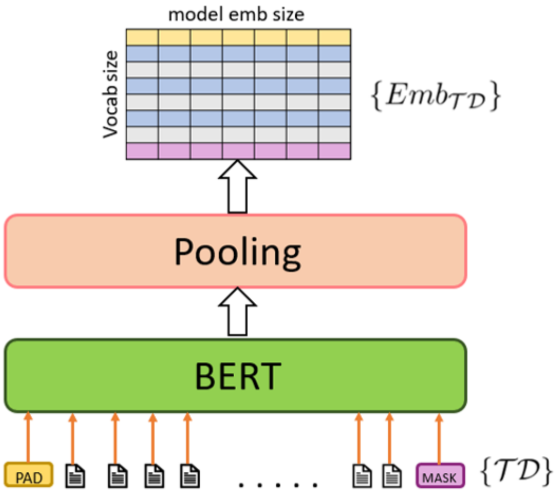


Figure 3.2: Design Architecture of Sentence-BERT

shown in equation 3.2.

$$\{\mathcal{TD}\} \xrightarrow{Sentence-BERT} \{Emb_{\mathcal{TD}}\} \tag{3.1}$$

In the proposed model, *d* dimensional embedding layer is constructed by summing up the item identifier embedding, the position embedding and the additional auxiliary information (item description) extracted from $\{Emb_{\mathcal{TD}}\}$. Thus, for a given item m_i, the input embedding matrix $\mathcal{EM}$ is formulated by adding the corresponding item embedding E_m , position embedding E_{pos} and textual description embedding E_{des} as:

$$\mathcal{EM}_m = E_m + E_{pos} + E_{des} \tag{3.2}$$

3.1.3 Transformer Layer

The summed embedding $\mathcal{EM}$ becomes the input to the transformer layer that iteratively calculates the hidden representations of each item at each layer. The structure of transformer layer or simply the encoder layer is build using the ”multi-head attention” technique. The layer piles up multiple encoder blocks [20] each consisting of *”Multi-Head Self Attention, MHSA Layer”* and a second layer that follows MHSA is the *”Position-wise Feed Forward Network Layer”*. Given that $\mathcal{E}^0 = [\mathcal{EM}_0^0, \mathcal{EM}_1^0, ...\mathcal{EM}_t^0]$ depict the dense vector embedding of the user sequence to the transformer, **multi head self attention layer, MHSA** is defined as:

$$MHSA(\mathcal{E}^l) = [h_1; h_2; ....h_h]W^0 \tag{3.3}$$

$$h_i = Attn(\mathcal{E}^l W_i^Q, \mathcal{E}^l W_i^K, \mathcal{E}^l W_i^V) \tag{3.4}$$

where W_i^Q, W_i^K and $W_i^V \in R^{dxd/h}$ are the three learnable projection weight matrices and $W_i^0 \in R^{dxd}$. $\mathcal{E}^l W_i^Q$, $\mathcal{E}^l W_i^K$, $\mathcal{E}^l W_i^V$ are the three linear transformation of input vecor representation $\mathcal{E}'$ for Query, Key and Value (Q,K,V) vectors. A function known as scale dot product has been used here as [20]:

$$Attn(Q, K, V) = \sigma\left(\frac{QK^T}{\sqrt{d/h}}\right)V \tag{3.5}$$

where the Query, Key and Value matrices are denoted by **Q, K, V** respectively and σ being the softmax function. Let MHSA at the l^{th} layer be S_i. Since, the MHSA block is based on linear projections, thus, the non-linearity to the MHSA is empowered by applying **position-wise feed-forward network layer, PFN** on all MHSA(S_i) separately.

$$PFN = [FNL(S_1^l)^T, FNL(S_2^l)^T......FNL(S_n^l)^T] \tag{3.6}$$

$$FN(S_i) = GELU\big(S_i W^{(1)} + b^{(1)}\big)W^{(2)} + b^{(2)} \tag{3.7}$$

A smoother GELU activation function is used inline with BERT [19] and OpenAI GPT [21]. GELU is calculated as $x\phi(x)$. $W^{(1)}, b^{(1)}, W^{(2)}$, and $b^{(2)}$ are all hyper-parameters communicated at all layers. Complexity of the model is reduced using residual connection at each sub layer. Dropout is applied followed by layer Normalization, LNorm. Thus, the sublayer output at each level is $LNorm(x + Dropout(sublayer(x)))$. Input at each layer is denoted by x in the LNorm and represented as:

$$\mathcal{E}^l = Trm\big(\mathcal{E}^{l-1}\big), \quad \forall i \in [1, 2, ...L] \tag{3.8}$$

$$Trm(\mathcal{E}^{l-1}) = LNorm\bigg(S_i^{l-1} + Dropout\big(PFN(S_i^{l-1})\big)\bigg) \tag{3.9}$$

$$S_i^{l-1} = LNorm\bigg(\mathcal{E}^{l-1} + Dropout\big(MHSA(\mathcal{E}^{l-1})\big)\bigg) \tag{3.10}$$

3.1.4 Output Layer

After passing through L layers and shared representations bidirectionally over each position in hierarchical manner, a final learned hidden representation $\mathcal{E}^{\mathcal{L}}$ is projected at output layer for all input item sequences. Considering the last item m_n in the sequence is masked, m_n is anticipated using embedding sequence $\mathcal{E}^{\mathcal{L}}$ that is depicted in Figure 3.1. The encoder Transformer's last layer applies linear transformation twice followed by softmax function is used to predict the masked item.

$$P(m) = softamx\big(GELU\big(\mathcal{E}_t^L W^P + b^P\big)\mathcal{E}\mathcal{M}^T + b^O\big) \tag{3.11}$$

where b^P and b^O are the bias at projection layer and W^P is the projection matrix. $\mathcal{E}\mathcal{M}^T$ is the item embedding matrix comprising of item identifier, positional and auxiliary information embedding. Here, shared item embedding is applied to minimize model size and relieve over-fitting.

Chapter 4

EXPERIMENTAL RESULT AND ANALYSIS

4.1 EXPERIMENTS

The present part of the paper elaborates the datasets used in the proposed model and their preparation followed by experiment setup, evaluation metrics and performance comparison.

4.1.1 Datasets Pre-processing

Performance of proposed model is demonstrated through experiments carried out on three benchmark datasets including movielens-1m , movieLens-20m (ml-1m[1] and ml-20m[2])and Amazon-Beauty[3] is described below:

- **MovieLens**: A well-known dataset most commonly used for evaluating the performance of SRS. MovieLens ratings dataset contains the user id, item id (IDs of the movies from ”movies” table), ratings and timestamp for movie ratings from each user. The auxiliary information (movie plot summary) for MovieLens is extracted through IMDbPY[4] using the **ImdbId** unique identifier, thus, making it information rich dataset.

- **Amazon - Beauty**: It is a set of dataset comprising of reviews of a number of products extracted from ”Amazon.com”. The data is split into multiple datasets based upon product categories on Amazon. In our experiments, ”Beauty” category is chosen that has a ”rating” and a ”meta” file. To incorporate the auxiliary information in the ”rating” dataset, ”description” of each product is extracted from the ”meta” dataset.

The following Table 4.1 summarizes the dataset statistics.

4.1.2 Evaluation Metrics

To measure the overall SR behavior, widely used leave-one-out strategy [53, 54, 62] is employed. The user sequence of all users is split into three part to train the model. The final

[1] https://grouplens.org/datasets/movielens-1m/

[2] 1https://grouplens.org/datasets/movielens-20m/

[3] 3http:/jmcauley.ucsd.edu/data/amazon/

[4] https://imdbpy.github.io/

Table 4.1: Datasets statistics.

Benchmark Dataset	Users Count	Items Count	interactions Count	Sparsity
MovieLens-1m	6,040	3,706	1.0m	95.16%
MovieLens-20m	138,493	26,744	20.0m	95.53%
Beauty	22,363	12,101	0.2m	99.93%

item being used for testing, the penultimate item has been earmarked for validation, and the remaining interaction in the user preferences are utilized for training. For fair assessment, commonly used sampling practice [53, 54, 62] i.e. the ground truth object is coupled item with 100 negative items in test set based on how popular they are.

For evaluating all methods, 'Normalized Discounted Cumulative Gain' (NDCG), 'Hit Ratio' (HR) and 'mean reciprocal rank' (MRR) are calculated. Higher values of these metrics depicts how better the recommendation performance is. Hit Ratio (HR) is used for measuring the ranking accuracy by comparing the test item set (T) with the ranked list. Mathematically it is expressed as:

$$H@K = \frac{Number of Hits@K}{|T|} \tag{4.1}$$

Hit ratio is denoted as H@K calculates the number of hits in a K-sized list. A hit occurs if the item tested is available in ranked list. Whereas the relative position of that item is assessed using NDCG in the ranked list . It assigns higher scores if the item is present at top position in ist. Mathematically it is evaluated by folllowing formula:

$$G@K = N_K \sum_{j=1}^{K} \frac{2^{z_j} - 1}{log_2(j+1)} \tag{4.2}$$

where N_K is the normalizer and z_j being the item's graded relevance at position j. We compute both the metrics of every test user items and then take their mean.

4.1.3 Baselines

For performance comparisons, we consider the following methods.

- **BPR-MF** [18]: This model is the first one that uses the Bayesian personalized ranking loss for the optimization of matrix factorization (MF).

- **NCF** [31]: This model utilizes MLP for capturing the item sequence interacted by user instead of using inner product in MF.

- **FPMC** [47]: Sums up both MF and MC to encapsulate the long-term user interaction.
- **GRU4Rec** [41]: It models the user click sequences using RNN-GRU for session based recommendation.
- **SASRec** [53]: It is a unidirectional (left-to-right) self attentive model for next item prediction.
- **BERT4Rec** [54]: This top of the line model uses bidirectional self attentive blocks and Cloze [40] masking for the recommendation task.
- **KeBERT4Rec** [62]: This model extends BERT4Rec [54] by integrating keywords as additional input layer.

4.1.4 Implementation Details

The proposed model is trained on machine having 16 GB RAM and NVIDIA GTX 3080Ti (11GB). The training of proposed model is done using Adam Optimizer [24] and having the initial lr and weight decay of 0.001 and 0.01 respectively. The hidden dimension is set to 128, dropout to 0.1 and 200 value used for maximum sequence length for MovieLens datasets and 50 value for Amazon Beauty. The masking probability of 0.15 is set for ML-20m and ML-1m. A 256 of batch size is used to traing the proposed model.

We cited the results from the author at [54] for BPR-MF, NCF, FPMC, GRU4Rec, BERT4Rec . For SASRec[5], KeBERT4Rec[6], the code provided by the corresponding authors were executed. The optimized settings for hyperparameter values are used for all baseline models. The *hidden dimensionality* is tested from {64,128,256}, *dropout* from {0.1-0.9}, l_2 *regularizer* from {0-0.0001}.

4.1.5 Comprehensive Performance Analysis

Table 4.2 presents the optimized outcomes of each baseline models on benchmark datasets. The final column displays how the proposed model performs in comparison to the best baseline. The results of G@1 corresponding to all baseline is ignored since in our experiments, it is same as H@1. It is evident from table 4.2 that outcomes of all the sequential models like GRU4Rec, BERT4Rec, SASRec etc outperformed the non-sequential models like

[5] https://github.con/kang205/SASRec

[6] https://github.com/elisabethfischer/KeBERT4Rec

Table 4.2: Comprehensive performance analysis of proposed model with referenced models for next item recommendations. The highest scores are shown in bold, while the 2nd place scores are underlined.

Datasets	Eval Met	BPR-MF	NCF	FPMC	GRU4Rec	SASRec	BERT4Rec	KeBERT4Rec	Ours	Improv
ML-1m	H@1	0.0914	0.0397	0.1386	0.1583	0.2351	0.2863	0.2615	**0.3672**	24.73%
	H@5	0.2866	0.1932	0.4297	0.4673	0.5434	0.5876	0.5873	**0.6690**	12.36%
	H@10	0.4301	0.3477	0.5946	0.6207	0.6629	0.6970	0.7651	**0.7761**	1.44%
	G@5	0.1903	0.1146	0.2885	0.3196	0.3980	0.4454	0.5134	**0.5287**	2.98%
	G@10	0.2365	0.1640	0.3439	0.3627	0.4368	0.4818	0.5488	**0.5633**	2.64%
	MRR	0.2009	0.1358	0.2891	0.3041	0.3790	0.4254	0.4322	**0.4484**	3.70%
ML-20m	H@1	0.0553	0.0231	0.1079	0.1459	0.2544	0.3440	0.5420	**0.6512**	20.15%
	H@5	0.2128	0.1358	0.3601	0.4657	0.5727	0.6325	0.8770	**0.9863**	12.46%
	H@10	0.3538	0.2922	0.5201	0.5844	0.7136	0.7473	0.9450	**0.9981**	5.62%
	G@5	0.1332	0.0771	0.2239	0.3091	0.4208	0.4967	0.7250	**0.7687**	6.03%
	G@10	0.1786	0.1271	0.2895	0.3637	0.4665	0.5340	0.7470	**0.8237**	10.27%
	MRR	0.1503	0.1072	0.2273	0.2967	0.4026	0.4785	0.4813	**0.5384**	11.86%
Beauty	H@1	0.0415	0.407	0.0435	0.0402	0.0906	0.0953	0.1846	**0.2036**	10.29%
	H@5	0.1209	0.1305	0.1387	0.1315	0.1934	0.2207	0.3751	**0.3884**	3.55%
	H@10	0.1992	0.2142	0.2401	0.2343	0.2653	0.3025	0.4753	**0.5321**	11.95%
	G@5	0.0814	0.855	0.0902	0.0812	0.1436	0.1599	0.2841	**0.3261**	14.78%
	G@10	0.1064	0.1124	0.1211	0.1074	0.1633	0.1862	0.3164	**0.3394**	7.27%
	MRR	0.1006	0.1043	0.1056	0.1056	0.1536	0.1701	0.2353	**0.2517**	6.97%

BPR-MC and NCF. The advantage of FPMC over BPR-MC is that it sequentially models users' previous records. From this observation, the importance of taking sequential pattern in consideration for recommendation systems can be ascertained.

Comparing the sequential baseline models, SASRec model outperforms GRU4Rec and FPMC on all benchmark datasets. This observation demonstrate that use of transformer based self attention models are more accurate than using traditional mechanisms. However, SASRec performance fall behind as compare to BERT4Rec which depicts that bidirectional model like BERT4Rec is more powerful as compared to unidirectional model like SASRec. Furthermore, KeBERT4Rec perform better than BERT4Rec suggesting that incorporating some kind of side information along with item can improve the recommender's performance.

In accordance with the results, on the three benchmark datasets, our proposed model clearly outperforms all baseline methods. The proposed model gains an average improvement of 6.34% on 'H@10', 6.72% on 'G@10' and 7.51% on 'MRR' as compared to the best baselines.

4.1.6 Impact of Integrating Auxiliary Information

To visualize the impact of using side information along with item identifier, the proposed model is initially trained by excluding the side information. The results are compared with

Table 4.3: Analysis on the incorporation of auxiliary information

Model	Beauty			ML-1m		
	H@10	G@10	MRR	H@10	G@10	MRR
BERT4Rec	0.3025	0.1862	0.1701	0.6970	0.4818	0.4254
Ours (without side Info)	0.3321	0.1922	0.1653	0.7023	0.4953	0.4308
Ours (with side Info)	0.4631	0.3120	0.2581	0.7761	0.5633	0.4484

the proposed model by integrating side information. It is evident from the results that auxiliary information can enhance the productivity of SR system. Only the results on ml-1m and beauty dataset with batch size 128 are reported here in 4.3 due to space limitations which clearly depicts that excluding the side information from proposed model degrades the performance.

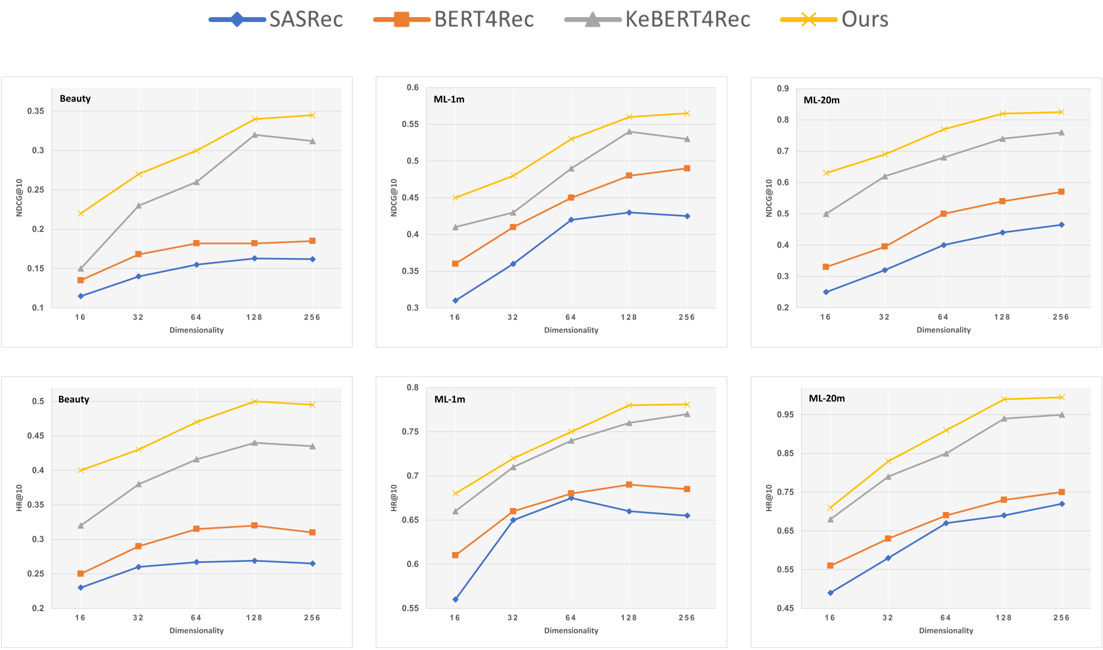


Figure 4.1: **hidden dimensionality,*d* impact on NDCG@10 and HR@10 for baseline SR models.**

4.1.7 Evaluating Effect of ”hidden dimensionality” *d*

The hidden dimensionality d has a great impact on the performance of recommendation system that is studied in this section. Figure 4.1 exhibits the values of H@10 and G@10 on different baseline sequential model by varying hidden dimensionality d ranges between 16,32,64,128,256. The emaining of the hyperparameters are constant and kept to their optimal values. It is obvious that with the increase of dimensionality, the graph of each model converges. However, improved model performance is not always achieved with bigger value of hidden dimensionality, particularly on sparse datasets such as Beauty.

	Metric	**With One Hot Encoding**	**Using Sentence-transformer**
Ml-1m	H@1	0.3502	0.3672
	H@5	0.6563	0.6690
	H@10	0.7651	0.7761
	G@5	0.5134	0.5287
	G@10	0.5488	0.5633
	MRR	0.4322	0.4484
Beauty	H@1	0.1897	0.2038
	H@5	0.3432	0.3884
	H@10	0.4983	0.5321
	G@5	0.3079	0.3261
	G@10	0.3187	0.3394
	MRR	0.2263	0.2517

Table 4.4: Impact of Using Contextual Embedding Technique

4.1.8 Ablation Study

Finally, to visualize the impact of incorporating auxiliary information, some ablation experiments were conducted. Sentence-Transformer is used to train the proposed model which is a pre-trained model for generating embeddings of item's side information. However, the KeBERT4Rec model is trained using one of its one hot encoding technique to generate the embedding. To analyze the impact of using contextual embedding instead of traditional techniques, the proposed model is tested using one hot encoding techniques to generate textual embedding. As depicted in 4.4, the results of proposed model on ml-1m and beauty datasets, using sentence-transformer, a contextual embedding technique for the generation of meaningful dense vector representations outperforms all other non-contextual methods like word2vec, doc2vec etc to generate embedding. This also emphasize the use of meaningful and context embedding generating technique for model training to produce relevant results.

Chapter 5

CONCLUSION AND FUTURE WORK DIRECTIONS

5.1 CONCLUSION

Self Attention and Transformer based recommendation system have proven to be more precise and accurate as compared to traditional RS. In this paper, a transformer based sequential RS have been proposed that enhances recommendation accuracy by incorporating contextual auxiliary information of items in a sequence. To generate the embedding of auxiliary information, a contextual based pre-trained model *sentence-transformer* is adopted. This model has the capability to generate meaningful embedding for a textual information. Comprehensive experiments on various datasets shows remarkable improvements as compared to top of the line models.

BIBLIOGRAPHY

[1] Khanian M, Mohd N (2016) A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback. Artif Intell Rev 45(2):167–201

[2] Covington P, Adams J, Sargin E (2016a) Deep neural networks for Youtube recommendations. In: *RecSys 2016—proceedings of the 10th ACM conference on recommender systems*

[3] Suryana, N.,Basari, A. S. B. H (2018). Deep learning for recommender system based on application domain classification perspective: A review. *Journal of Theoretical & Applied Information Technology*, 96(14).

[4] Yao CSL, Sun A (2017) Deep learning based recommender system: a survey and new perspectives. *ACM J Comput Cult Herit Article 1(1):*1–35

[5] Lu J, Wu D, Mao M, Wang W, Zhang G (2015) Recommender system application developments: a survey. *Decision Support Systems*.

[6] Lee G, Jeong J, Seo S, Kim C, Kang P (2018) Sentiment classifcation with word localization based on weakly supervised learning with a convolutional neural network. *Knowl-Based Syst* 0: 1–13.

[7] Wang Y, Liu Y, Yu X (2012) Collaborative fltering with aspect-based opinion mining: a tensor factorization approach. In: Proceedings—IEEE international conference on data mining, ICDM, pp 1152–1157

[8] Batmaz Z, Yurekli A, Bilge A, Kaleli C (2018) A review on deep learning for recommender systems: *challenges and remedies.* Artif Intell Rev

[9] Wang H, Wang N, Yeung D-Y (2014) Collaborative deep learning for recommender systems. In: *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining*, 2015–August.

[10] Sharma R, Gopalani D, Meena Y (2017) Collaborative filtering-based recommender system: approaches and research challenges. In: *2017 3rd international conference on computational intelligence & communication technology (CICT)*, pp 1–6.

[11] Da'u, A., & Salim, N. (2020). Recommendation system based on deep learning methods: a systematic review and new directions. *Artificial Intelligence Review*, 53(4), 2709-2748.

[12] Isinkaye FO, Folajimi YO, Ojokoh BA (2015) Recommendation systems: principles, methods and evaluation.Egypt Inf J 16(3):261–273

[13] Aslanian E, Radmanesh M, Jalili M (2016) Hybrid recommender systems based on content feature relationship. *IEEE Trans Ind Inf 3203(c)*:1

[14] Paradarami TK, Bastian ND, Wightman JL (2017) A hybrid recommender system using artifcial neural networks. *Expert Syst Appl 83:300–313*.

[15] Latifi, Sara, Dietmar Jannach, and Andrés Ferraro. ”Sequential recommendation: A study on transformers, nearest neighbors and sampled metrics.” *Information Sciences* 609 (2022): 660-678.

[16] Aleksandr Petrov and Craig Macdonald. 2022. A Systematic Review and Replicability Study of BERT4Rec for Sequential Recommendation. In *Proc. RecSys*

[17] Wilson L. Taylor. 1953. Cloze Procedure: *A New Tool for Measuring Readability Journalism Bulletin* 30, 4 (1953), 415–433.

[18] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “BPR: bayesian personalized ranking from implicit feedback,” in *UAI, 2009*.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. CoRR abs/1810.04805 (2018)

[20] Vaswani, et. al. 2017 *Attention Is All You Need.* 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA.

[21] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. *Improving language understanding by generative pre-training.* In OpenAI Technical report.

[22] 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.

[23] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. Neural Computation 9, 8 (Nov. 1997), 1735–1780.

[24] Kingma, Diederik P., and Jimmy Ba. ”Adam: A method for stochastic optimization.” In *Proceedings of ICLR*.

[25] Qiang Liu, Shu Wu, Diyi Wang, Zhaokang Li, and Liang Wang. 2016. Contextaware sequential recommendation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 1053–1058.

[26] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 30*

[27] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*. 165–174.

[28] Guy Shani, David Heckerman, and Ronen I Brafman. 2005. An MDP-based recommender system. *Journal of Machine Learning Research* 6, Sep (2005), 1265-1295.

[29] Ruining He and Julian McAuley. 2016. Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th International Conference on Data Mining (ICDM).* IEEE, 191–200.

[30] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. Computer 42, 8 (2009), 30–37.

[31] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In Proceedings of *WWW. 173–182.*

[32] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web.* 811–820.

[33] Potter, Michael, Hamlin Liu, Yash Lala, Christian Loanzon, and Yizhou Sun. ”GRU4RecBE: A Hybrid Session-Based Movie Recommendation System (Student Abstract).” (2022).

[34] Liu, Zhiwei, et al. ”Contrastive self-supervised sequential recommendation with robust augmentation.” arXiv preprint arXiv:2108.06479 (2021).

[35] Zhou, Kun, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. ”S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization.” In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 1893-1902. 2020.

[36] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. In *Proceedings of CIKM.* ACM, New York, NY, USA, 843–852

[37] Hidasi, B., Quadrana, M., Karatzoglou, A., Tikk, D. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems. pp. 241–248*. Rec-Sys ’16, ACM, New York, NY, USA (2016).

[38] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *ICML*, 2015.

[39] Tuan, T.X., Phuong, T.M. Cloze Procedure: A New Tool for Measuring Readability In *Proceedings of the Eleventh ACM Conference on Recommender Systems. pp. 138–146*. Rec-Sys ’17, ACM, New York, NY, USA (2017).

[40] Wilson L. Taylor. 1953. Cloze procedure: a new tool for measuring readability. Journalism & Mass Communication Quarterly 30 (1953), 415–433.

[41] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, “Session-based recommendations with recurrent neural networks,” *in 4th International Conference on Learning Representations*

[42] Tim Donkers, Benedikt Loepp, and J¨urgen Ziegler. 2017 Sequential User-based Recurrent Neural Network ecommendations. In *Proceedings of RecSys,* ACM, New York, NY, USA, 152–160.

[43] Qiang Cui, Shu Wu, Qiang Liu, Wen Zhong, and Liang Wang. 2018. MV-RNN: A multi-view recurrent neural network for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering 32, 2 (2018),* 317–331.

[44] Reimers, Nils, and Iryna Gurevych. ”Sentence-bert: Sentence embeddings using siamese bert-networks.” *arXiv preprint arXiv:1908.10084 (2019).*

[45] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of WSDM.* 565–573.

[46] Jianling Wang, Kaize Ding, Liangjie Hong, Huan Liu, and James Caverlee. 2020. Next-item Recommendation with Sequential Hypergraphs. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 1101–1110.

[47] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, “Factorizing personalized markov chains for next-basket recommendation,” *in Proceedings of the 19th International Conference on World Wide Web, ser. WWW ’10.* ACM, 2010, p. 811–820.

[48] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of EMNLP. Association for Computational Linguistics,* 1724–1734.

[49] J. Tang and K. Wang, “Personalized top-n sequential recommendation via convolutional sequence embedding,” *in Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining,* ser. WSDM ’18. ACM, 2018, p. 565–573.

[50] Y. Ma, H. Peng, T. Khan, E. Cambria, A. Hussain, Sentic LSTM: a hybrid network for targeted aspect-based sentiment analysis, Cogn. Comput. 10 (4) (2018) 639–650.

[51] D. Ganguly, D. Roy, M. Mitra, G.J.F. Jones, Word embedding based generalized language model for information retrieval, in: R. Baeza-Yates, M. Lalmas, A. Moffat, B.A. Ribeiro-Neto (Eds.), Proceedings of the 38th International ACM SIGIR Conference on Research and Development in *Information Retrieval, ACM, 2015, pp. 795–798.*

[52] L. Dong, F. Wei, M. Zhou, K. Xu, Question answering over freebase with multi-column convolutional neural networks, in: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics* and the *7th International Joint Conference on Natural Language Processing, Association for Computational Linguistics, Beijing, China, 2015, pp. 260–269.*

[53] W.-C. Kang and J. McAuley, “Self-attentive sequential recommendation,” in *2018 IEEE International Conference on Data Mining (ICDM).* IEEE, 2018, pp. 197–206.

[54] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, “Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer,” in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM ’19. ACM, 2019, p. 1441–1450

[55] Juyong Jiang, Jie Zhang, and Kai Zhang. 2020. Cascaded Semantic and Positional Self-Attention Network for Document Classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020.* Association for Computational Linguistics, Online, 669–677.

[56] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 729–732.

[57] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks. In *Proceedings of RecSys. ACM, New York, NY, USA, 130–137*

[58] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M Jose, and Xiangnan He. 2019. A simple convolutional generative network for next item recommendation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 582–590.

[59] Xu Chen, Hongteng Xu, Yongfeng Zhang, Jiaxi Tang, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2018. Sequential Recommendation with User Memory Networks. In *Proceedings of WSDM. ACM, New York, NY, USA, 108–116.*

[60] Kyeongpil Kang, Junwoo Park, Wooyoung Kim, Hojung Choe, and Jaegul Choo. 2019. Recommender system using sequential and global preference via attention mechanism and topic modeling. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1543–1552.

[61] T. Zhang, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, D. Wang, G. Liu, and X. Zhou. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *IJCAI 2019. 4320–4326.*

[62] Elisabeth Fischer, Daniel Zoller, Alexander Dallmann, and Andreas Hotho. 2020. Integrating Keywords into BERT4Rec for Sequential Recommendation. In *KI 2020: Advances in Artificial Intelligence*.

[63] Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. ”Distributed representations of words and phrases and their compositionality.” *Advances in neural information processing systems 26 (2013).*

[64] Le, Quoc, and Tomas Mikolov. ”Distributed representations of sentences and documents.” In *International conference on machine learning, pp. 1188-1196. PMLR, 2014.*

[65] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. ”Glove: Global vectors for word representation.” In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532-1543. 2014.*

[66] Joulin, Armand, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. ”Bag of tricks for efficient text classification.” *arXiv preprint arXiv:1607.01759 (2016).*

[67] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *arXiv preprint arXiv:1802.05365, 2018*

[68] Zihang Dai, Zhilin Yang, Yiming Yang, William W Cohen, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860.*