Advanced Recommendations System using Hierarchical Recurrent Neural Network

1st Md. Anonto Shuvo
Dept. of CSE
BRAC University
Dhaka, Bangladesh
md.anonto.shuvo@g.bracu.ac.bd
ID: 23141036

2nd Md. Sifat Kamal
Dept. of CSE
BRAC University
Dhaka, Bangladesh
md.sifat.kamal@g.bracu.ac.bd
ID: 20101231

3rd Kazi Shahed Mamun
Dept. of CSE
BRAC University
Dhaka, Bangladesh
kazi.shahed.mamun@g.bracu.ac.bd
ID: 20301471

4th Sania Azhmee Bhuiyan
Dept. of CSE
BRAC University
Dhaka, Bangladesh
sania.azhmee.bhuiyan@g.bracu.ac.bd

5th Abid Hossain
Dept. of CSE
BRAC University
Dhaka, Bangladesh
abid.hossain@g.bracu.ac.bd

6th Annajiat Alim Rasel
Dept. of CSE
BRAC University
Dhaka, Bangladesh
annajiat@gmail.com

***Abstract*—Due to the prevalence of complicated data structures and the demand for individualised user experiences, the field of recommendation systems has made tremendous advancements. Yet traditional recommendation models sometimes struggle to understand intricate links in sequential and hierarchical data, which restricts their capacity to provide reliable recommendations. This research suggest proposing an "Advanced Recommendations System using Hierarchical Recurrent Neural Network" (HRNN) as a recommender system. By tackling the problems of scarcity, concerns that start to cool, and shifting user preferences, the suggested HRNN stretches the limits of conventional recommendation systems. Experiments were run on a variety of real-world datasets to assess the performance of the advanced recommendation system using a hierarchical recurrent neural network. Benchmarks are run using traditional recommendation algorithms, emphasising HRNN's in terms of accuracy, recall, and user satisfaction metrics. This approach to recommender systems, harnessing the capabilities of hierarchical and recurrent neural networks to provide advanced, context-aware recommendations. This abstraction of the complexity of data is poised to redefine user experience in various domains, heralding a new era of intelligent and adaptive recommendation systems.**

***Index Terms*—Recommendation systems, Sequence Modeling, HRNN, RNN, Sequential Recommendation**

I. Introduction

Recently, recommendation systems have become popular on many online platforms, especially in the e-commerce space. These systems use user data to suggest products and items that match the user's potential interests and purchasing preferences. Personalizing these recommendation systems for individual users is crucial to their effectiveness. Various approaches have been proposed to improve the accuracy of these systems in predicting user behavior and providing improved suggestions. A recommender system typically predicts a user's unique preferences and learns from past usage patterns to identify the items they are most likely to like. In this paper, we introduced an RNN model - 'Hierarchical Recurrent Neural Network' that will help in advanced recommendations system. About the model, according to the axiom of differentiability and continuity, a function must be continuous at every point within its domain in order for it to be differentiable at a point x_0. Additionally, all of the parameters used in the back propagation. must have continuous values. The situation might not always be the same, thus we might prefer a network made up of neurons that make challenging stochastic decisions about temporal occurrences at various time scales, such as a 0/1 value signifying the end of a video's narrative or the beginning of a text's word or phrase boundaries. These binary outputs also bring sparse representations that might be used as a regularization method, enabling the creation of gating units to determine which portion of the model actually has to be computed for a particular instance. [1] Thus, back propagation fails. In many online systems that use recommendations, interactions between users and systems are organized into sessions. A session is a series of interactions that take place within a particular environment time frame. Recurrent Neural Networks (RNNs) have been recently used for the purpose of sessionbased recommendations [2] outperforming item-based methods by 15% to 30% in terms of ranking metrics. In these cases it is reasonable to assume that the user behavior in past sessions might provide valuable information for providing recommendations in the next session. A novel algorithm based on RNNs that can deal with both cases: (i) session-aware recommenders, when user identifiers are present and propagate information from the previous user session to the next, thus improving the recommendation accuracy, and (ii) session-based recommenders, when there are no past sessions (i.e., no user identifiers). The algorithm is based on a Hierarchical RNN (HRNN) where the hidden state of a lower-level RNN at the end of one user session is passed as an input to a higher-level RNN [3]

II. RELATED WORKS

In this paper, Qu et al. [4] proposed a new RNN-based user behavior prediction model (RNN-BPM) to analyze different behavior sequences within online content consumption systems (e.g., data sparsity/cold-start/temporal context in recommendation systems). The RNN-BPM is based on the RNN language model and combines RNNs (Recurrent Neural Networks) and BPNNs (Backpropagation Neural Networks) to capture time-related and recurrent patterns in behavior. In the proposed RNNCM, RNNs are used to predict behavior by taking into account temporal context and repeating patterns. At the same time, three different types of behavior sequences i.e long-term behavior, short-term behavior and popular behavior are analyzed simultaneously. The authors compute the initial probabilities of probable items that a user would consume next time under various types of user behaviors using RNNs. They then used BPNN to build their BPNNCM model. BPNN calculates the final probability for each item that a user would likely consume next time based on the initial probabilities of those three behavior sequences. The RNN and BPNN models are nested to build the RNN-PBNCM, which predicts a user's next consumption. The experiments were performed on the Last.fm music app dataset and the results showed that the RNN – BPNNCM model was more effective than the baseline models and the single RNN model at predicting user behavior in the online content consumption system.

In this paper, Rosa et al. [5] presented a knowledge-based recommendation system (KBRS) which is designed to monitor the emotional health of users through an online social network (OSN). The system makes use of sentiment analysis and ontology to identify on customers with underlying mental troubles along with melancholy and pressure. If detected, the system sends suitable messages to the consumer primarily based totally on their emotional state.The system makes use of convolutional neural network (CNN) and bidirectional long-term memory (BLSTM) - recurrent neural network (RNN) to categorise sentences with melancholy or pressure content. Test results display better accuracy in detecting customers laid low with melancholy and pressure. The recommended eSM2 Sentiment Index takes under consideration consumer profiles, consisting of age, gender, geographic location, and subject, to offer greater personalised sentiment analysis. The metric turned into advanced primarily based totally on subjective assessments finished with members assessing the sentimental cost of sentences. KBRS structure includes consumer profiles, messaging, melancholy/pressure detection through machine learning, sentiment analysis the use of eSM2, ontology, and recommendation engine. The system turned into examined and in comparison with conventional RS with out sentimental or ontological evaluation. The proposed KBRS carried out extensively better satisfaction rates than conventional RS, demonstrating the effectiveness of the use of sentiment evaluation and personalised ontology.

In this paper Choe et al. [6] presented a new innovative recommendation system that uses the user's time intervals along with the item usage sequence. The authors have created it on a Hierarchical Recurrent Neural Network (HRNN) model. This model consists of two layers. These are short-term which deal with short events and other one is long-term which deals with long-term information over a longer time range. Movielens data sets and Steam data sets were used by them in their research. Their recommended Hi-RNN model was compared to other baseline RNN recommendation methods and Hit@k (a metric that measures how accurately a top-k recommendation is made). Results found that the Hi-RNN model showed better performance when considering longer sequences, suggesting that temporal properties play an important role in recommendation systems (e.g., sequence length, time interval, etc.). The authors suggested that future work might include additional information, such as user trust relationships, review content, etc., to further improve the Hi-RNN model.

In this paper, Liu et al. [7] presented a model called Rating Graph Neural Network (RGNN) that looks to use user reviews to increase the overall accuracy of the recommendation system. For each user/item, by taking the relationship between the words in the review, a rating graph was built by the the model. For bringing out the hierarchical semantic representations of the revision graph, the proposed model uses a type-aware graph attention mechanism and custom graph clustering operators. The model was compared with other recommendation models and the results showed that in case of mean squared error (MSE), RGNN performs better demonstrating its effectiveness in capturing important patterns in user reviews to have better suggestions.

In the paper Zhang et al. [8] presented Sema which is a framework for recommendation. It was build to increase the accuracy of recommendations by using deep learning techniques along with semantic meaning and temporal dynamics. For modeling temporal dynamics and semantic meaning, symmetric RNN and hierarchical was used by the framework. The framework was experimented by the authors Amazon dataset and Yelp dataset. The results after comparing SEMA with other recommended methods, demonstrate that in case of root mean square error (RMSE) and mean absolute error (MAE), SEMA shows better improvements. Overall, SEMA presents a comprehensive recommendation framework that effectively combines semantic analysis, time dynamics, and deep learning to improve personalized recommendations.

In the paper, Xu et al. [9] presented an innovative process called Slander Detection Recommendation System (SDRS). In referral systems, the authors wanted to identifying users who intentionally give misleading reviews and ratings to manipulate the system. A framework is being used by the SDRS to combines sentiment analysis, user behavior analysis, and recommendation techniques. A framework was proposed by the authors which is made up of 4 main modules. These modules are - sentiment analysis to evaluate, integrate words, detect and recommend slanderous users. In this research paper, goal of the authors was to identify slanderous users based on their ratings and reviews and remove their influence. Datasets like Amazon, Yelp, Taobao and Jingdong were used by the authors in these experiment. SDRS was compared with various baseline RNN models. The results showed SDRS is better than other sentiment and recommendation analysis methods in identifying these users and making accurate recommendations.

In this paper, Donkers et al. [10] presented an innovative way to provide personalized recommendations using recurrent neural networks (RNNs). The authors focus is on the recommendation systems domain. They have used 3 variations of user-based RNNs. These variations are linear user integration, edited linear user integration, and attention user integration. These RNN variants were used to compare with the baseline recommendation models. MovieLens and LastFM dataset were used for the experiment. MRR@20 (Mean Reciprocal Rating) and Recall@20 were used as the rating metrics. The result shows that user-based RNN consistently performs better than the other baseline models. Compared to other standard RNNs, user-attention-integrated variant improve the overall accuracy of recommendations.

In the paper, Ko et al. [11] proposed an approach that uses online user's activity sequence for the recommendation system. For capturing the temporal patterns in user behavior, based on recurrent neural networks (RNNs) the authors came up with a sequence model. In this proposed approach, it looks at the user action sequences and aims to predict future actions.Model traits consist of flexibility for exclusive obligations, the capacity to control long-time period dependencies, and a stability among personalization and collaboration aspects. The authors have evaluated this approach on music recommendation and mobile prediction tasks using real data sets, demonstrating its superiority over different methods.

In the paper, Song et al. [12] presented an innovative model for sequential recommendation systems called HCRNN (Hierarchical Contextual Recurrent Neural Network). The authors have build this model for solving the problems like the long-term dependencies and changing user behavior over time.They have used four different versions of HCRNN which are - HCRNN-1, HCRNN-2, HCRNN-3 and HCRNN-3+Bi. CiteULike, LastFM and MovieLens datasets were used by the authors for the experiment. The HCRNN variants were compared with the other RNN baseline models and the result shows that HCRNN in terms of recall and average reciprocal ratings, outperforms the baseline models . HCRNN-3+Bi i.e bichannel attentions perform better than other versions of HCRNN. The paper also provides qualitative insights, such as context integration, portal analysis, and case studies, to better understand model behavior and performance.

In the paper, Villatel et al. [13] addressed that for predicting user preferences over short and long time periods an effective Recurrent Neural Network (RNN) model is necessary for recommendation systems. The authors proposed enhancements to RNN models and conducted experiments to improve their performance. The RNN models were compared against several baselines models like POP, Item- KNN, CoEvent MF. The result of the research shows that RNN model outperforms the baselines. The authors have suggested that right configuration for the RNN model is the stacked GRU, which includes layer normalization, a tied item embedding, and a tied-item matrix.

III. HIERARCHICAL RECURRENT NEURAL NETWORK

Hierarchical Recurrent Neural Network (HRNN) is a class of stacked RNN models designed to model hierarchical structures in sequential data (text, video streams, speech, programs, etc.). Our HRNN model is based on RNN in the following way. (i) Adding a GRU layer to model information across user sessions and track the evolution of user interests over time. (ii) Using a powerful user-parallel mini-batch mechanism for efficient training.

A. Architecture

There are two GRU layers in HRNN in contrast to one layer in RNN [3]: (i) Session-level which is similar to the RNN's GRU layer. (ii) User-level GRU. The session-level GRU models the user activity within sessions and generates recommendations. The user-level GRU models the evolution of the user across sessions and provides personalization capabilities to the session-level GRU by initializing its hidden state. In this way, the information relative to the preferences expressed by the user in the previous sessions is transferred to the session-level GRU.

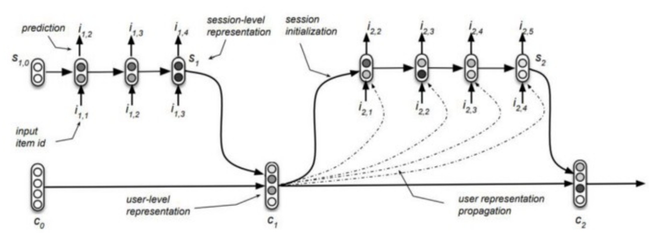

Fig. 1. Graphical representation of the proposed Hierarchical RNN model for personalized session-based recommendation

The following two different HRNN settings are considered depending on whether the user representation cm is considered:

• HRNN Init, in which cm is used only to initialize the representation of the next session.

• HRNN All, in which cm is used for initialization and propagated in input at each step of the next session.

In HRNN Init, the session-level GRU can exploit the historical preferences along with the session-level dynamics of the user interest. HRNN All instead enforces the usage of the user representation at session-level at the expense of a slightly greater model complexity. As it will be seen, this can lead to substantially different results depending on the recommendation scenario.

B. Training

There are various methods for training such networks: (a) Gradient estimation techniques, that train binary neurons as an integral component of the network, [1] (b) Policy gradientbased reinforcement learning methods [14] that bypass the necessity for binary neurons by instead creating a map of stateaction pairs to decide on what action to take on observing a particular input. These can be conducted by two classes of HRNN namely Hierarchical Multiscale Recurrent Neural Networks (HM-RNN) & Hierarchically Structured LSTMs (HS-LSTMs). Moreover, (c) With user-parallel mini-batches [15] we can train HRNNs efficiently over users having different number of sessions and sessions of different length. Moreover, this mechanism allows to sample negative items in a user-independent fashion, hence reducing the chances of 'contamination' of the negative samples with actual positive items. The sampling procedure is still popularity-based, since the likelihood for an item to appear in the mini-batch is proportional to its popularity

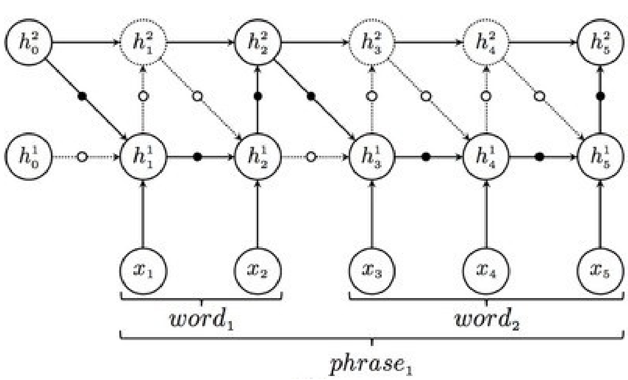

Fig. 2. Hierarchical Multiscale Recurrent Neural Networks (HM-RNN)

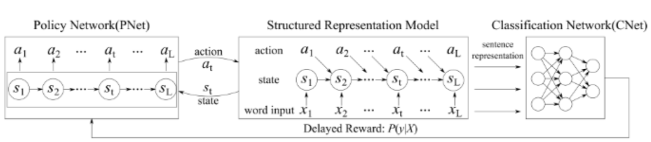

Fig. 3. Hierarchically Structured LSTMs (HS-LSTMs)

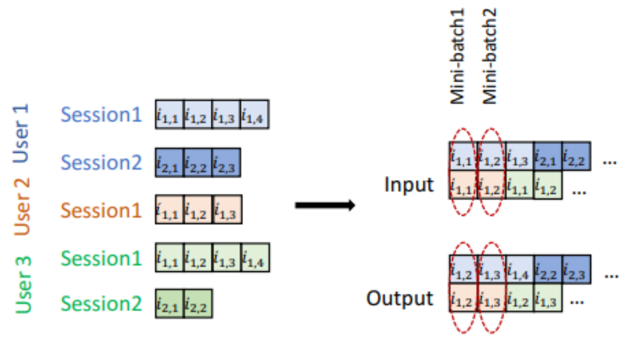

Fig. 4. User-parallel mini-batches for mini-batch size

IV. Experiments

A. Dataset

For the research, 4 publicly available real-world datasets are used which are Movielens1M, CiteULike, Steam and LastFM. Two types of RNN models are going to compare with the baseline models and also use the dataset for each model separately. The two types of RNN models are Hierarchical Recurrent Neural Network (Hi-RNN) and Hierarchical Context Enabled Recurrent Neural Network (HC-RNN). Both of these are variations of recurrent neural networks that are designed to capture hierarchical structures and dependencies in sequential data.

For HRNN, Movielens1M and Steam data-set are going to be used. In the Movielens1M dataset, there are 6040 users and 3416 items. The average sequence length per user is 163.5. [6] Moreover, Steam is the game platform dataset containing 334730 users and 13047 items. The average sequence length per user is 12.26. [6] For test dataset, most recent item of each user was used. For verification, the 2nd most recent item was used and the rest for training.[6] From both the datasets, users with less than 5 events was filtered out. On the other hand, for HC-RNN, Movielens1M, CiteULike and LastFM data-sets are chosen. Sequence having length less than 10 are removed. Also item which are less than 50/50/25 each in the 3 different dataset. Cross-validation was performed by selecting 10% of the randomly chosen training set as the validation set. [12] The dataset has five part. These are sequence(train), sequence(test), clicks, items, average length. Sequence(train) has 38,724 CiteULike data, 73,420 LastFM data and 136,233 Movielens1M data. Sequence(test) has 9,140 CiteULike data, 17,829 LastFM data and 34,682 Movielens1M data. Click has 1,163,813 CiteULike data, 4,575,159 LastFM data and 5,041,882 Movielens1M data. Items has 1980 CiteULike data, 5778 LastFM data and 930 Movielens1M data. Average length of has CiteULike data is 24.31, LastFM data is 50.14 and Movielens1M data is 29.50. [12]

B. Baselines

Baseline Models are some common models that is used in recommendation system. We are going compare these baseline models with the proposed models. Some baseline models are-

- **POP:** It is one of the most popular models. This model makes recommendations only based on user popularity. It does not consider any user-specific information. Then it provides suggestions of the most popular items to the user. With the description, it can sound very simple but often it works as an effective baseline.

- **SPOP (Session-Based Most Popular):** It is an extension of the POP model that is better suited for session-based recommendations since this model focuses on recommending popular items by using the user's current session or recent browsing history..

- **ITEM-KNN (Item-Based k-Nearest Neighbors):** It is a collaborative filtering model that recommends with similarity of items that the user has previously interacted with. It basically calculates the similarities and recommends items that are most like the ones the user has interacted with.

- **BPR (Bayesian Personalized Ranking):** It is a matrix factorization model that maximizes the ranking of items in a personalized way. For recommending items it uses pairwise ranking. It helps the model learn user and item latent factors, aiming to rank items that the user has interacted with much more than those they have not.

- **BPR-MF (BPR Matrix Factorization):** It is an extension of the BPR model. For personalized recommendations, it specifically uses matrix factorization techniques to learn user and item embeddings.

- **GRU4REC(Gated Recurrent Unit for Recommendations):** It is a session-based recommendation model. It uses Gated Recurrent Units (GRUs) to determine the sequential patterns in user interactions within a session and recommend items.

- **GRU4Rec+ (Enhanced GRU4Rec):** It is an improved version of the GRU4REC model. It follows some additional techniques to enhance recommendation performance such as negative sampling and adaptive learning rates.

- **LSTM4REC (Long Short-Term Memory for Recommendations):** It uses Long Short-Term Memory(LSTM) networks to determine sequential behaviors and make personalized recommendations within sessions similar to GRU4REC.

- **NARM (Neural Attentive Session-based Model):** It is a session-based recommendation model that employs neural attention mechanisms to focus on important items in a user's session history while making recommendations.

- **STAMP (Session-based Temporal Attention Model with Personalization):** It is also a session-based recommendation model that considers both sequential patterns and user preferences and integrates temporal attention mechanisms with personalization.

- **FMC (Factorization Machines for Context-Aware Recommendation):** This model takes advantage of factorization machines. In order to capture interactions between user preferences and contextual features, it enables context-aware recommendations.

- **FPMC (Factorized Personalized Markov Chains):** This model is different from the models that so far mentioned. This model works as a hybrid model. For improved recommendations, it integrates matrix factorization and Markov chain techniques in order to capture both user-item interactions and sequential dependencies.

We are going to compare Hi-RNN with POP, BPR, FMC, FPMC, GRU4Rec and GRU4Rec+ baseline models. [6] On the other hand, HC-RNN with POP, SPOP, Item-KNN, BPR-MF, GRU4Rec, LSTM4REC, NARM AND STAMP. [12]

C. Results

*1) **Hierarchical Recurrent Neural Network**:* For comparing the measurement between the base models and Hierarchical Recurrent Neural Network, Hit@k was used instead of Recall as it is more suitable to top-N recommendation system evaluation. [6]

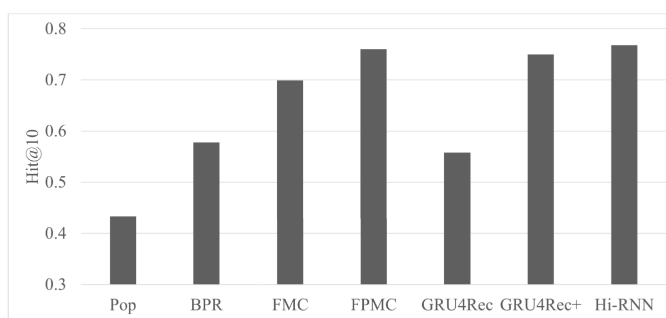

Fig. 5. Hit@10 for Movielens1M dataset using the baseline methods and Hi-RNN

In the figure 5 and 6 the results of the comparison of the base models and Hierarchical Recurrent Neural Network on both the datasets has been shown. From both the figure, we can see that Hi-RNN, which takes into account long-term

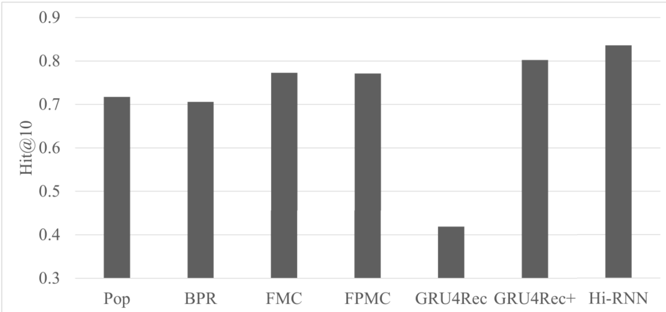

Fig. 6. Hit@10 for Steam dataset using the baseline methods and Hi-RNN

user history, are significantly higher than other models. The Hit@10 of the existing methods was not more than 76%, where Hi-RNN was 78%. It differs by more than 10% from Pop and BPR model, which do not use temporal properties. GRU4Rec and GRU4Rec+ model, which consider the properties of sequence data using RNN, perform better than Pop and BPR model, but the performance of the Hi-RNN model, which considers time intervals, is better. The performance difference between Pop and BPR model was not significant in the Steam dataset compared to the Movienlens1M dataset. Because the average sequence length is longer in the Movielens1M dataset than in the Steam dataset, we suspect that the importance of the temporal property increases as the sequence gets longer, leading to this difference.

*2) **Hierarchical Context Enabled Recurrent Neural Network**:* Song [12] designed a gate structure that reflected the interests drift assumptions through hierarchical contexts. To reflect this interest drift assumption, they modified the reset gate to HCRNN-2. In addition, HCRNN-3 has drift gate whose rest gate is based upon the previous temporary context h_{t-1} and aslo has local context, and current item embedding together. Their hierarchical context also results in changes in attention when considering HCRNN cell structure. To exploit the hierarchical contexts of HCRNN,they implemented the complementary bi-channel attention as the local context attention and the temporary context attention. [12]

In figure 7, 8 and 9 shows the performance of the baseline models and HCRNN. [12] Two recall measurements at K(R@K) and mean reciprocal ranking at K(M@K) where used which are widely used in Sequentially recommendation. HCRNN's experiment includes variantion of the ablation study HCRNN-1, HCRNN-2, HCRNN-3, and HCRNN-3 with bichannel attentions (HCRNN-3+Bi). Quantitative evaluation indicates the performance improvements of variation of HCRNN for all data and metrics. Bi-channel HCRNN-3 shows the best performance. Moreover, the performance improvement. Additionally, HCRNN performs better results compared to RNN-based recommendations i.e NARM, GRU4REC, and LSTM4REC. it's also seems that HCRNN-3

	CiteULike			
	R@3	R@20	M@3	M@20
POP	1.44	5.78	0.92	1.44
S-POP	1.26	4.99	0.79	1.23
Item-KNN	0.00	6.90	0.00	4.79
BPR-MF	0.49	3.15	0.27	0.60
LSTM4REC	7.07	23.33	4.93	6.82
GRU4REC	8.37	24.19	5.98	7.86
NARM	7.81	24.82	5.40	7.41
STAMP	5.09	21.93	3.25	5.22
HCRNN- 1	8.60	25.36	6.18	8.16
HCRNN- 2	8.83	25.10	6.41^{*}	8.38^{*}
HCRNN- 3	9.21^{*}	25.42^{*}	6.65^{*}	8.61^{*}
HCRNN-3 + Bi	$\mathbf{9.33}^{*}$	$\mathbf{25.81}^{*}$	$\mathbf{6.74}^{*}$	$\mathbf{8.70}^{*}$
Improvement(%)	11.47	3.99	12.71	10.69

Fig. 7. Performance evaluation of baseline Models with HCRNN on CiteULike dataset.

	LastFM			
	R@3	R@20	M@3	M@20
POP	0.37	1.99	0.34	0.51
S-POP	0.87	3.65	0.55	0.87
Item-KNN	0.00	11.59	0.00	8.00
BPR-MF	0.82	2.15	0.59	0.73
LSTM4REC	15.29	24.75	12.68	13.95
GRU4REC	18.29	26.46	15.85	16.95
NARM	18.30	33.60	13.12	15.25
STAMP	9.29	19.84	6.62	8.01
HCRNN- 1	20.67^{*}	34.40^{*}	15.77	17.68^{*}
HCRNN- 2	20.78^{*}	34.14^{*}	16.20	18.08^{*}
HCRNN- 3	21.39^{*}	34.72^{*}	16.66^{*}	18.52^{*}
HCRNN-3 + Bi	$\mathbf{21.90}^{*}$	$\mathbf{34.80}^{*}$	$\mathbf{17.33}^{*}$	$\mathbf{19.12}^{*}$
Improvement(%)	19.67	3.57	9.34	12.80

Fig. 8. Performance evaluation of baseline Models with HCRNN on LastFM dataset.

shows better results than HCRNN-1 and HCRNN-2. As HCRNN-3+Bi is showing the best reuslt, it can be said that paying attention to hierarchical context can improve performance experimentally. [12]

V. Conclusion

Hierarchical Recurrent Neural Network (HRNN) provides a way to seamlessly transfer the acquired knowledge about the long-term dynamics of user interest to the session level, providing personalized, session-based recommendations to returning customers. The proposed HRNN model outperforms both state-of-the-art session-based his RNN and other basic personalization strategies for session-based recommendations on four disparate real-world datasets. In particular, it is found that the simpler approach, where only the session-level repre-

	MovieLens			
	R@3	R@20	M@3	M@20
POP	2.43	12.51	1.54	2.65
S-POP	2.27	12.23	1.42	2.52
Item-KNN	0.00	6.32	0.00	4.28
BPR-MF	1.69	8.93	1.07	1.91
LSTM4REC	8.52	32.80	5.63	8.45
GRU4REC	8.50	32.74	5.60	8.42
NARM	<u>9.14</u>	<u>33.42</u>	<u>6.09</u>	<u>8.93</u>
STAMP	3.95	20.52	2.65	4.47
HCRNN- 1	9.23	33.78*	6.13	9.00
HCRNN- 2	9.22	33.76*	6.14	9.01
HCRNN- 3	9.38*	33.67*	6.23*	9.08*
HCRNN-3 + Bi	**9.53***	**33.83***	**6.38***	**9.21***
Improvement(%)	4.27	1.23	4.76	3.14

Fig. 9. Performance evaluation of baseline Models with HCRNN on Mocielens1M dataset.

sentation is initialized with the user's evolving representation (HRNN), gives the best results. After delving into the dynamics of session-based RNN models within and between sessions, it seems to provide extensive evidence for the superiority of the proposed HRNN approach, and provide new state-of-the-art results for personalized session-based recommendations.

REFERENCES

[1] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical multiscale recurrent neural networks," *arXiv preprint arXiv:1609.01704*, 2016.

[2] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," *arXiv preprint arXiv:1511.06939*, 2015.

[3] M. Quadrana, A. Karatzoglou, B. Hidasi, and P. Cremonesi, "Personalizing session-based recommendations with hierarchical recurrent neural networks," in *proceedings of the Eleventh ACM Conference on Recommender Systems*, 2017, pp. 130–137.

[4] Z. Qu, S. Zheng, X. Wang, X. Song, B. Li, and X. Song, "Converged recommendation system based on rnn and bp neural networks," in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2018, pp. 308–313. DOI: 10.1109/BigComp.2018.00052.

[5] R. L. Rosa, G. M. Schwartz, W. V. Ruggiero, and D. Z. Rodríguez, "A knowledge-based recommendation system that includes sentiment analysis and deep learning," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2124–2135, 2019. DOI: 10.1109/TII.2018.2867174.

[6] B. Choe, T. Kang, and K. Jung, "Recommendation system with hierarchical recurrent neural network for long-term time series," *IEEE Access*, vol. 9, pp. 72 033–72 039, 2021. DOI: 10.1109/ACCESS.2021.3079922.

[7] Y. Liu, S. Yang, Y. Zhang, C. Miao, Z. Nie, and J. Zhang, "Learning hierarchical review graph representations for recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 658–671, 2023. DOI: 10.1109/TKDE.2021.3075052.

[8] J.-D. Zhang and C.-Y. Chow, "Sema: Deeply learning semantic meanings and temporal dynamics for recommendations," *IEEE Access*, vol. 6, pp. 54 106–54 116, 2018. DOI: 10.1109/ACCESS.2018.2871970.

[9] Y. Xu, Y. Yang, J. Han, E. Wang, J. Ming, and H. Xiong, "Slanderous user detection with modified recurrent neural networks in recommender system," *Information Sciences*, vol. 505, pp. 265–281, 2019.

[10] T. Donkers, B. Loepp, and J. Ziegler, "Sequential user-based recurrent neural network recommendations," in *Proceedings of the eleventh ACM conference on recommender systems*, 2017, pp. 152–160.

[11] Y.-J. Ko, L. Maystre, and M. Grossglauser, "Collaborative recurrent neural networks for dynamic recommender systems," in *Asian Conference on Machine Learning*, PMLR, 2016, pp. 366–381.

[12] K. Song, M. Ji, S. Park, and I.-C. Moon, "Hierarchical context enabled recurrent neural network for recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 4983–4991.

[13] K. Villatel, E. Smirnova, J. Mary, and P. Preux, "Recurrent neural networks for long and short-term sequential recommendation," *arXiv preprint arXiv:1807.09142*, 2018.

[14] T. Zhang, M. Huang, and L. Zhao, "Learning structured representation for text classification via reinforcement learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[15] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel recurrent neural network architectures for feature-rich session-based recommendations," in *Proceedings of the 10th ACM conference on recommender systems*, 2016, pp. 241–248.