



Webpage Recommendation System for Healthcare issue Using Reinforcement Learning Algorithm

S. Anusuya*, K. Anandapadmanabhan†

Abstract

World Wide Web is an immense repository of information, but the sheer volume of data makes it challenging to locate specific information easily. Recommendation Systems (RS) are designed to alleviate this issue by identifying similar items and customers based on their behaviour, subsequently suggesting items tailored to individual preferences.

The RS involves RL (Reinforcement Learning)-based RS approaches used to overcome existing limitations by fusing RL with self-supervised sequential learning. However, these approaches often suffer from biases in estimating Q-values, skewing based on positive values because the absence of negative reward signals. It exclusively depends on positive signals to create a model with a positivity bias using clicks and views. To rectify this, a novel strategy incorporating Supervised Negative Q-learning, Supervised Advantage Actor-Critic has been introduced.

This work proposes a solution to enhance stability by merging RL with sequential modelling, then integrating contrastive-based objectives, and negative sampling methods with contrastive learning and conservative Q-learning. Together, these components improve performance and increased stability. This proposed method validates the effectiveness of Empirical results on real-world datasets.

Keywords: Recommender system, Q-learning, Contrastive learning, Self-supervised Learning.

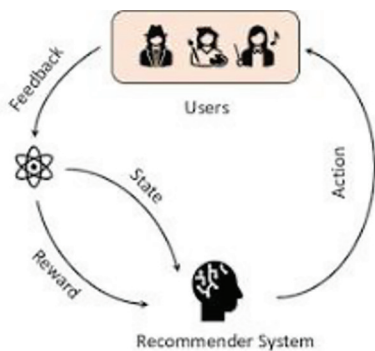
1. Introduction

A recommender system (RS) is a type of computer program that gives customized direction to the end users. Recommender systems are mostly used in all areas like Industries, e-commerce, social media, streaming services

* Department of Computer Science, Sri Vasavi College, Self Finance Wing, Erode
Anuciaa@gmail.com; kapn0305@gmail.com

and etc. to increase the performance and to have a unique experience for users. The System used to maintain the timing information of our orders, and also the interaction of users that occurs frequently in chronological order. The existing Sequential recommendation methods uses the Recurrent Neural Networks (RNNs) (Wu et al., 2017), but it is difficult to capture the convolution of nonlinear sequences of user behaviour. The latest updates in recommendation systems have encircled the Modernisers (Vaswani et al., 2017), a high recommended architecture was initially designed mainly for natural language processing. The Modernizers excel to represent the interdependent (Zhou et al., 2020) in user behaviour that ultimately increases the accuracy of Recommendation System. The large language models like GPT-4 (OpenAI, 2023) also had proven that their mastery in RS, even in conversant situations (Li et al., 2023). But the sequence modelling is important in RS, to optimize and deliver personalized recommendation with high quality of data.

The Reinforcement Learning is an upcoming structure to develop the RS. This proposal hits a balance between finding new recommendations and explorer the known preferences that automatically increase user satisfaction in the long-term. (Chen, 2021). However, merging the RL into RS will get various challenges like, training instability, mainly for larger and complex models (Tang et al., 2023). Even though Researchers are actively working to address these issues, and focus to remove all the potential of RL-powered recommender systems. In above summary, RS has progress significantly, by utilizing the latest models such as Transformers and RL to address more specific and adjustable recommendations that helps to improve overall experience of user.



The important contributions of works are listed below:

- This study deals with strategies of integrating sequential augmentation with the investigation of contrastive learning objective, to authenticate the efficacy across world and to demanding real-world datasets.

- It put forwards more attentive objective to reduce uncertainties during reinforcement learning (RL) training by identifying intrinsic challenges come from negative action sampling.
- The Sequential Negative Q-Network (SNQN) is proposed mainly for negative sampling within the RL training of recommendation system. SA2C method determines the superiority to get normalized Q-values through positive actions.
- This strategy identifies the importance of monitoring RL-based training model to progress and to identify unreliability during online deployment.

The remaining content of this paper is organized as follows: Section 2 shares reviews of related State & arts in, Section 3 shares the detailed description of Proposed Methodology, previous method is tested & verified with other methods for further discussions in Section 4, overall paper is conclusion is described in Section 5.

2. Related works

Recommendation systems have progressed substantially with the incorporation of latest techniques and models to give more accurate and personalized solutions. The importance of following this is to combine the user feedback into recommendation algorithms.

S. No.	Description	Author & Year	Implementation
1.	Deep Q-Network (DQN)	Zhao et al. 2018	This approach analyses both positive and negative feedback to fine-tune recommendations.
2.	Recommendation system using DRL Deep Reinforcement Learning	Gao et al., 2019	It gives user interactive advice systems, by combining Convolutional Neural Networks, Generative Adversarial Networks
3.	Direct feedback scheme	Lei and Li (2019)	The DQN method that take on matrix factorization to create individualized states and by considering biases in Q-value analysis.
4.	Markov Chain (MC)-based techniques	Markov & McAuley (2016)	These methods account for longer interaction sequences
5.	sequential recommendation methods	Zhou et al. (2020)	Focus on predicting next-item prediction in terms of learning task gives more effective and adaptable recommendation systems.
6.	Recommendation system	Wu et al. (2021)	To improve recommendation quality presented a versatile framework contrastive learning is used.
7.	contrastive learning in recommendation	Qiu et al. (2022)	To enhance user representation and to develop contrastive signals.

3. Methodology

3.1 Supervised Negative Q-learning (SNQN)

The supervised training loss is the cross-entropy over the classification distribution with an input from user-item sequence $x_{i:t}$ and recommendation model ($\bullet$),

$$L_s = - \sum_{i=1}^n y_i \log(p_i), \text{ where } p_i = \frac{e^{y_i}}{\sum_{i'=1}^n e^{y_{i'}}} \quad (1)$$

In the given context, Y_i is indicator function, equal to 1 when user interacts with i -th item in further elements else it is 0. Main aspect is advantageous in a recommender system setup where the primary goal is ranking items in the top- k positions. Given that ($\bullet$) encodes data sequence into a latent state, s_t , this state is utilized for RL training. Building upon the shared base model ($\bullet$), an additional output layer is formulated to map the state into Q-values, facilitating RL-based recommendation strategies.

$$Q(s_t, a_t) = \delta(s_t h_t^T + b) = \delta(G(x_{i:t} h_t^T + b)) \quad (2)$$

The activation function δ is applied to the Q-learning output layer's parameters, h_t and b . Algorithm 1 outlines the SNQN training process. However, Q values derived solely from observed positive actions are inadequate for generating recommendations. Supervised Negative Q-learning architecture is shown in Figure 1.

Algorithm 1 Training Procedure of SNQN

Input: X is the user-item interaction sequence, $G(\cdot)$ is recommendation model, reinforcement head $Q(\cdot)$, Supervised head $f(\cdot)$, pre-defined reward function $r(s,a)$

Output: all parameters in the learning space θ

All the trainable parameters are initialized

Create $G'(\cdot)$ and $Q(\cdot)$ as a copies of $G(\cdot)$ and $Q(\cdot)$ respectively.

repeat

Draw a mini-batch of $(x_{1:t}, a_t^*)$ from X

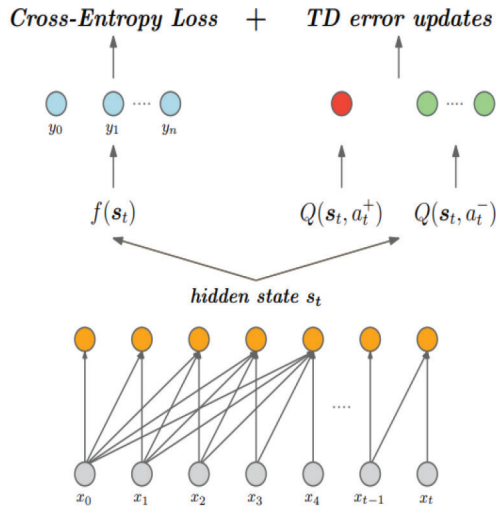


Figure 1: Architecture of SNQN

$$L_q = L_p \text{ (positive TD error)} + L_n \text{ (negative TD error)} \quad (3)$$

Draw negative actions set N_t for $x_{1:t}$

$$S_t = G(x_{1:t}), S'_t = G'(x_{1:t})$$

$$S_{t+1} = G(x_{1:t+1}), S'_{t+1} = G'(x_{1:t+1})$$

Create random variable $z \in (0,1)$ uniformly

If $z \leq 0.5$ then

$$a_*^+ = \operatorname{argmax}_a Q(S_{t+1}, a), a_*^- = \operatorname{argmax}_a Q(S_t, a)$$

$$L_p = (r(S_t, a_t^+) + \gamma Q(S'_{t+1}, a_*^+) - Q(S_t, a_t^+))^2$$

$$L_n = \sum_{a_t \in N_t} (r(S_t, a_t^-) + \gamma Q(S'_{t+1}, a_*^-) - Q(S_t, a_t^-))^2$$

Calculate L_s and $L_{\text{snqn}} = L_s + L_p + L_n$

Perform updates by $\nabla_{\theta} L_{\text{snqn}}$

Else

$$a_*^+ = \operatorname{argmax}_a Q'(S_{t+1}, a), a_*^- = \operatorname{argmax}_a Q'(S_t, a)$$

$$L_p = (r(S_t, a_t^+) + \gamma Q(S_{t+1}, a_*^+) - Q'(S_t, a_t^+))^2$$

$$L_n = \sum_{a_t \in N_t} (r(S_t, a_t^-) + \gamma Q(S_t, a_*^-) - Q'(S_t, a_t^-))^2$$

Calculate L_s and $L_{sqn} = L_{s+} L_{p+} L_n$

Perform updates by $\nabla_{\theta} L_{sqn}$

Here, a_t^+ and a_t^- represent positive, negative actions at timestamp t , negative TD error, is highest process occurs in (s_t, a') instead of $Q(s_{t+1}, a')$, assuming negative actions don't impact the user state.

$$L_{sqn} = L_s + L_q \quad (4)$$

Double Q-learning, inspired by Hasselt (2010), enhances learning stability by training two copies of model parameters alternately. A constant reward r_n is assigned to negative actions.

3.2 Supervised Advantage Actor-Critic (SA2C)

SA2C computes the merits of actions, effectively by using normalized Q-values, based on sampled actions, and this advantage as estimate critic to re-weight the actor. Actor-Critic (AC) methods are widely successful in the field of RL. Figure 2 provides a visual representation of SA2C's architecture.

endif

until coverage

return all parameters in θ

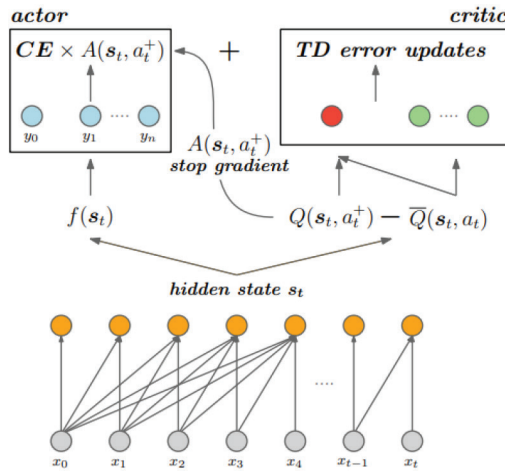


Figure 2: Architecture of SA2C

The average of Q-values will be said as:

$$\bar{Q}(s_t, a) = \frac{\sum_{a' \in a_t^+ \cap N_t} Q(s_t, a')}{|N_t| + 1} \quad (5)$$

An observed (positive) action benefit can be expressed in terms of:

$$A(s_t, a_t^+) = Q(s_t, a_t^+) - \bar{Q}(s_t, a) \quad (6)$$

In SA2C, positive actions with higher advantages over the average are assigned greater weight in the supervised training process, while the opposite applies to those with lower advantages. To bolster stability, the gradient flow is halted, and Q values are fixed when calculating average and advantage. Subsequently, actor, critic is combining trained. Training loss of SA2C is calculated as:

$$L_{sa2c} = L_a + L_q, \text{ where } L_a = L_s \cdot A(s_t, a_t^+) \quad (7)$$

During the training process, Q-value learning can exhibit instability, especially in the early stages, as noted by Parisotto et al. (2019). To address this, SNQN pre-trained the model for first T training steps. Once Q-values stabilize, the advantage is used to re-weight actor and perform updates following defined architecture. Advantage values act as normalized Q-values, addressing bias from negative action overestimation. They are combined with propensity scores for off-policy correction in the recommendation phase.

3.3 Recommendation phase

Propensity scoring, is a statistical technique commonly applied in observational studies, which used to estimate intervention effects while considering covariates predicting treatment receipt. In RL, an action's propensity score typically specifies the probability of action opted by the behaviour method, as demonstrated by Chen et al. (2021). The system's flow is depicted in Figure 3.

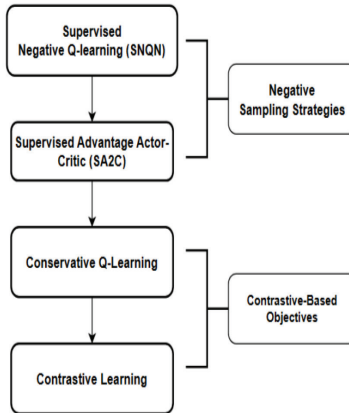


Figure 3: Flow of Proposed System

3.4 Conservative Q-Learning

Conservative Q-Learning (CQL) (Kumar et al., 2020) mitigates Q-learning's overestimation problem. It employs a conservative value function, estimating the optimal Q-function. This function returns the minimum value for the current calculation and the maximum recorded value for a specific scenario-decision pair. It is accomplished by minimizing a defined objective.

$$L_{CQL}(\theta) = E_{(s,a,r,s') \sim D} \left[\left(Q_{\theta}(s,a) - r - \gamma E_{a' \sim \pi_{\phi}(a'|s')} [Q_{\theta'}(s',a')] \right)^2 \right] \\ + \alpha E_{s \sim D} \left[\log \sum_a \exp Q_{\theta}(s,a) - E_{a \sim \pi_{\beta}(a|s)} [Q(s,a)] \right]$$

In this context, D denotes the unchanging dataset, θ and θ' are Q function which target network parameters, ϕ specifies policy parameters, γ stands the discount factor, and α serves as temperature parameter regulating balance between Q-function minimization and conservative regularization.

3.5 Contrastive Learning

InfoNCE (Noise Contrastive Estimation), as proposed by van den Oord et al. in 2018, serves as a crucial loss function in contrastive learning, facilitating the acquisition of valuable representations. In this framework, question sentences represent by Q , context signified by C , context sentences are represented by c_i except for those deceptive sentence with answers (A), and V. Pairs of positive samples, sharing similar colours, emerge via dropout. When encoding adversarial samples, the contrastive learning module effectively separates answer sentences from misleading ones. Objective is calculated using the collection of negative samples ($y_{j,k}$) and positive pairs (x_j, y_j), , enhancing representation learning through contrastive principles.

$$L_{InfoNCE} = - \frac{1}{M} \sum_{j=1}^M \log \frac{\exp(f(x_j, y_j))}{\exp(f(x_j, y_j)) + \sum_{k=1}^K \exp(f(x_j, y_{j,k}))} \quad (8)$$

Where M represents count positive pairs, and K denotes total negative samples associated with each positive pair. The similarity function, $f(x_j, y_j)$ gauges the likeness between x_j and y_j , while $f(x_j, y_{j,k})$ quantifies similarity between x_j and a negative sample, $y_{j,k}$. InfoNCE helps to maximize the similarity between the positive pairs and minimize it for negative pairs, fostering the acquisition of valuable representations. Combining this method with contrastive learning significantly enhances model performance, especially in situations where real-time interaction is costly or infeasible, such as in recommender systems using static datasets.

4. Result and Discussion

4.1 Dataset

This topic discusses the experiment data performed on two datasets to find the output by using this method. The exploitation of electronic data facilitated the extraction of valuable information. Working with electronic medical data from patients presented a notable challenge. The proposed model was simulated by using two distinct diabetes disease datasets. One dataset, the "PIMA INDIAN Diabetes Database," was sourced from an Indian-based health organization; another was obtained electronically from "Hospital Frankfurt Germany". These datasets, retrieved from Kaggle, comprise 2000 cases for the Frankfurt dataset and 768 cases for the PIMA dataset, each containing nine attributes, primarily focused on the results.

4.2 Baselines

There are three baseline methods were used to evaluate the performance. Such as

SASRec (Wang & Julian, 2018) is a self-attention mechanism to predict the next item by identifying relevant items.

SNQN and SA2C (Xin et al., 2022) are baseline models. SNQN employs negative sampling, while SA2C incorporates calculation to recalibrate Q-values.

SASRec_AC (Xin et al., 2020) is a self-supervised reinforcement learning model for sequential recommendation tasks. All models utilize the SASRec model as the base to operate within an actor-critic framework.

4.3 Evaluation Protocols

To evaluate recommendation system's performance, Precision, Recall, and F1 score were utilized. Precision is to minimize false positives and the cost of making incorrect positive predictions is high. Recall is to capture as many true positives as possible and the cost of missing positive cases is high. F1 Score is to balance between precision and recall, and there is an uneven class distribution or when false positives and false negatives have different costs.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP represents the count of correctly diagnosed patients, FP stands for the count of cases wrongly classified as patients, TN denotes the count of correctly recognized healthy individuals, and FN signifies the count of cases erroneously categorized as healthy.

4.4 Tabular Results

For Experimenting this proposed method, initially data from sensors was collected and transmitted through the Internet of Medical Things (IoMT) to the feature database. Table 1 and 2 present performance comparisons of various models on Hospital Frankfurt Germany diabetes dataset, Pima Indians diabetes dataset were employed to train the diabetes prediction model. Cross-validation, following the data split suggested in (Xin et al., 2022), was used to assess the model’s efficiency.

Table 1 performance comparisons of Hospital Frankfurt Germany diabetes dataset

Methods	Accuracy	Precision	Recall	F1
SASRec	69.5	0.53	0.69	0.58
SASRec_AC	65.1	0.62	0.72	0.68
SNQN	79.2	0.75	0.83	0.6
SA2C	82.3	0.78	0.79	0.75
SA2C Smooth Enabled	87.6	0.86	0.85	0.81
Proposed	95.2	0.92	0.91	0.93

Table 2 performance comparisons of Pima Indians diabetes dataset

Methods	Accuracy	Precision	Recall	F1
SASRec	75.2	0.62	0.62	0.62
SASRec_AC	74.1	0.71	0.72	0.71
SNQN	80.5	0.69	0.69	0.76
SA2C	72.2	0.75	0.75	0.82
SA2C Smooth Enabled	85.7	0.83	0.83	0.87
Proposed	93.3	0.96	0.96	0.97

Figures 4, 5, 6, and 7 depict the evolution metrics of proposed recommendation system model. In contrast, baseline methods SNQN and SA2C exhibit declining and diverging performance as negative samples increase.

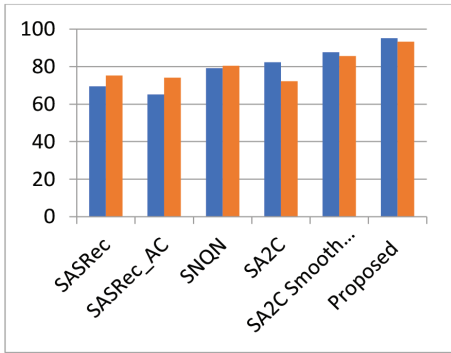


Figure 4: Accuracy

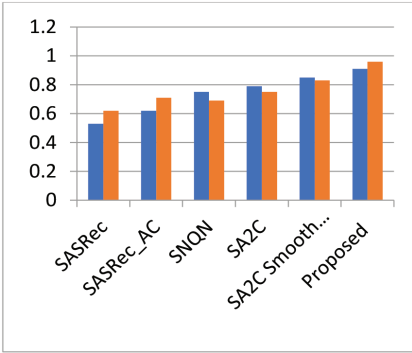


Figure 5: Precision

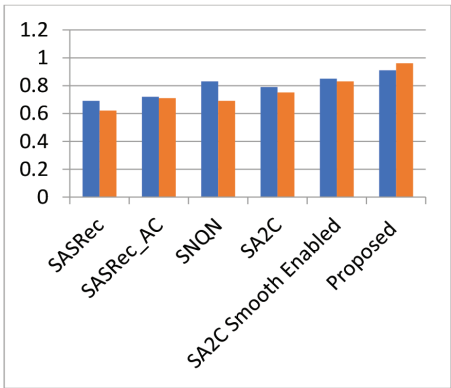


Figure 6: Recall

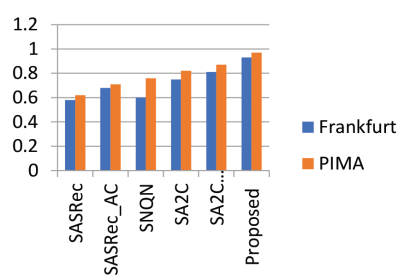


Figure 7: F1 Score

5. Conclusion & Future work

A researcher has unveiled groundbreaking understanding into the potential of incorporating contrastive learning into RS, revolutionizing their effectiveness. The proposed recommender systems are two-step process. First, the recommender agent is practiced solely from stored information without direct collaboration in environments. Second, it eliminates negative rewards, addressing key challenges in this domain. To overcome these issues, two frameworks such as SNQN and SA2C, were introduced with recommendation settings to explore the utility of RL. Conservative nature of Q-learning is leveraged to preventing overestimation of Q-values, which could otherwise guide to suboptimal protocols. This adjustment safeguards against overly optimistic reward assumptions associated with certain actions. To validate this approach, it is integrated into five recommendation models, tested on two real-world datasets. Future research directions include online testing, experiments in various use cases (e.g., promoting recommendation

diversity, enhancing video watching time), and expanding the framework to slate-based recommendations, where actions involve recommending sets of items.

Reference

1. Chen, M. (2021). Exploration in recommender systems. In Proceedings of the 15th ACM Conference on Recommender Systems, pages 551–553.
2. Emilio Parisotto, H Francis Song, Jack W Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant M Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, et al. 2019. Stabilizing Transformers for Reinforcement Learning. arXiv preprint arXiv:1910.06764 (2019).
3. Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M Jose, and Xiangnan He. 2019. A simple convolutional generative network for next item recommendation. In WSDM. 582–590
4. Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In CIKM. 1441–1450
5. Gao, R.; Xia, H.; Li, J.; Liu, D.; Chen, S.; Chun, G. DRCGR: Deep Reinforcement Learning Framework Incorporating CNN and GAN-Based for Interactive Recommendation. In Proceedings of the 2019 IEEE ICDM, Beijing, China, 8–11 November 2019; pp. 1048–1053.
6. H. F. Germany, “Diabetes data set,” 2023, <https://www.kaggle.com/johndasilva/diabetes>.
7. Hado V Hasselt. 2010. Double Q-learning. In Advances in Neural Information Processing Systems. 2613–2621
8. Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In SIGIR. 726–735.
9. Jiayi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In WSDM. 565–573.
10. Jin Huang, Zhaochun Ren, Wayne Xin Zhao, Gaole He, Ji-Rong Wen, and Daxiang Dong. 2019. Taxonomy-aware multi-hop reasoning networks for sequential recommendation. In WSDM. 573–581.
11. Kumar, A., Zhou, A., Tucker, G., and Levine, S. (2020). Conservative q-learning for offline reinforcement learning. Advances in Neural Information Processing Systems, volume 33, pages 1179–1191. Curran Associates, Inc.
12. Lei, Y.; Li, W. Interactive Recommendation with User-Specific Deep Reinforcement Learning. ACM Trans. Knowl. Discov. Data 2019, 13, 1–15.
13. Li, J., Zhang, W., Wang, T., Xiong, G., Lu, A., and Medioni, G. (2023). Gpt4rec: A generative framework for personalized recommendation and user interests interpretation.
14. Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H Chi. 2019. Top-k off-policy correction for a REINFORCE recommender system. In WSDM. 456–464.

15. OpenAI (2023). Gpt-4 technical report.
16. Pima Indians Diabetes Database. <https://kaggle.com/uciml/pima-indians-diabetes-database> (accessed August , 2023).
17. Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive learning for representation degeneration problem in sequential recommendation. In WSDM. 813–823.
18. Ruining He, Chen Fang, Zhaowen Wang, and Julian McAuley. 2016. Vista: A visually, socially, and temporally-aware model for artistic recommendation. In RecSys. 309–316.
19. Van Den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. CoRR, abs/1807.03748. 11
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
21. Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In ICDM. 197–206.
22. Wu, C.-Y., Ahmed, A., Beutel, A., Smola, A. J., and Jing, H. (2017). Recurrent recommender networks. In *Proceedings of the tenth ACM international conference on web search and data mining*, pages 495–503.
23. Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, and Joemon M Jose. 2020. Self Supervised Reinforcement Learning for Recommender Systems. SIGIR (2020).
24. Xin, X., Karatzoglou, A., Arapakis, I., and Jose, J. (2020). Self-supervised reinforcement learning for recommender systems. In *Proceedings of the 43th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*.
25. Xin, X., Karatzoglou, A., Arapakis, I., and Jose, J. M. (2022). Supervised advantage actor-critic for recommender systems. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, WSDM '22*, page 1186–1196, New York, NY, USA. Association for Computing Machinery.
26. Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. 2022. Contrastive learning for sequential recommendation. In ICDE. 1259–1273.
27. Zhao, X.; Xia, L.; Zhang, L.; Tang, J.; Ding, Z.; Yin, D. Recommendations with Negative Feedback via Pairwise Deep Reinforcement Learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, London, UK, 19–23 August 2018; pp. 1040–1048.
28. Zhou, F.; Luo, B.; Hu, T.; Chen, Z.; Wen, Y. A Combinatorial Recommendation System Framework Based on Deep Reinforcement Learning. In *Proceedings of the 2021 IEEE SIGKDD International Conference on Big Data (Big Data)*, Orlando, FL, USA, 15–18 December 2021; pp. 5733–5740.

29. Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., Wang, Z., and Wen, J.-R. (2020). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. page 1893–1902, New York, NY, USA. Association for Computing Machinery.

Acknowledgment:

This research doesn't received any specific grant from any funding agency in the public, commercial, or non-profit sectors and this study was done without any financial support only.