

SUBER: An RL Environment with Simulated Human Behavior for Recommender Systems

Nathan Corecco*
ETH Zurich
Zurich, CH

Giorgio Piatti*
ETH Zurich
Zurich, CH

Luca A. Lanzendörfer†
ETH Zurich
Zurich, CH

Flint Xiaofeng Fan
National University of Singapore
Singapore, SG

Roger Wattenhofer
ETH Zurich
Zurich, CH

ABSTRACT

Reinforcement learning (RL) has gained popularity in the realm of recommender systems due to its ability to optimize long-term rewards and guide users in discovering relevant content. However, the successful implementation of RL in recommender systems is challenging because of several factors, including the limited availability of online data for training on-policy methods. This scarcity requires expensive human interaction for online model training. Furthermore, the development of effective evaluation frameworks that accurately reflect the quality of models remains a fundamental challenge in recommender systems. To address these challenges, we propose a comprehensive framework for synthetic environments that simulate human behavior by harnessing the capabilities of large language models (LLMs). We complement our framework with in-depth ablation studies and demonstrate its effectiveness with experiments on movie and book recommendations. By utilizing LLMs as synthetic users, this work introduces a modular and novel framework for training RL-based recommender systems. The software, including the RL environment, is publicly available.¹

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Theory of computation** → *Reinforcement learning*; • **Computing methodologies** → *Natural language generation*.

KEYWORDS

Recommender Systems, Reinforcement Learning, Large Language Models

ACM Reference Format:

Nathan Corecco, Giorgio Piatti, Luca A. Lanzendörfer, Flint Xiaofeng Fan, and Roger Wattenhofer. 2024. SUBER: An RL Environment with Simulated Human Behavior for Recommender Systems. In *Proceedings of First*

*Equal contribution.

†Corresponding Author. Email: lanzendoerfer@ethz.ch

¹<https://github.com/SUBER-Team/SUBER>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GenAIRecP@KDD 2024, August 25, 2024, Barcelona, Spain

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM

<https://doi.org/XXXXXXXX.XXXXXXX>

Workshop on Generative AI for Recommender Systems and Personalization (GenAIRecP@KDD 2024). ACM, New York, NY, USA, 21 pages. <https://doi.org/XXXXXXXX.XXXXXXX>

1 INTRODUCTION

In an age defined by the ubiquitous presence of digital platforms in both leisure and commerce, recommender systems have emerged as instrumental tools in guiding user choices. From Netflix tailoring movie suggestions to match the cinematic tastes of users to Amazon presenting personalized lists of products to shoppers, recommendation systems are the engines driving enhanced user experiences and platform engagement [3, 40]. Reinforcement Learning (RL), with its principles rooted in learning by interaction, provides a compelling approach to dynamically and adaptively tailor recommendations. Recommender systems should take into account both short- and long-term rewards and direct the interests of users towards appropriate recommendations. An increasing body of research has investigated the use of RL in recommender systems [1, 8, 16, 19, 20]. Although promising, the use of RL for recommendation systems comes with its own set of challenges:

Data Availability: RL algorithms require a significant amount of data from interactions with the environment to learn effective policies. However, in the case of recommender systems, users may quickly abandon the service if they receive random or irrelevant recommendations. This makes it impractical to collect the large amount of data needed to train an RL model without compromising the user experience [48].

Unknown user model: In RL, a reward function is crucial to allow the model to learn effectively. In the context of recommender systems, designing an appropriate synthetic reward function that accurately reflects user satisfaction or preferences can be challenging due to the complexity of modeling human behavior [9, 38].

Model evaluation: A key challenge in recommender systems is the evaluation of models without directly interacting with real users, thus avoiding any potential negative impact on the user experience. On the other hand, evaluating on offline data does not guarantee good recommendation performance in the real world [12, 37].

In this work, we propose a "Simulated User Behavior for Recommender Systems" (SUBER), a novel framework for recommender systems to address the aforementioned challenges. SUBER is a framework for synthetic environments that use Large Language Models (LLM) at their core. SUBER leverages recent advances in LLMs to simulate human behavior [4, 28]. Furthermore, by training on large amounts of data, LLMs have obtained inherent knowledge

about movies, books, and various other objects. These strengths, the ability to mimic human behavior coupled with vast knowledge about humanity, uniquely position LLMs as a powerful tool to simulate users in synthetic environments for recommender systems. Therefore, SUBER serves as a versatile playground for researchers, allowing them to experiment with different LLM configurations, fine-tune user specifications, and improve their RL strategies. Our contributions can be summarized as follows.

- We propose SUBER, a general framework to train and evaluate RL-based recommender systems. Our framework consists of a gym environment containing an LLM that is capable of simulating human behavior and rate recommended items accordingly.
- We conduct extensive ablation studies to examine the influence of the different components in our proposed framework. Moreover, we present findings across several families of LLMs and their effect on the performance of the environment, highlighting the effectiveness of various LLMs in replicating human choices for item recommendations.
- We experimentally assess our environment by evaluating both a movie recommendation setting and a book recommendation setting. We also make all code available as open-source.¹

2 RELATED WORK

Recommender systems. Platforms such as YouTube [8, 16] and BytePlus [20] are two of many recent successful examples of training and evaluating recommender systems with online data. Traditional neural recommender systems and have been extensively researched in the past three decades [5, 13, 21, 39, 41, 47]. In the area of RL for recommender systems, most of the work has been based mainly on training and evaluation based on offline datasets [1, 19]. However, it presents significant challenges with respect to complexity and expense [1]. In contrast, offline evaluation takes place in a static and biased environment. To this end, many simulation environments for recommender systems have been developed [9, 15, 33, 38, 45, 49], and we present in Table 1 a qualitative comparison. Unlike previous approaches, our work leverages natural language by using LLMs to simulate user behavior. In addition, our framework is not dataset dependent, and therefore, the set of users and items are not restricted to specific domains.

Large Language Models. There have been significant recent advances in the field of LLMs [6, 17, 27, 42, 43]. LLMs can generate textual content that rivals the quality of human-generated text [6]. Park et al. [28] demonstrated how LLMs can be used to simulate human behavior. Furthermore, Argyle et al. [4] suggests the use of language models as human surrogates within social science research. Contemporary work has also integrated LLMs into recommender systems. Kang et al. [18] demonstrated that fine-tuned LLMs outperform traditional supervised methods in predicting user ratings with fewer training data, while Wang et al. [46] employed LLMs as a recommendation agent, showcasing their potential to improve recommender systems. Our work diverges from these approaches by using LLMs as simulation environments for item recommendation, in contrast to prior efforts that focused on training LLMs to function as the recommender system itself.

3 FRAMEWORK

To address the aforementioned challenges of data availability, unknown user model, and model evaluation, we propose SUBER, an environment designed to simulate human behavior through the integration of LLMs. SUBER serves a dual purpose by generating synthetic data and harnessing the capabilities of LLMs to replicate the behavior of individuals with unknown patterns. Additionally, this dynamic environment can function as a model evaluation tool for recommender systems.

SUBER consists of an LLM component and three separate modules that contain multiple individual components. An overview of the overall structure is presented in Figure 1. The internal memory module of the environment contains two separate datasets, one for users and one for items. The environment also includes a preprocessing module that retrieves raw data from the memory module and transforms it to ensure compatibility with the LLM. Finally, a postprocessing component transforms the output produced by the LLM before returning it to the RL model.

The interaction with an RL model involves the following information flow: initially, the environment selects a user from memory, along with their interaction history (i.e., items and associated ratings) as the observation for the RL model. The RL model then recommends an item to the user as its action, with an action space equal to the number of items in the environment. The action and

Table 1: Comparison of simulation environments for RecSys. We list whether the user and item dataset are real or synthetic. Simulation Engine indicates the different approaches used. For the evaluation strategy, we distinguish between offline evaluation on the original dataset used to train the simulator, online testing on a platform, sanity checks, and case studies.

Simulators	User dataset	Item dataset	Simulation engine	Evaluation strategy
Adversarial [9]	Real	Real	GAN	Offline
VirtualTaobao [38]	Real	Real	GAN	Online
RL4RS [45]	Real	Real	Transformer	Online
KuaiSim [49]	Real	Real	Transformer	Offline
RecoGym [33]	Synthetic	Synthetic	Statistical modelling	Sanity checks
RecoSim [15]	Synthetic	Synthetic	Statistical modelling	Case studies
SUBER (our)	Synthetic	Real	LLM	Sanity checks & case studies

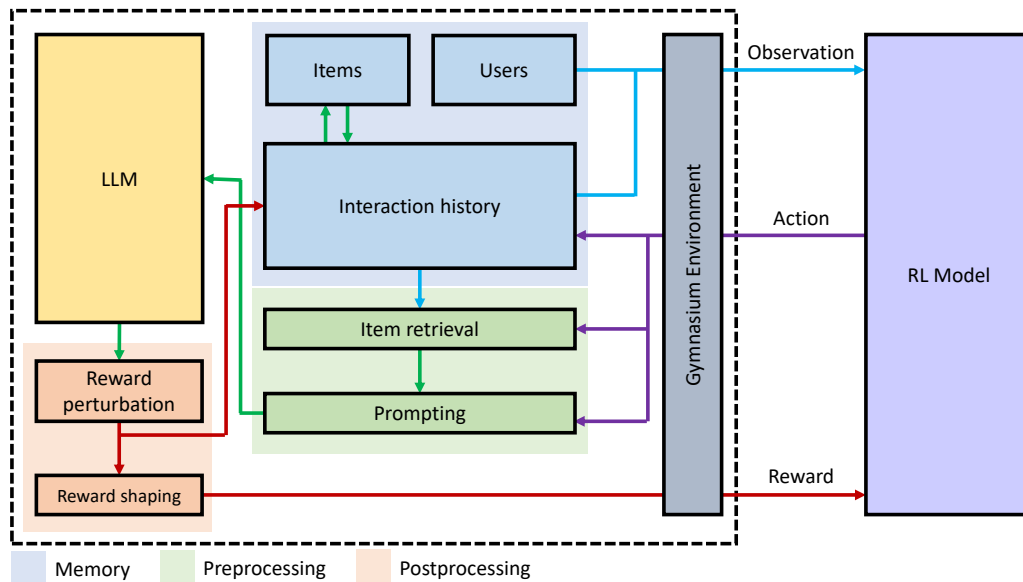


Figure 1: Overview of SUBER. The environment is built as a modular framework where each component can be modified as required. The basic control flow is as follows: The environment provides an observation using the memory module; the RL model returns an item recommendation in the form of an action, which is processed into a prompt by the memory and preprocessing component before being passed to the LLM. The score returned by the LLM is postprocessed, stored in memory and returned as a reward to the RL model.

observation are subsequently processed through the preprocessing module, the LLM component, and the postprocessing module. Finally, the environment returns a reward corresponding to the postprocessed rating predicted by the LLM. We describe each module in more detail in the following sections.

Our environment is designed with easy accessibility and extensibility in mind. Therefore, we chose a modular approach and based the environment interface on the standardized API of Gymnasium [44]. Different components can be modified at will, providing additional flexibility in future design choices.

3.1 Memory

We introduce the following notation. We define U as the set of users and I as the set of items. For every user-item pair $(u, i) \in U \times I$, we have a set $R_{u,i}$ that records all interactions between user u and item i . Similarly, for every user u we define with R_u the set of all interactions with all movies, defined as follows:

$$R_u = \{(i, h) | i \in I, h \in R_{u,i}\}. \quad (1)$$

The memory module consists of three components: an item dataset, a user dataset, and a record of all interactions between users and items. This interaction history stores the set of interactions $R_{u,i}$ for each user-item pair (u, i) . Every interaction between the RL model and the environment produces a new interaction record between a user and an item, which is appended to the interaction history.

3.2 Preprocessing

Item retrieval. As the RL model interacts with the environment, the interaction history grows. It may be challenging to extract

relevant information from long histories, and increasing history length will likely exceed the context length of current LLMs [28]. To address this issue, we propose an item-retrieval component responsible for retrieving the most appropriate items for the current query from the interaction history of a user. Additionally, as user interests and preferences may evolve over time, relying solely on user features may not accurately capture current interests. Therefore, historical rating data are used to provide a more detailed depiction of their evolving preferences.

Prompting. The prompting component aggregates the information retrieved by the item retrieval component, creating a prompt that contains the necessary details for the LLM, including the user and query item data. The objective of this prompt is to enable the LLM to accurately predict the rating of the current query item. An example of this prompt in an interaction can be seen in Figure 2.

3.3 Postprocessing

Reward perturbation. The reward perturbation component introduces noise into the ratings generated by the LLM. This component functions as a simulation of “concept drift” for users [51]. Concept drift refers to the notion that users may change their interests over time and are unlikely to maintain static preferences.

Reward shaping. Similarly to the reward perturbation component, reward shaping modifies the reward. However, unlike the perturbed reward which is added to the memory, the reward modified by the reward shaping component is returned directly to the RL model and is not stored in memory. The reward shaping module aims to reflect changes in the reward that are not related to a change in

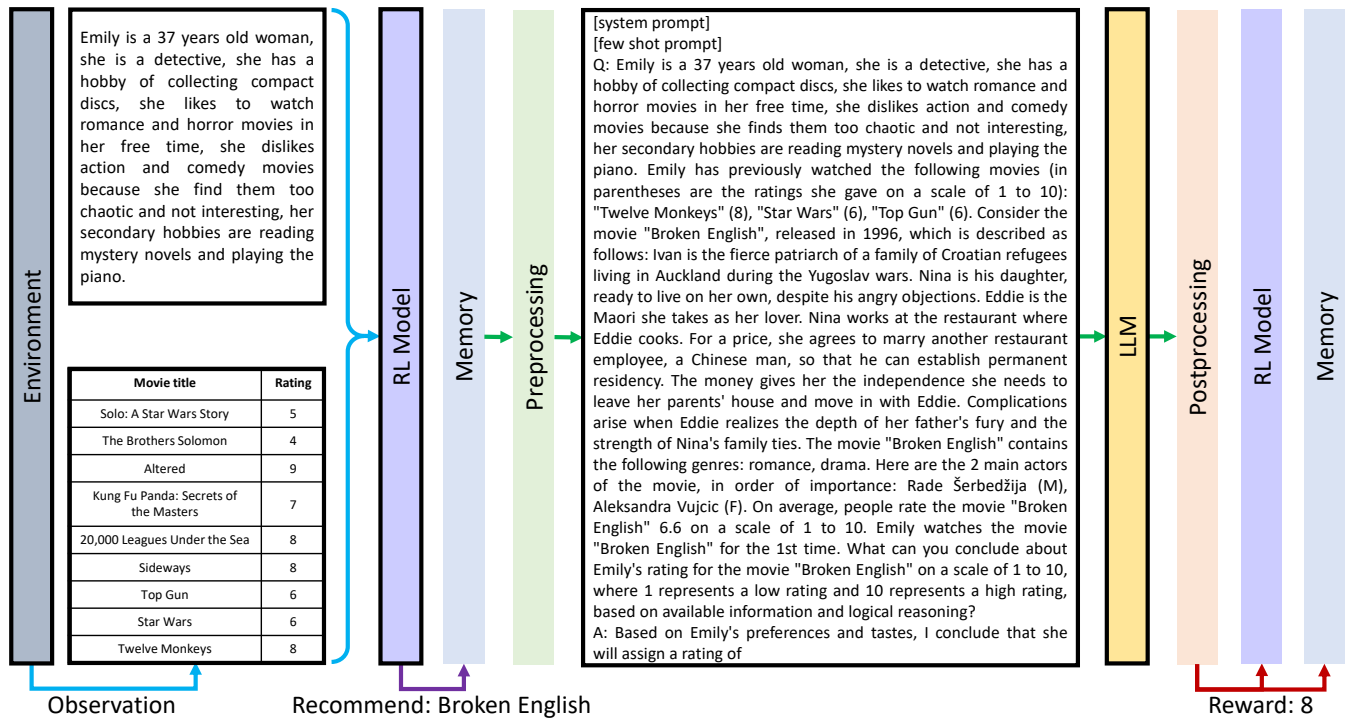


Figure 2: Pipeline of one interaction between the RL model and SUBER. The environment provides an observation in the form of a user description and user-item interaction history to the RL model. The RL model then recommends an item, which is processed into a prompt together with the user description and interaction history. The LLM uses this prompt to generate a reward for the recommended item. The reward is stored as part of the user-item interaction history and returned to the RL model.

the preference of a user, such as spontaneous decisions or fleeting interests.

4 EXPERIMENTS

To evaluate SUBER, we followed the approach of Rohde et al. [33] and Ie et al. [15]. We perform case studies and sanity checks, which we present in Section 4.3 and Appendix C.1. To achieve this, we implemented a movie recommendation and a book recommendation environment in our framework. In the following sections, we discuss our implementation and design choices for these environments. For the movie setting, we use rewards from 1 to 10, similar to TMDB², while for the book setting we use rewards from 1 to 5, as found in the Amazon Reviews Dataset [26]. For both environments, we created a synthetic user dataset using Vicuna [50] with Guidance [22]. To generate user descriptions, we condition the LLM with information such as the age, liked and disliked genres, hobbies, and profession of the user (cf. Listing 1). We generate the user age by sampling from the age distribution in the United States [7], while hobbies and professions were sampled from a precompiled list (cf. Appendix G). For the movie dataset we use MovieLens [14]. For the book dataset, we used a subset of the Amazon Book Dataset. For more details, see Appendix A and Appendix B.

Can you generate details for a person, you need to generate a name, an age, a hobby, a job and a detailed, long and original description that contains the persons interests and secondary hobbies. Please outline the cinematic preferences of the individual, detailing their favorite and least favorite genres. Kindly provide explanations for each genre preference.
 Name: *Emily Johnson*, Age: *37*, Gender: *F*, Hobby: *compact discs*, Job: *detective*
 Genres liked: *romance, horror*, Genres disliked: *action, comedy*
 Description: *she is a detective, she has a hobby of collecting compact discs, she likes to watch romance and horror movies in her free time, she dislikes action and comedy movies because she find them too chaotic and not interesting, her secondary hobbies are reading mystery novels and playing the piano.*

Listing 1: User generation. Black text shows the template, red marks the sampled information from external distributions, blue indicates the content generated by LLM.

4.1 Setup

In this section we describe the standard configuration of our framework, we also conduct ablations experiments that justify these choices, for more details see Appendix C.1.

We retrieve items from the history based on their similarity to the query item. We generate item-description embeddings using a Sentence-T5 model [25] and compute their similarities based on

²<https://www.themoviedb.org/>

Table 2: Results from RL methods trained on SUBER: "Liked genres" represents the percentage of user-preferred genres among the top ten recommendations (cf. Appendix F). "Pers.@10" denotes personalization within the initial ten recommendations.

Algorithm	Average reward	MAP@10 \uparrow	MMR@10 \uparrow	Pers.@10 \uparrow	% Liked genres \uparrow	% Disliked genres \downarrow
DQN	6.79 \pm 0.06	0.53 \pm 0.06	0.85 \pm 0.06	0.00 \pm 0.00	0.42 \pm 0.01	0.15 \pm 0.01
PPO	6.91 \pm 0.03	0.59 \pm 0.01	0.84 \pm 0.01	0.99\pm0.00	0.44 \pm 0.01	0.15 \pm 0.00
TRPO	7.25 \pm 0.08	0.67 \pm 0.06	0.91 \pm 0.02	0.35 \pm 0.06	0.45 \pm 0.02	0.14 \pm 0.01
A2C	7.93\pm0.07	0.88\pm0.01	0.96\pm0.01	0.91 \pm 0.03	0.49\pm0.02	0.11\pm0.01

the cosine distance. To select the item-rating pair to retrieve from memory, we compute the similarity between the query item and all items previously viewed by the current user, selecting the items with the highest similarity. We use the items returned from the retrieval component to construct a prompt to query the LLM. The LLM is tasked with generating a rating of the queried item by the current user, where the queried item corresponds to the item suggested by the recommender system. We construct the prompt such that the user description comes first, allowing us to leverage the key-value cache [29], eliminating the need to recalculate all intermediate embeddings within the layers of the LLM for already encountered prefixes, therefore, increasing execution speed. We use few-shot prompting to improve model performance as shown by Brown et al. [6]. In addition we use a custom system prompt see Appendix A for movie and Appendix B for books. We experimented with various GPTQ [11] versions of Llama, Vicuna, Mistral using LLMs that could run within a 24GB memory limit, for more details see Appendix D and we choose Mistral-7B as default.

Our implementation of reward shaping operates on the following premise: as a user engages with an item more frequently, their interest in revisiting it diminishes. Conversely, as time passes, the likelihood of the user interacting with the item increases again [34]. Given this insight, let us consider a user u from the set U and an item i with which the user has interacted n_{ui} times. When a time span of Δt has passed since the last interaction with the item, the reward r undergoes a reshaping process, characterized by the following equation:

$$r \leftarrow \max(1, \lfloor r \cdot q^{n_{ui}/\Delta t} \rfloor), \quad (2)$$

where $q \in [0, 1]$. This adjustment takes into account both the frequency of user interaction with the item and the time elapsed since their last interaction, resulting in the modified reward r .

4.2 Human evaluation

We conducted a case study to better evaluate the quality of different LLMs in the rating simulation task. For the study, we sampled ten user-movie interactions, and for each interaction we queried four different LLMs. The random rating in Table 3 serves as a baseline comparison, allowing us to compare the quality of our proposed approach with a random signal. The answer is constructed by sampling a rating uniformly at random between 1 and 10, and having the LLM (*Vicuna 13B*) generate the explanation for the rating. We then asked participants to rate the quality of the LLM's response on a scale of 1 to 5. More information on the study setting is provided in Appendix E.

Table 3: Human evaluation scores for various LLMs.

LLM	Score \uparrow
Random rating	2.87 \pm 1.51
Vicuna 13B	3.22 \pm 1.32
Llama-2-Chat 13B	3.42 \pm 1.22
Mistral 7B	3.80 \pm 1.27
GPT-4	4.47\pm0.77

4.3 Benchmarks

We demonstrate the viability of our environment to train an RL recommender system. The architecture of the RL model is inspired by the principles of Low-Rank Approximations in collaborative filtering [2]. We implemented four different agents based on A2C [23], PPO [36], TRPO [35], and DQN [24]. We train all models for 1.6M steps on SUBER. Due to space constraints, a more detailed discussion on the training of the reinforcement learning models is deferred to Appendix F. Our evaluation indicates that the A2C algorithm demonstrates the best overall performance in our case study (cf. Table 2). While the PPO algorithm registers a higher personalization score, indicative of its ability to tailor recommendations, it is less effective than A2C in aligning recommendations with user interests, as reflected in the percentage of liked genres metric. This suggests that A2C is more adept at discerning and catering to user preferences. For a qualitative insight into the performance of our model see Appendix F.

5 CONCLUSION

Our research offers a possible avenue to address the persistent challenge of training recommender systems in the absence of real user interactions. Conventional approaches that depend on user-item interaction histories or synthetic data have often failed to replicate real-world usage scenarios accurately. By introducing SUBER, a novel RL environment designed specifically for recommender system training, and incorporating recent advances in LLMs to emulate human behavior in the training environment, we have proposed a potential solution to this long-standing issue. Our results, as demonstrated through a series of ablation studies, experiments, and human evaluation, underscore the efficacy of our approach. We believe that this work marks a step toward achieving more realistic and practical training environments for recommender systems, even when direct user interactions are unavailable.

REFERENCES

- [1] M Mehdi Afsar, Trafford Crump, and Behrouz Far. 2022. Reinforcement learning based recommender systems: A survey. *Comput. Surveys* 55, 7 (2022), 1–38.
- [2] Charu C Aggarwal et al. 2016. *Recommender systems*. Vol. 1. Springer.
- [3] Sanjay Agrawal, Srjana Merugu, and Vivek Sembium. 2023. Enhancing e-commerce product search through reinforcement learning-powered query reformulation. (2023).
- [4] Lisa P Argyle, Ethan C Busby, Nancy Fulda, et al. 2023. Out of one, many: Using language models to simulate human samples. *Political Analysis* 31, 3 (2023), 337–351.
- [5] Jesús Bobadilla, Fernando Ortega, et al. 2013. Recommender systems survey. *Knowledge-based systems* 46 (2013), 109–132.
- [6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. 2020. Language models are few-shot learners. *NeurIPS* 33 (2020), 1877–1901.
- [7] United State Census Bureau. 2022. National Population by Characteristics: 2020–2022. <https://www.census.gov/data/tables/time-series/demo/popest/2020s-national-detail.html> 09.21.2023.
- [8] Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, et al. 2019. Top-k off-policy correction for a REINFORCE recommender system. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 456–464.
- [9] Xinshi Chen, Shuang Li, Hui Li, Shaohua Jiang, Yuan Qi, and Le Song. 2019. Generative adversarial user model for reinforcement learning based recommendation system. In *ICML*. PMLR, 1052–1061.
- [10] Morning Consult. 2018. Most popular movie genres among adults in the United States as of December 2018, by gender. *Dec. 2018c. Statista, Graph* (2018).
- [11] Elias Franter, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323* (2022).
- [12] Florent Garcin, Boi Faltings, Olivier Donatsch, Ayar Alazzawi, et al. 2014. Offline and online evaluation of news recommender systems at swissinfo. ch. In *Proceedings of the 8th ACM Conference on Recommender systems*. 169–176.
- [13] David Goldberg, David Nichols, Brian M Oki, and Douglas Terry. 1992. Using collaborative filtering to weave an information tapestry. *Commun. ACM* 35, 12 (1992), 61–70.
- [14] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* 5, 4 (2015), 1–19.
- [15] Eugene Ie, Chih-wei Hsu, Martin Mladenov, Vihan Jain, et al. 2019. Recsim: A configurable simulation platform for recommender systems. *arXiv preprint arXiv:1909.04847* (2019).
- [16] Eugene Ie, Vihan Jain, Jing Wang, Sanmit Narvekar, et al. 2019. Reinforcement learning for slate-based recommender systems: A tractable decomposition and practical methodology. *arXiv preprint arXiv:1905.12767* (2019).
- [17] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, et al. 2023. Mistral 7B. [arXiv:2310.06825 \[cs.CL\]](https://arxiv.org/abs/2310.06825)
- [18] Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, et al. 2023. Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction. *arXiv preprint arXiv:2305.06474* (2023).
- [19] Yuanguo Lin, Yong Liu, Fan Lin, Lixin Zou, et al. 2023. A survey on reinforcement learning for recommender systems. *IEEE Transactions on Neural Networks and Learning Systems* (2023).
- [20] Zhuoran Liu, Leqi Zou, Xuan Zou, Caihua Wang, et al. 2022. Monolith: real time recommendation system with collisionless embedding table. *ORSUM@ACM RecSys 2022* (2022).
- [21] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. 2015. Recommender system application developments: a survey. *Decision support systems* 74 (2015), 12–32.
- [22] Scott Lundberg. 2023. Guidance. <https://github.com/guidance-ai/guidance>.
- [23] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, et al. 2016. Asynchronous methods for deep reinforcement learning. In *ICML*. PMLR, 1928–1937.
- [24] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, et al. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).
- [25] Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, et al. 2022. Sentence-T5: Scaling up Sentence Encoder from Pre-trained Text-to-Text Transfer Transformer. (2022).
- [26] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *EMNLP-IJCNLP*. Association for Computational Linguistics, Hong Kong, China, 188–197. <https://doi.org/10.18653/v1/D19-1018>
- [27] OpenAI. 2023. GPT-4 Technical Report. [arXiv:2303.08774 \[cs.CL\]](https://arxiv.org/abs/2303.08774)
- [28] Joon Sung Park, Joseph C O’Brien, Carrie J Cai, Meredith Ringel Morris, et al. 2023. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442* (2023).
- [29] Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, et al. 2023. Efficiently scaling transformer inference. *Proceedings of Machine Learning and Systems* 5 (2023).
- [30] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, et al. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *JMLR* 22, 268 (2021), 1–8. <http://jmlr.org/papers/v22/20-1364.html>
- [31] Arjun Raj. 2022. A Comprehensive Collection of Hobbies. <https://www.kaggle.com/datasets/mrhell/list-of-hobbies>
- [32] Marlon Ramos, Angelo M Calvão, and Celia Anteneodo. 2015. Statistical patterns in movie rating behavior. *Plos one* 10, 8 (2015), e0136083.
- [33] David Rohde, Stephen Bonner, Travis Dunlop, Flavian Vasile, and Alexandros Karatzoglou. 2018. Recogym: A reinforcement learning environment for the problem of product recommendation in online advertising. *arXiv preprint arXiv:1808.00720* (2018).
- [34] Cristel Antonia Russell and Sidney J. Levy. 2011. The Temporal and Focal Dynamics of Volitional Reconsumption: A Phenomenological Investigation of Repeated Hedonic Experiences. *Journal of Consumer Research* 39, 2 (10 2011), 341–359. <https://doi.org/10.1086/662996> [arXiv:https://academic.oup.com/jcr/article-pdf/39/2/341/5086800/39-2-341.pdf](https://academic.oup.com/jcr/article-pdf/39/2/341/5086800/39-2-341.pdf)
- [35] John Schulman, Sergey Levine, Pieter Abbeel, et al. 2015. Trust region policy optimization. In *ICML*. PMLR, 1889–1897.
- [36] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).
- [37] Guy Shani and Asela Gunawardana. 2011. Evaluating recommendation systems. *Recommender systems handbook* (2011), 257–297.
- [38] Jing-Cheng Shi, Yang Yu, Qing Da, Shi-Yong Chen, and An-Xiang Zeng. 2019. Virtual-taobao: Virtualizing real-world online retail environment for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4902–4909.
- [39] Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)* 47, 1 (2014), 1–45.
- [40] Harald Steck, Linas Baltrunas, Ehtsham Elahi, et al. 2021. Deep learning for recommender systems: A Netflix case study. *AI Magazine* 42, 3 (2021), 7–18.
- [41] Xiaoyuan Su and Taghi M Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in artificial intelligence* 2009 (2009).
- [42] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
- [43] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).
- [44] Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, et al. 2023. Gymnasium. <https://doi.org/10.5281/zenodo.8127026>
- [45] Kai Wang, Zhene Zou, Minghao Zhao, et al. 2023. RL4RS: A Real-World Dataset for Reinforcement Learning based Recommender System. In *SIGIR’23*. 2935–2944.
- [46] Yan Cheng Wang, Ziyang Jiang, Zheng Chen, Fan Yang, et al. 2023. Recmind: Large language model powered agent for recommendation. *arXiv preprint arXiv:2308.14296* (2023).
- [47] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys (CSUR)* 52, 1 (2019), 1–38.
- [48] Weinan Zhang, Ulrich Paquet, and Katja Hofmann. 2016. Collective noise contrastive estimation for policy transfer learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 30.
- [49] Kesen Zhao, Shuchang Liu, Qingpeng Cai, et al. 2023. KuaiSim: A comprehensive simulator for recommender systems. *arXiv preprint arXiv:2309.12645* (2023).
- [50] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, et al. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *arXiv preprint arXiv:2306.05685* (2023).
- [51] André Zliobaitė, Mykola Pechenizkiy, and João Gama. 2016. *An Overview of Concept Drift Applications*. Springer International Publishing, Cham, 91–114.

A DETAILS OF MOVIE ENVIRONMENT

A.1 Users generation

When generating synthetic users, our process begins by randomly sampling an age from a distribution that reflects age demographics in the United States [7]. In addition, we randomly select a hobby and a profession from predefined lists. These hobby lists are divided into two categories: one tailored for children (aged 4-17) and another for adults (aged 18-75). Users not of working age are assigned the profession "student," while those of retirement age are categorized as "retired."

In total, children users can have one of 33 hobbies, while adult users have a choice of 422 hobbies. With regard to professions, there are 200 different options available. Once all user attributes are determined, they are incorporated into a prompt that generates a user description (see Listing 1 for an example). All lists are generated using the *GPT-3.5* model, with the exception of the list of adult hobbies, for which we used data from [31].

For a complete list of hobbies and professions generated, refer to Appendix G. An illustrative example of a synthetic user can be found in Listing 1.

To train the RL model, we created an additional dataset using a similar approach. The primary distinction lies in how we sampled the user's preferred and disliked genres, which were not generated using the LLM. This modification was made to ensure that the dataset includes users with a more diverse range of preferences. We sampled the preference for movie genres according to the distribution of preferred genres in the US in 2018 [10]. We show in Figures 3 and 4 how this strategy affects the genre preference of users.

A.2 Items

We use the same set of movies as contained in MovieLens (*ml-latest-small*) [14] for our experiments and collect the respective movie features from TMDB, we show which feature we use in Table 4.

A.3 Prompting

In this section, we provide several examples of different prompting strategies. We primarily focus on three key approaches: using

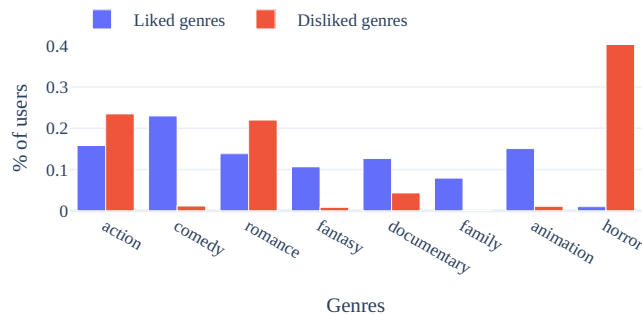


Figure 3: Genre preferences of user generated via LLM. For each movie genre, we show in blue the percentage of generated users who like the genre. Similarly, we show in red the percentage of users who do not like the genre.

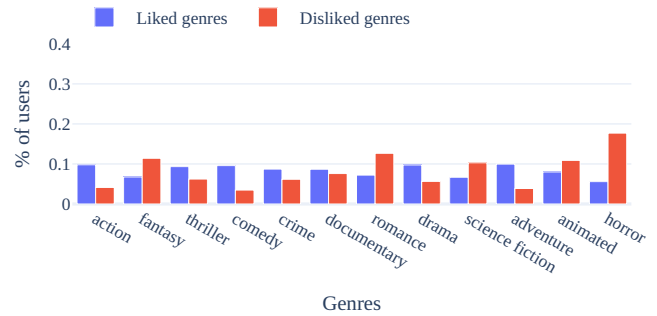


Figure 4: Real distribution of genre preferences. For each movie genre, we show in blue the percentage of generated users who like the genre. Similarly, we show in red the percentage of users who do not like the genre.

Table 4: For each movie we retrieve the features showed in the table, in our implementation we only use a subset to describe the item.

Feature	Used	Notes
Actors	Yes	2 principal actors
Budget	No	
Director	Yes	Only for feature similarity
Original language	No	
Original title	No	
Overview	Yes	Story-line of the movie
Popularity	No	
Release date	Yes	
Revenue	No	
Runtime	No	
Title	Yes	
TMDB ID	No	Unique id in the dataset
Vote average	Yes	
Vote count	No	

prompts with digits ranging from 1 to 10, utilizing prompts with digits from 0 to 9, and employing prompts with word representations for numbers from one to ten.

Both the approach of using digits from 0 to 9 and the word-based approach are designed to address tokenization ambiguity. This ambiguity arises because the number 10 can be tokenized in two different ways: as the token "10" directly or as separate tokens "1" and "0." We also explored generating numbers directly without restricting them to a single token in the 1-10 approach. However, this approach exhibited poor performance, leading us to refrain from further experimentation.

```
[system prompt]
[few shot prompts]
Q: Emily is a 37 years old woman, she is a detective, she has a hobby of collecting compact discs, she likes to watch romance and horror movies in her free time, she dislikes action and comedy movies because she finds them too chaotic and not interesting, her secondary hobbies are reading mystery novels and playing the piano.
```

Emily has previously watched the following movies (in parentheses are the ratings she gave on a scale of 0 to 9): "Twelve Monkeys" (7), "Star Wars" (6), "Top Gun" (5).

Consider the movie "Broken English", released in 1996, which is described as follows: Ivan is the fierce patriarch of a family of Croatian refugees living in Auckland during the Yugoslav wars. Nina is his daughter, ready to live on her own, despite his angry objections. Eddie is the Maori she takes as her lover. Nina works at the restaurant where Eddie cooks. For a price, she agrees to marry another restaurant employee, a Chinese man, so that he can establish permanent residency. The money gives her the independence she needs to leave her parents' house and move in with Eddie. Complications arise when Eddie realizes the depth of her father's fury and the strength of Nina's family ties. The movie "Broken English" contains the following genres :

*-romance
-drama*

Here are the 2 main actors of the movie, in order of importance: Rade SerbedZija (M), Aleksandra Vujcic (F). On average, people rate the movie "Broken English" 5.6 on a scale of 0 to 9. Emily watches the movie "Broken English" for the 1st time.

What can you conclude about Emily's rating for the movie "Broken English" on a scale of 0 to 9, where 0 represents a low rating and 9 represents a high rating, based on available information and logical reasoning?

A: Based on Emily's preferences and tastes, I conclude that she will assign a rating of 7

Listing 2: Example query for rating to the LLM (Vicuna-v1.5-13B) using the 0-9 scale.

[system prompt]

[few shot prompts]

Q: Emily is a 37 years old woman, she is a detective, she has a hobby of collecting compact discs, she likes to watch romance and horror movies in her free time, she dislikes action and comedy movies because she finds them too chaotic and not interesting, her secondary hobbies are reading mystery novels and playing the piano.

Emily has previously watched the following movies (in parentheses are the ratings she gave on a scale of 1 to 10): "Twelve Monkeys" (8), "Star Wars" (7), "Top Gun" (6).

Consider the movie "Broken English", released in 1996, which is described as follows: Ivan is the fierce patriarch of a family of Croatian refugees living in Auckland during the Yugoslav wars. Nina is his daughter, ready to live on her own, despite his angry objections. Eddie is the Maori she takes as her lover. Nina works at the restaurant where Eddie cooks. For a price, she agrees to marry another restaurant employee, a Chinese man, so that he can establish permanent residency. The money gives her the independence she needs to leave her parents' house and move in with Eddie. Complications arise when Eddie realizes the depth of her father's fury and the strength of Nina's family ties. The movie "Broken English" contains the following genres :

*-romance
-drama*

Here are the 2 main actors of the movie, in order of importance: Rade SerbedZija (M), Aleksandra Vujcic (F). On average, people rate the movie "Broken English" 6.6 on a scale of one to ten. Emily watches the movie "Broken English" for the 1st time.

What can you conclude about Emily's rating for the movie "Broken English" on a scale of one to ten, where one represents a low rating and ten represents a high rating, based on available information and logical reasoning?

A: Based on Emily's preferences and tastes, I conclude that she will assign a rating of eight

Listing 3: Example query for rating to the LLM (Vicuna-v1.5-13B) using rating scale one-ten.

A.3.1 Custom system prompt. We also experimented with various system prompts, which are predefined text or instructions used to initiate a conversation or request from a user when interacting with a language model. The primary objective was to encourage the model to generate ratings that are less biased and more closely aligned with the information provided to the model. This includes factors such as the user description, the list of movies watched previously, and the overview of the queried movie, all of which play a role in shaping the predictions of a model. In Listing 4, we present our custom system prompt utilized for various analyses in Section 4.

You are a highly sophisticated movie rating assistant, equipped with an advanced understanding of human behavior. Your mission is to deliver personalized movie recommendations by carefully considering the unique characteristics, tastes, and past-seen films of each individual. When presented with information about a specific movie, you will diligently analyze its plot, primary genres, actors, and average rating. Using this comprehensive understanding, your role is to provide thoughtful and accurate ratings for movies on a scale of 1 to 10, ensuring they resonate with the person's preferences and cinematic inclinations. Remain impartial and refrain from introducing any biases in your predictions. You are an impartial and reliable source of movie rating predictions for the given individual and film descriptions.

Listing 4: An advanced system prompt guiding the model to provide personalized and unbiased movie ratings based on detailed user and movie data.

A.3.2 Query template. In the following section, we provide an example prompt and accompanying LLM answer. It is important to note that Listing 5 displays the complete response of the model, not just the rating. During interaction with an RL model, we stop generation after producing the rating.

[system prompt]

[few shot prompts]

Q: Emily is a 37 years old woman, she is a detective, she has a hobby of collecting compact discs, she likes to watch romance and horror movies in her free time, she dislikes action and comedy movies because she finds them too chaotic and not interesting, her secondary hobbies are reading mystery novels and playing the piano.

Emily has previously watched the following movies (in parentheses are the ratings she gave on a scale of 1 to 10): "Twelve Monkeys" (8), "Star Wars" (7), "Top Gun" (6).

Consider the movie "Broken English", released in 1996, which is described as follows: Ivan is the fierce patriarch of a family of Croatian refugees living in Auckland during the Yugoslav wars. Nina is his daughter, ready to live on her own, despite his angry objections. Eddie is the Maori she takes as her lover. Nina works at the restaurant where Eddie cooks. For a price, she agrees to marry another restaurant employee, a Chinese man, so that he can establish permanent residency. The money gives her the independence she needs to leave her parents' house and move in with Eddie. Complications arise when Eddie realizes the depth of her father's fury and the strength of Nina's family ties. The movie "Broken English" contains the following genres :

-romance
-drama

Here are the 2 main actors of the movie, in order of importance: Rade SerbedZija (M), Aleksandra Vujcic (F). On average, people rate the movie "Broken English" 6.6 on a scale of 1 to 10. Emily watches the movie "Broken English" for the 1st time.

What can you conclude about Emily's rating for the movie "Broken English" on a scale of 1 to 10, where 1 represents a low rating and 10 represents a high rating, based on available information and logical reasoning?

A: Based on Emily's preferences and tastes, I conclude that she will assign a rating of 8 on a scale of 1 to 10, because

- 1) she has previously rated movies within the same genre, such as "Twelve Monkeys," which also falls under drama and romance categories, and gave it a higher score (8 out of 10);
- 2) Her secondary hobbies include reading mystery novels and playing the piano, suggesting an appreciation for more introspective and emotionally engaging content;
- 3) The plot of "Broken English" revolves around family dynamics, cultural clashes, and personal growth, all of which align with her interests in watching romance and drama films. Although Emily doesn't typically enjoy action or comedy movies, this does not necessarily mean that she would dislike "Broken English." Instead, it suggests that she might find other genres less appealing due to their lack of depth. Overall, considering her previous ratings and general preferences, it is reasonable to assume that Emily would rate "Broken English" highly.

Listing 5: Example query for rating to the LLM (Vicuna-v1.5-13B). For each user we inject their description, which contains preferences and tastes. Then we provide the movie details: storyline, genres, main actors and vote average.

A.3.3 N-shot prompting. In this section, we present two illustrative examples (Listing 6 and 7) used for both one-shot and two-shot prompts. These examples adhere to the query template shown in Appendix A.3.2, guiding the model on how to formulate responses considering all the features. In addition, both examples provide comprehensive and detailed answers.

One advantage of a few-shot prompting is that the examples are given at the start of the prompt. This gives us the opportunity to use the key-value cache [29] without slowing down the environment.

Furthermore, these two examples are designed to cover both positive and negative rewards. In the first example, the user rates the queried movie highly, while in the second example, the user rates the queried movie poorly, thereby encompassing a range of user preferences and responses.

Q: Alex is a 12 years old boy, he is captivated by space exploration. With wide eyes and endless wonder, he devours books on galaxies and dreams of becoming an astronaut. Nights find him gazing at the stars, his imagination soaring with each cosmic discovery. Alex's passion for space knows no bounds as he reaches for the stars.

Alex has previously watched the following movies (in parentheses are the ratings he gave on a scale of 1 to 10): "Madagascar" (7), "Foodfight!" (2), "Interstellar" (10).

Consider the movie "Zootopia", released in 2016, which is described as follows: determined to prove herself, Officer Judy Hopps, the first bunny on Zootopia's police force, jumps at the chance to crack her first case – even if it means partnering with scam-artist fox Nick Wilde to solve the mystery. The movie "Zootopia" contains the following genres:

-animation
-adventure
-family
-comedy

Here are the 2 main actors of the movie, in order of importance: Jason Bateman (M), Ginnifer Goodwin (F). On average, people rate the movie "Zootopia" 7.7 on a scale of 1 to 10. Alex watches the movie "Zootopia" for the 1st time.

What can you conclude about Alex's rating for the movie "Zootopia" on a scale of 1 to 10, where 1 represents a low rating and 10 represents a high rating, based on available information and logical reasoning?

A: Based on Alex's preferences and tastes, I conclude that he will assign a rating of 9 on a scale of 1 to 10, because, even though the movie is not space-related, he previously enjoyed watching "Madagascar" and gave it a high rating. Since "Zootropolis" shares many similarities with "Madagascar," such as being animated movies with animals living in a society resembling humans, it is highly likely that Alex also likes "Zootropolis." Both films explore themes of friendship, identity, and the challenges of coexistence within diverse communities. Although "Zootropolis" doesn't have a direct connection to space, Alex still appreciates it and gives it a high rating. Furthermore, considering that children generally have a fondness for animated movies, it is reasonable to assume that Alex would rate "Zootropolis" high." Additionally, "Zootropolis" is generally favored by a larger audience compared to "Madagascar," further supporting the idea that Alex would give it a higher rating.

Listing 6: First shot example for movies environment.

Q: Nicholas is a 26 years old man, he is a thrill-seeker who loves action and adventure and doesn't care much for romance. He enjoys activities like rock climbing, extreme sports, and exciting trips that make his heart race. As a wilderness guide, he leads others through tough terrains and shows them the beauty of wild places. During his free time, Nicholas loves reading gripping books that take him on exciting journeys. He looks for friends who also enjoy thrilling experiences and share his passion for living life to the fullest. Adrenaline gives him an amazing feeling, and he prefers action and adventure over romance any day.

Nicholas has previously watched the following movies (in parentheses are the ratings he gave on a scale of 1 to 10): "Fifty Shades of Grey" (3), "John Wick" (9), "The Dark Knight" (10).

Consider the movie "La La Land", released in 2016, which is described as follows: mia, an aspiring actress, serves lattes to movie stars in between auditions and Sebastian, a jazz musician, scrapes by playing cocktail party gigs in dingy bars, but as success mounts they are faced with decisions that begin to fray the fragile fabric of their love affair, and the dreams they worked so hard to maintain in each other threaten to rip them apart. The movie "La La Land" contains the following genres:

-comedy
-drama

–romance
–music

Here are the 2 main actors of the movie, in order of importance: Ryan Gosling (M), Emma Stone (F). On average, people rate the movie "La La Land" 7.9 on a scale of 1 to 10. Nicholas watches the movie "La La Land" for the 1st time.

What can you conclude about Nicholas's rating for the movie "La La Land" on a scale of 1 to 10, where 1 represents a low rating and 10 represents a high rating, based on available information and logical reasoning?

A: Based on Nicholas's preferences and tastes, I conclude that he will assign a rating of 4 on a scale of 1 to 10, because Nicholas has a strong preference for adrenaline-inducing action, thriller, and horror movies, he would likely rate the movie "La La Land" 4 out of 10. This is evident from the description of Nicholas, which highlights his enthusiasm for action-packed films that provide a surge of thrill and excitement. In his previous film ratings, action movies generally received higher scores, while films that didn't offer the same adrenaline rush, like "Fifty Shades of Grey", received lower ratings, such as a 3. As "La La Land" is a romantic musical and not focused on action, it may not resonate as strongly with Nicholas's taste for thrilling experiences. While the film is generally well-liked with an average rating of 7.9, Nicholas's preference for adrenaline-filled plots might lead him to rate "La La Land" lower than the overall community rating. However, it's likely that he wouldn't rate it as low as "Fifty Shades of Grey" due to its higher popularity and appreciation among viewers who enjoy romance and musical genres.

Listing 7: Second shot example for movies environment.

B DETAILS BOOKS ENVIRONMENT

B.1 Users generation

We generate the user dataset in the same way as we did for the users in the movie dataset, by sampling the user features from the same lists using the same method.

B.2 Items

We filter books from the Amazon Book Dataset [26] by removing books that did not have all the features: categories, description, title, and publication date. We also limit the categories to those with at least 100 books, so we do not get fine-grained categories.

B.3 Prompting

B.3.1 Custom system prompt. We also experimented with various system prompts, which are predefined text or instructions used to initiate a conversation or request from a user when interacting with a language model. The primary objective was to encourage the model to generate ratings that are less biased and more closely aligned with the information provided to the model. This includes factors such as the user description, the list of books read previously, and the back-cover of the queried book, all of which play a role in shaping the predictions of a model.

You are a highly sophisticated book rating assistant, equipped with an advanced understanding of human behavior. Your mission is to deliver personalized book recommendations by carefully considering the unique characteristics, tastes, and past read books of each individual. When presented with information about a specific book, you will diligently analyze its backcover, primary category, authors, and average rating. Using this comprehensive understanding, your role is to provide thoughtful and accurate ratings for books on a scale of 1 to 5, ensuring they resonate with the person's preferences and reading inclinations. Remain impartial and refrain from introducing any biases in your predictions. You are an impartial and reliable source of book rating predictions for the given individual and book descriptions.

Listing 8: An advanced system prompt guiding the model to provide personalized and unbiased movie ratings based on detailed user and movie data.

B.3.2 Query template.

[system prompt]
[few shot prompts]

Samuel is a 17 years old boy, he is an apprentice and loves to work with his hands. He is very interested in animal fancy and loves to breed and show his animals. Samuel is very fitness-conscious and loves to stay active. He enjoys hiking and playing sports. Samuel is a big fan of the Spirit category and enjoys reading books that can help him improve his spiritual life. He also loves reading books about crafts and enjoys learning new techniques. Samuel is also very close to his family and enjoys reading books about family relationships. He is not a big fan of religion and finds it to be boring. He also dislikes music, literary collections and juvenile fiction. He finds them to be too slow paced and not interesting enough for him. Samuel has previously read the following books (in parentheses are the ratings he gave on a scale of 1 to 5): "The Two Towers" (5), "The Fellowship of the Ring" (5), "The Horse and His Boy" (3). Consider the book "The Return of the King", released in 1955, which is described as follows: one Ring to rule them all, One Ring to find them, One Ring to bring them all and in the darkness bind them. The Dark Lord has risen, and as he unleashes hordes of Orcs to conquer all Middle-earth, Frodo and Sam struggle deep into his realm in Mordor. To defeat Sauron, the One Ring must be destroyed in the fires of Mount Doom. But the way is impossibly hard, and Frodo is weakening. The Ring corrupts all who bear it and Frodo's time is running out. Will Sam and Frodo succeed, or will the Dark Lord rule Middle-earth once more? The book "The Return of the King" belongs to the following categories:

- Fantasy
- Classic
- Fiction
- Adventure

The author of the book is J.R.R. Tolkien. On average, people rate the book "The Return of the King" 4.6 on a scale of 1 to 5. Samuel reads the book "The Return of the King" for the 1st time.

What can you conclude about Samuel's rating for the book "The Return of the King" on a scale of 1 to 5, where 1 represents a low rating and 5 represents a high rating, based on available information and logical reasoning?

Q: Based on Samuel's preferences and tastes, I conclude that he will assign a rating of 5

Listing 9: Example query for rating to the LLM. For each user we inject their description, which contains preferences and tastes. Then we provide the movie details: backcover, category, authors and vote average.

B.3.3 N-shot prompting. In this section, we present two illustrative examples (Listing 10 and 11) used for both one-shot and two-shot prompts. These examples adhere to the query template shown in Appendix B.3.2, guiding the model on how to formulate responses considering all the features. In addition, both examples provide comprehensive and detailed answers.

Furthermore, these two examples are designed to cover both positive and negative rewards. In the first example, the user rates the queried book highly, while in the second example, the user rates the queried book poorly, thereby encompassing a range of user preferences and responses.

Q: Emilia is a 20 years old woman, she is an avid reader, she spends much of her free time lost in the pages of books, especially those filled with magical worlds, exciting adventures and tales of elves. Her passion for the magical realms of literature is evident in her vivid imagination and the way her eyes light up when discussing stories. As well as reading, she enjoys drawing, attending book club meetings, stargazing, sipping tea on rainy days, baking and getting lost in stories about elves.

Emilia has previously read the following books (in parentheses are the ratings she gave on a scale of 1 to 5): "Harry Potter and the Chamber of Secrets" (5), "Harry Potter and the Philosopher's Stone" (5), "Eragon" (5).

Consider the book "Harry Potter and the Prisoner of Azkaban", released in 1999, which is described as follows: Harry Potter, along with his best friends, Ron and Hermione, is about to start his third year at Hogwarts School of Witchcraft and Wizardry. Harry can't wait to get back to school after the summer holidays. (Who wouldn't if they lived with the horrible Dursleys?) But when Harry gets to Hogwarts, the atmosphere is tense. There's an escaped mass murderer on the loose, and the sinister prison guards of Azkaban have been called in to guard the school... The book "Harry Potter and the Prisoner of Azkaban" belongs to the following categories:

- Fiction
- Young Adult
- Magic
- Classic

The author of the book is J.K. Rowling. On average, people rate the book "Harry Potter and the Prisoner of Azkaban" 4.6 on a scale of 1 to 5. Emilia reads the book "Harry Potter and the Prisoner of Azkaban" for the 1st time.

What can you conclude about Emilia's rating for the book "Harry Potter and the Prisoner of Azkaban" on a scale of 1 to 5, where 1 represents a low rating and 5 represents a high rating, based on available information and logical reasoning?

A: Based on Emilia's preferences and tastes, I conclude that she will assign a rating of 5 on a scale of 1 to 5, because from Emilia's description we can clearly see her love for magic and fantasy books, moreover the book "Harry Potter and the Prisoner of Azkaban" is the third book of the Harry Potter series, and from her history we can see that she has already read the first two books of the series and she loved them, because she assigned a perfect score of 5. Moreover, the third book that she has read has a lot to do with magic, which underlines her interest in magical words and stories. The book also has a very high average rating, suggesting that people love the book.

Listing 10: First shot example for books environment

Q: Mary is a 12 years old girl, she is a person with an overflowing heart, shares an extraordinary bond with the animal kingdom. Her eyes light up with wonder at the sight of a furry friend, and her days are filled with joyful adventures exploring the world's wildlife. From rescuing lost kittens to befriending birds in her backyard, Mary's compassion knows no bounds. Her room is a sanctuary of stuffed animals and nature books, a testament to her unwavering love for all creatures great and small. She is afraid of shadows and loves to sleep with the light on.

Mary has previously read the following books (in parentheses are the ratings she gave on a scale of 1 to 5): "Charlotte's Web" (5), "The Shining" (1), "The Trouble with Tuck" (4).

Consider the book "Coraline", released in 2002, which is described as follows: the day after they moved in, Coraline went exploring.... In Coraline's family's new flat are twenty-one windows and fourteen doors. Thirteen of the doors open and close. The fourteenth is locked, and on the other side is only a brick wall, until the day Coraline unlocks the door to find a passage to another flat in another house just like her own. Only it's different. At first, things seem marvelous in the other flat. The food is better. The toy box is filled with wind-up angels that flutter around the bedroom, books whose pictures writhe and crawl and shimmer, little dinosaur skulls that chatter their teeth. But there's another mother, and another father, and they want Coraline to stay with them and be their little girl. They want to change her and never let her go. Other children are trapped there as well, lost souls behind the mirrors. Coraline is their only hope of rescue. She will have to fight with all her wits and all the tools she can find if she is to save the lost children, her ordinary life, and herself. Critically acclaimed and award-winning author Neil Gaiman will delight readers with his first novel for all ages. The book "Coraline" belongs to the following categories:

- Horror
- Fantasy
- Fiction
- Young Adult

The author of the book is Neil Gaiman. On average, people rate the book "Coraline" 4.1 on a scale of 1 to 5. Mary reads the book "Coraline" for the 1st time.

What can you conclude about Mary's rating for the book "Coraline" on a scale of 1 to 5, where 1 represents a low rating and 5 represents a high rating, based on available information and logical reasoning?

A: Based on Mary's preferences and tastes, I conclude that she will assign a rating of 2 on a scale of 1 to 5 because, although it is a book for children, as it also falls into the Young Adult category, it is not a book that suits Mary's personality well; in fact, she is afraid of shadows when she needs to sleep, which suggests that the book "Coraline", which is mainly a horror book, is not well suited to Mary. Also, given her sensitivity and love of animals, the creepy and potentially frightening aspects of the story are too much for her. We can also see from Mary's previous red books that she has had a bad experience with horror books, in fact she rated "The Shining" 1 out of 5, whereas "Coraline" is more suitable for children, which explains why Mary probably rated "Coraline" 2 while she rated "The Shining" 1.

Listing 11: Second shot example for books environment

Table 5: Ablation results for the movie setting using *Mistral 7B* as our environment. We test the LLM on coherency and realistic ratings for user-movie interactions. We achieve best performance with 0-9 digit rating scale, 2-shot prompting, and our custom system prompt.

Prompt component			Genres \uparrow	High/Low \uparrow	Collection of movies \uparrow	Similarity to ML \uparrow	Agg. score \uparrow
Rating scale	N-shot	System prompt					
0-9	0-shot	default	0.80 \pm 0.00	1.00\pm0.00	0.67 \pm 0.02	0.54 \pm 0.00	0.75 \pm 0.01
0-9	0-shot	custom	0.87\pm0.00	1.00\pm0.00	0.68 \pm 0.02	0.70 \pm 0.00	0.81 \pm 0.01
0-9	1-shot	default	0.72 \pm 0.00	0.96 \pm 0.00	0.71\pm0.03	0.73 \pm 0.01	0.78 \pm 0.01
0-9	1-shot	custom	0.81 \pm 0.00	1.00\pm0.00	0.71\pm0.02	0.78 \pm 0.00	0.82\pm0.01
0-9	2-shot	default	0.78 \pm 0.00	0.99 \pm 0.00	0.69 \pm 0.01	0.80\pm0.00	0.82\pm0.00
0-9	2-shot	custom	0.79 \pm 0.00	1.00\pm0.00	0.67 \pm 0.03	0.78 \pm 0.00	0.81 \pm 0.01
1-10	2-shot	custom	0.50 \pm 0.0	0.50 \pm 0.00	0.50 \pm 0.00	0.51 \pm 0.00	0.50 \pm 0.00
one-ten	2-shot	custom	0.79 \pm 0.00	1.00\pm0.00	0.66 \pm 0.01	0.72 \pm 0.00	0.79 \pm 0.00

Table 6: Ablation results for the book environment using *Mistral 7B* as our environment. We test the performance of the LLM to give coherent and realistic ratings for user-book interactions. We achieve best overall performance when using 2-shot prompting and our custom system prompt.

Prompt component			Category \uparrow	High/low \uparrow	Collection of books \uparrow	Agg. score \uparrow
Rating scale	N-shot	System prompt				
1-5	0-shot	default	0.68 \pm 0.00	1.00\pm0.00	0.65 \pm 0.01	0.77 \pm 0.00
1-5	0-shot	custom	0.83 \pm 0.00	1.00\pm0.00	0.68 \pm 0.02	0.83 \pm 0.01
1-5	1-shot	default	0.87 \pm 0.00	1.00\pm0.00	0.81 \pm 0.04	0.89 \pm 0.01
1-5	1-shot	custom	0.89\pm0.00	1.00\pm0.00	0.82\pm0.02	0.90\pm0.01
1-5	2-shot	default	0.83 \pm 0.00	0.98 \pm 0.00	0.73 \pm 0.02	0.85 \pm 0.01
1-5	2-shot	custom	0.85 \pm 0.00	1.00\pm0.00	0.76 \pm 0.02	0.87 \pm 0.01

C ABLATIONS

C.1 Test Cases

To determine the effect of each component in our environment, we performed ablations across four different test cases.

Genres/Categories. We assess the environment’s ability to recognize movie and book genres, and its ability to correlate those genres with user preferences to accurately predict ratings. User profiles were created manually for each movie genre, ensuring that they express a preference for the selected genre while disliking all others. Afterwards, we queried the environment with users and movies from both their favored and disliked genres. The accuracy of rating predictions is used to measure performance. A similar process is used for the book environment, replacing movie genres with book categories.

For the genre test set, we manually created four distinct users for each genre: action, animation, comedy, documentary, family, fantasy, horror, and romance. These users included two women and two men, with one younger individual and one older individual for each gender. In constructing the user descriptions, we ensured that each person consistently rated a specific genre highly (between 8 and 10) while assigning lower ratings to all other genres (between 1 and 5). The asymmetry in high and low ratings is motivated by research done by Ramos et al. [32], where they analyze rating

behavior on IMDB. We then presented these users with a set of 20 movies from their preferred genres and another 20 movies from genres they disliked. Our evaluation metric was the percentage of successful predictions in these scenarios.

For the book environment, users are created in the same manner with the exception of their genre preferences, which are specific to book categories rather than movies. The available book categories include fiction, biography, economics, health, philosophy, computer, humor, and drama.

Oliver is a 27 years old man, he is a gentle and introspective man, holds a deep affection for animation films. He possesses a keen eye for detail and an appreciation for the craftsmanship that goes into creating animated works. Oliver’s love for animation is evident in his collection of concept art and his fascination with the behind-the-scenes process. Oliver gives only a high rating to animation films, the motivation lies in their ability to convey profound messages in a visually captivating manner. He believes that animation has a unique power to touch the hearts of both children and adults alike. On the other hand Oliver thinks that a film which is not an animated film is not worth watching, since realism is bad for people, for this reason he assigns a low rating (between 1-5) to every film, which is not an animation film.

Listing 12: Example description of a user of the genres test set

High/Low rating. We assess whether the environment can accurately infer high ratings for users who provide positive-leaning descriptions, while inferring low ratings for users whose descriptions are negative-leaning. We give each user a set of items and test whether the environment is able to generate high or low ratings, depending on the description of the user.

We created eight hand-made users: four women and four men, with two young and two elderly individuals of each gender. Within each age group, there is one user who consistently rates items highly and one who consistently rates items low. In this evaluation, we present 160, 20 when using a model based on a paid API (GPT-3.5, GPT-4), items to each of these users, and assess the environment performance by measuring the percentage of successful predictions. The correctness of the environment is determined by its ability to predict high ratings for users whose descriptions explicitly indicate a preference for higher ratings and low ratings for users whose descriptions imply a preference for lower ratings.

Ava is a 80 years old woman, she is an elderly woman finds great pleasure in reading books, as they are her sole source of passion and entertainment. With no other hobbies to occupy her time, she devotes herself entirely to the world of books. As a token of her appreciation for the writers, she consistently awards a perfect rating of 5 to express her gratitude.

Listing 13: Example description of a user of the high/low test set

Collection of items. We evaluate the ability of the environment to leverage the historical item ratings of a user to predict their future ratings. We conduct this test by manually selecting a set of item collections belonging to a series (e.g., James Bond, Toy Story, etc.). Subsequently, we randomly select a sample of users from our synthetic dataset and fill their history with items from our collection as well as random items. We assign a high rating to all items in the collection history, and the corresponding average rating to the remaining random items. Success is measured by a high rating for the queried item that is part of the collection. The experiment is repeated by assigning low ratings to the collection items to test the ability of the environment to predict low ratings.

To evaluate the movie environment, we took a selection of 22 movie franchises as our test cases. For each franchise, we sample a set of 100 users from our dataset, 50 when using a model based on a paid API (GPT-3.5, GPT-4). To construct the user histories, we included all movies from the respective franchise, except for one, and filled the histories with additional randomly chosen movies. For every user, we designed two distinct queries for the environment. In the first query, all the movies were ones that the user had rated highly in the past. In contrast, in the second query, the user had assigned low ratings to all the movies. Subsequently, we requested a rating for the movie that had been excluded. For the first type of query, we consider the environment to be successful if the user assigns a high rating (consistent with their previous high ratings). In the second type of query, success is determined by the user assigning a low rating (in agreement with their previous low ratings).

This methodology resulted in the creation of 200 queries for the environment for each franchise. The overall score is calculated

based on the percentage of successful predictions in all tests in the 22 different film franchises.

We tested the book environment in a similar way, with the only exception that we used 20 book collections.

Similarity to real rating distribution. We evaluate whether the rating distribution obtained from our movie environment accurately reflects human behavior by comparing it to the rating distribution from MovieLens, which are representative samples of human ratings. We sample with replacement from both our environment and the MovieLens dataset. We then compute the empirical distribution across the dataset and use the total variation distance as a metric to measure similarity.

To calculate the similarity with the true data distribution of MovieLens, we begin by sampling two datasets, D_E (for our environment) and D_M (for MovieLens), with replacement. Our sampling process is as follows: for the movie environment, we randomly select a user and a movie and request the rating. Similarly, for MovieLens, we start by choosing a movie uniformly at random and then choosing one of its ratings randomly.

Let $D_E = \{(m_1, u_1, r_1), \dots, (m_N, u_N, r_N)\}$ be the sampled dataset from our environment, and let $D_M = \{(m'_1, u'_1, r'_1), \dots, (m'_{N'}, u'_{N'}, r'_{N'})\}$ be the MovieLens dataset. Where a triplet (m, u, r) represents a user's rating of movie m with a score of r .

We compute empirical rating distributions for both MovieLens and the movie dataset from our environment as follows:

$$p_{D_M}(j) = \frac{|\{(m, u, r) \in D_M \mid r = j\}|}{|D_M|}$$

$$p_{D_E}(j) = \frac{|\{(m, u, r) \in D_E \mid r = j\}|}{|D_E|}$$

To compare these two distributions, we calculate the total variation distance between the discrete probability distributions p_{D_M} and p_{D_E} :

$$\delta(p_{D_M}, p_{D_E}) = \frac{1}{2} \|p_{D_M} - p_{D_E}\|_1 = \frac{1}{2} \sum_{j \in [10]} |p_{D_M}(j) - p_{D_E}(j)|. \quad (3)$$

We then compute the similarity using the variation distance as follows:

$$\text{sim}(D_M, D_E) = 1 - \delta(p_{D_M} - p_{D_E}). \quad (4)$$

For the book environment, we find that existing datasets are too biased toward high ratings to be good candidates, so the aggregated score does not include this test case.

Aggregated score. The aggregated score is the mean of all test cases. All ablations, except where defined otherwise, were performed using the following configurations. We used the 2-shot prompting, a custom system prompt, three item retrieval via T5-similarity, and no reward perturbation. For movies, we used *Mistral 7B* with rating scale 0-9, and for books we use *Mistral 7B* with scale 1-5.

C.2 Setup

We implemented three different approaches for the retrieval component: feature retrieval, recency retrieval, and similarity retrieval. The feature-based approach retrieves items based on the Sorensen coefficient of movie genres, actors, director, and average rating,

while for books, we use book category, author, and average rating. The recency algorithm retrieves the most recent interactions. The similarity approach retrieves items from the history based on their similarity to the query item. We generate item-description embeddings using a Sentence-T5 model [25] and compute their similarities based on the cosine distance. To select the item-rating pair to retrieve from memory, we compute the similarity between the query item and all items previously viewed by the current user, selecting the items with the highest similarity.

Furthermore, we experimented with one-shot and two-shot prompting to improve model performance, which has been shown to increase generation quality [6]. In addition to the default system prompt, we created a custom system prompt see Appendix A for movie and Appendix B for books.

Tokenization ambiguity can become an issue when generating numbers with LLMs. Since we are dealing with ratings on a scale from one to ten, and because the number “10” can be tokenized in two different ways, this can cause unwanted side effects. To tackle this challenge, we tested two additional strategies for the movie setting: shifting all rewards to the scale of 0-9, and using words for numbers from “one” to “ten.”

Since different LLMs influence the simulation of human behavior in different ways, it is important to highlight the inherent trade-off between model size and processing speed. In particular, during training an RL model, a fast environment is desirable to acquire more samples in a shorter time span. However, smaller LLMs may not adequately emulate the desired human behavior of our synthetic users.

For the reward perturbation experiment, we compared Gaussian noise and greedy noise. Greedy noise alters the LLM rating by 1 with a probability of q , while it remains unchanged with a probability of $1 - q$.

C.3 Results

In the movie environment, we observe that different prompt strategies generally do not differ significantly from each other in the case of Mistral, with the only two exceptions being the 0-shot prompt with the default system prompt, which performs slightly worse, and the weak performance of the 1-10 rating scale due to tokenization ambiguity. Vicuna, on the other hand, is more affected by different prompt strategies, as shown in Appendix D. Table 5 shows the general trend on how the environment is able to capture human concepts such as genres and movie franchises. For the book environment, see Appendix D. In general, we observe that larger models perform better across model families (cf. Figure 5). In addition, we can see how Mistral performs best among open-source models.

Our ablation of the retrieval component demonstrates that this component plays a crucial role in understanding user interests (cf. Table 7). Furthermore, the recency approach proves inadequate, while the best-performing retrieval approach is predicated on the similarity of item features.

Table 7: Comparison of different retrieval strategies on the *collection of items* test set for movies and books. We only show the entries of the test case where the result changed significantly, for extended results see Appendix D.

Retrieval component	Collection of movies \uparrow	Collection of books \uparrow
None	0.49 \pm 0.01	0.50 \pm 0.00
Most recent	0.62 \pm 0.01	0.64 \pm 0.01
T5 similarity	0.67 \pm 0.03	0.76 \pm 0.02
Feature similarity	0.68\pm0.01	0.79\pm0.01

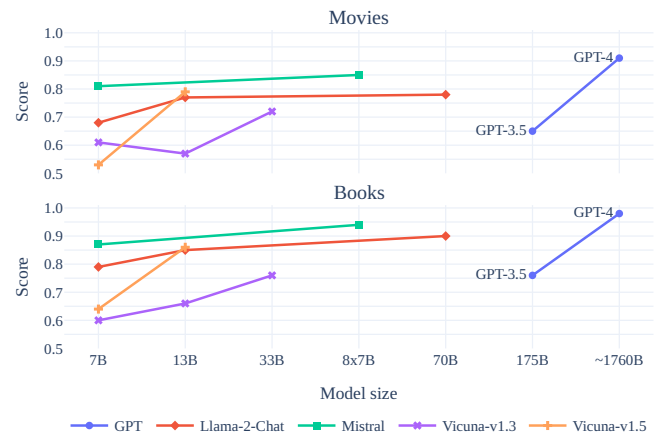


Figure 5: Aggregated score across LLM families for the movie environment (top), and for the book environment (bottom) by varying only the LLM component. For details see Appendix D.

D EXTENDED ABLATIONS RESULTS

In the following section, we present additionally results for both the movie and book environments. We show more detailed scores for the different test set of the ablation for the different models, as well as the impact of the different perturbator and retrieval component on the scores of *Mistral 7B*. Additionally we provide more detail on the impact of different prompting strategies for *Vicuna-v1.5-13B* in the movie setting

Table 8: Ablation results for the movie environment using the following settings: 0-9 rating scale, 2-shot, custom system prompt, T5-similarity movie retrieval, and perturbation none. For each prompt component, we show the aggregated score and specific subscores for the various test cases.

LLM	Size	Genres \uparrow	High/Low \uparrow	Collection of movies \uparrow	Similarity to ML \uparrow	Agg. score \uparrow
GPT-4	1760B	0.96±0.00	1.00±0.00	0.98±0.00	0.69±0.04	0.91±0.01
GPT-3.5	175B	0.66±0.00	0.94±0.00	0.50±0.00	0.49±0.02	0.65±0.00
Llama-2-Chat	70B	0.80±0.00	1.00±0.00	0.67±0.01	0.66±0.00	0.78±0.00
	13B	0.76±0.00	1.00±0.00	0.72±0.01	0.60±0.00	0.77±0.00
	7B	0.55±0.01	1.00±0.00	0.72±0.02	0.47±0.00	0.68±0.00
Vicuna-v1.3	33B	0.48±0.00	1.00±0.00	0.78±0.02	0.64±0.00	0.72±0.01
	13B	0.42±0.01	0.62±0.02	0.64±0.01	0.59±0.00	0.57±0.01
	7B	0.37±0.00	0.85±0.01	0.61±0.02	0.61±0.00	0.61±0.01
Vicuna-v1.5	13B	0.69±0.00	1.00±0.00	0.77±0.02	0.68±0.00	0.79±0.01
	7B	0.44±0.00	0.34±0.00	0.73±0.04	0.63±0.00	0.53±0.01
Mistral	8x7B	0.81±0.01	1.00±0.00	0.78±0.03	0.82±0.00	0.85±0.01
	7B	0.79±0.00	1.00±0.00	0.67±0.03	0.78±0.00	0.81±0.01

Table 9: Ablation results for the movie environment using the following settings: LLM Mistral 7B, 0-9 rating scale, 2-shot, custom system prompt, and perturbation none. For each prompt component we show the aggregated score and specific sub-scores for the various test cases.

Retrieval component	Genres \uparrow	High/Low \uparrow	Collection of movies \uparrow	Similarity to ML \uparrow	Agg. score \uparrow
Features similarity	0.79±0.01	1.00±0.00	0.68±0.01	0.78±0.00	0.81±0.00
T5 similarity	0.79±0.00	1.00±0.00	0.67±0.03	0.78±0.00	0.81±0.01
Most recent	0.80±0.00	1.00±0.00	0.62±0.01	0.78±0.00	0.80±0.00
None	0.80±0.00	1.00±0.00	0.49±0.01	0.78±0.00	0.77±0.00

Table 10: Ablation results for the movie environment using the following settings: LLM Mistral 7B, 0-9 rating scale, 2-shot, custom system prompt, T5-similarity movie retrieval. For each prompt component we show the aggregated score and specific sub-scores for the various test cases.

Perturbator component	Genres \uparrow	High/Low \uparrow	Collection of movies \uparrow	Similarity to ML \uparrow	Agg. score \uparrow
gaussian	0.79±0.00	1.00±0.00	0.68±0.01	0.82±0.00	0.82±0.00
greedy	0.78±0.01	1.00±0.00	0.67±0.01	0.81±0.00	0.82±0.00
none	0.79±0.00	1.00±0.00	0.67±0.03	0.78±0.00	0.81±0.01

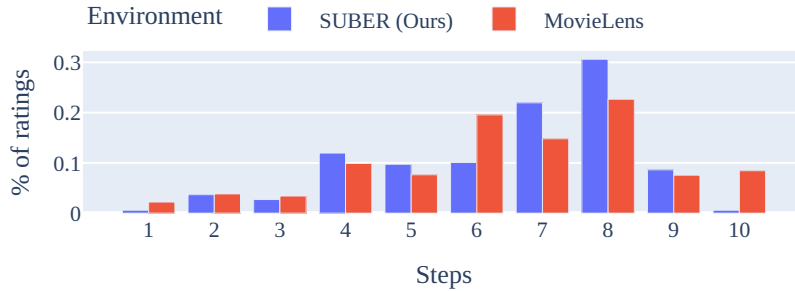


Figure 6: Rating distribution for SUBER movie environment is shown in blue, while the distribution for MovieLens is displayed in red.

Table 11: Ablation results for the movie environment using *Vicuna-v1.5-13B* as our environment. We test the LLM on coherency and realistic ratings for user-movie interactions. We achieve best performance with 0-9 digit rating scale, 2-shot prompting, and our custom system prompt.

Rating scale	Prompt component		Genres \uparrow	High/Low \uparrow	Collection of movies \uparrow	Similarity to ML \uparrow	Agg. score \uparrow
	N-shot	System prompt					
0-9	0-shot	default	0.65 \pm 0.00	0.99 \pm 0.00	0.62 \pm 0.02	0.64 \pm 0.00	0.72 \pm 0.00
0-9	0-shot	custom	0.69 \pm 0.00	0.99 \pm 0.00	0.64 \pm 0.02	0.65 \pm 0.00	0.74 \pm 0.00
0-9	1-shot	default	0.61 \pm 0.00	1.00\pm0.00	0.71 \pm 0.01	0.75\pm0.00	0.77 \pm 0.00
0-9	1-shot	custom	0.72\pm0.00	1.00\pm0.00	0.74 \pm 0.03	0.74 \pm 0.00	0.80 \pm 0.01
0-9	2-shot	default	0.63 \pm 0.01	1.00\pm0.00	0.81 \pm 0.02	0.74 \pm 0.00	0.80 \pm 0.00
0-9	2-shot	custom	0.69 \pm 0.00	1.00\pm0.00	0.82\pm0.02	0.75\pm0.00	0.81\pm0.00
1-10	2-shot	custom	0.64 \pm 0.01	0.72 \pm 0.03	0.68 \pm 0.01	0.72 \pm 0.00	0.69 \pm 0.01
one-ten	2-shot	custom	0.71 \pm 0.01	1.00\pm0.00	0.72 \pm 0.03	0.64 \pm 0.00	0.77 \pm 0.01

Table 12: Ablation results for the book environment using the following settings: 1-5 rating scale, 2-shot, custom system prompt, *T5-similarity* book retrieval, and *no perturbation*. For each prompt component we show the aggregated score and specific sub-scores for the various test cases.

LLM	Size	Category \uparrow	High/low \uparrow	Collection of books \uparrow	Agg. score \uparrow
GPT-4	1760B	0.96\pm0.00	1.00\pm0.00	0.97\pm0.00	0.98\pm0.00
GPT-3.5	175B	0.65 \pm 0.00	0.99 \pm 0.00	0.63 \pm 0.00	0.76 \pm 0.00
Llama-2-Chat	70B	0.95 \pm 0.00	1.00\pm0.00	0.74 \pm 0.02	0.90 \pm 0.01
	13B	0.76 \pm 0.00	1.00\pm0.00	0.78 \pm 0.06	0.85 \pm 0.02
	7B	0.65 \pm 0.01	1.00\pm0.00	0.73 \pm 0.02	0.79 \pm 0.01
Vicuna-v1.3	33B	0.63 \pm 0.00	0.95 \pm 0.00	0.71 \pm 0.00	0.76 \pm 0.00
	13B	0.51 \pm 0.02	0.87 \pm 0.02	0.58 \pm 0.01	0.66 \pm 0.02
	7B	0.45 \pm 0.03	0.74 \pm 0.01	0.57 \pm 0.02	0.60 \pm 0.02
Vicuna-v1.5	13B	0.75 \pm 0.01	0.99 \pm 0.00	0.83 \pm 0.03	0.86 \pm 0.01
	7B	0.56 \pm 0.01	0.66 \pm 0.02	0.72 \pm 0.02	0.64 \pm 0.01
Mistral	8x7B	0.92 \pm 0.00	1.00\pm0.00	0.89 \pm 0.01	0.94 \pm 0.00
	7B	0.85 \pm 0.00	1.00\pm0.00	0.76 \pm 0.02	0.87 \pm 0.01

Table 13: Ablation results for the book environment using the following settings: LLM *Mistral 7B* 1-5 rating scale, 2-shot, custom system prompt, and *no perturbation*. For each prompt component we show the aggregated score and specific sub-scores for the various test cases.

Retrieval component	Category \uparrow	High/low \uparrow	Collection of books \uparrow	Agg. score \uparrow
None	0.85 \pm 0.00	0.99 \pm 0.00	0.50 \pm 0.00	0.78 \pm 0.00
Most recent	0.86\pm0.00	1.00\pm0.00	0.64 \pm 0.01	0.83 \pm 0.00
T5 similarity	0.85 \pm 0.00	1.00\pm0.00	0.76 \pm 0.02	0.87 \pm 0.01
Features similarity	0.85 \pm 0.00	1.00\pm0.00	0.79\pm0.02	0.88\pm0.01

Table 14: Ablation results for the book environment using the following settings: LLM *Mistral 7B* 1-5 rating scale, 2-shot, custom system prompt, *T5-similarity* book retrieval. For each prompt component we show the aggregated score and specific sub-scores for the various test cases.

Perturbator component	Category \uparrow	High/low \uparrow	Collection of books \uparrow	Agg. score \uparrow
None	0.85\pm0.00	1.00\pm0.00	0.76\pm0.02	0.87\pm0.01
Greedy	0.82 \pm 0.00	0.95 \pm 0.00	0.76\pm0.05	0.84 \pm 0.02
Gaussian	0.81 \pm 0.00	0.91 \pm 0.00	0.75 \pm 0.02	0.83 \pm 0.01

E DETAILS HUMAN EVALUATION

Question:

Samuel is a 24 years old man, he is a student who loves chemistry and watch movies on his free time. His favorite genres are drama, comedy, action and crime. He enjoys watching movies that make him think and feel, and that have a good plot and good acting. He likes to watch movies that are based on true events or that have a good message. He dislikes science fiction, horror and documentary movies, as they are too unrealistic or too intense for him. He likes to watch movies with his friends and classmates, and he always looks for a good laugh or a good cry. Samuel is a very analytical person, who loves to read and research about different topics. He is very focused on his studies, and he is always looking for new challenges. He is a good listener and a good conversationalist, and he likes to hear different perspectives.

Samuel has previously watched the following movies (in parentheses are the ratings he gave on a scale of 1 to 10): "Blade Runner 2049" (5), "Inception" (10), "A Beautiful Mind" (9).

Consider the movie "E.T. the Extra-Terrestrial", released in 1982, which is described as follows: an alien is left behind on Earth and saved by the 10-year-old Elliot who decides to keep him hidden in his home. While a task force hunts for the extra-terrestrial, Elliot, his brother, and his little sister Gertie form an emotional bond with their new friend, and try to help him find his way home. The movie "E.T. the Extra-Terrestrial" contains the following genres:

- science fiction
- adventure
- family
- fantasy

Here are the 2 main actors of the movie, in order of importance: Henry Thomas (M), Drew Barrymore (F). On average, people rate the movie "E.T. the Extra-Terrestrial" 7.5 on a scale of 1 to 10. Samuel watches the movie "E.T. the Extra-Terrestrial" for the 1st time.

What can you conclude about Samuel's rating for the movie "E.T. the Extra-Terrestrial" on a scale of 1 to 10, where 1 represents a low rating and 10 represents a high rating, based on available information and logical reasoning?

Answer:

Based on Samuel's preferences and tastes, I conclude that he will assign a rating of 8 on a scale of 1 to 10, because

- 1) He enjoys watching movies that make him think and feel, and have good plots and acting, which is evident from his previous ratings of "A Beautiful Mind" (9) and "Inception" (10);
- 2) "E.T. the Extra-Terrestrial" has elements of adventure and family, which align with his preferred genres; and
- 3) Although it belongs to the science fiction genre, which Samuel generally dislikes, the film's emotional themes and focus on friendship could resonate well enough with him to warrant a higher rating.

- (4) The rating and explanation make sense, but not all important information is considered.
- (5) The rating and explanation perfectly match the user's interests and you completely agree with the rating.

We selected 10 different queries, 4 from the genre test set of the ablation study, 2 from the movie franchise test set of the ablation study, and the remaining 4 we randomly sampled user and movie while manually filling in the previous ratings of the selected users. We successively queried each model with all ten questions using the default configuration (2 shot prompt and custom system prompt with no perturbator), this way we collected 50 different answers, ten for each of the five models. We perform the study by asking 14 participants to score the LLM interactions. In the questionnaire, the order of the questions is randomized for each participant. We then aggregate the score by averaging each model.

Participants in this study were recruited from among our colleagues and provided informed consent to participate. The study was designed with strict adherence to randomized double-blind procedures to ensure impartiality and reliability of the results. As this user study did not involve ongoing follow-up or monitoring of the participants, our institutional review board (IRB) determined that formal approval was not required. From the survey (cf. Table 3), we find that users agree most with *GPT-4*, outperforming all other models. Furthermore, we find that *Mistral 7B* is the best LLM among open-source models despite only having 7B parameters.

Listing 14: Example LLM interaction of the human evaluation

In the human evaluation, we show participants pairs of prompt and LLM response as shown in Listing 14 and ask them to assign a score between 1 and 5 according to the following guideline:

- (1) The LLM's answer does not make sense.
- (2) The rating and explanation do not match the person's interests and preferences.
- (3) The answer makes some sense, but a different rating would be more appropriate.

F RL MODELS

For A2C and PPO we implemented the actor based on the principles of low-rank approximation [2]. For each user u within the set U , we maintain a feature vector e_u . Similarly, for each movie m , we use its feature vector e_m and bias b_m . In addition, we introduce the movie embedding matrix E , and the bias vector b . The probability of recommending movie m to user u is calculated as follows:

$$\text{softmax}(A + E \cdot e_u + b)_m, \tag{5}$$

where A serves as a mask to assign a probability of zero to movies that user u has already viewed. In other words, the entry A_m is set to negative infinity if user u has previously watched movie m . We employ A2C [23] to train the agent. The actor network, which is responsible for recommending movies, samples actions according to Equation (5), while the critic consists of a basic two-layer neural network, which takes the user together with the past movie ratings of users as input. We train the model with the default configuration of SB3 [30] for 1.6M steps on SUBER. All parameters are default, except for gamma, which is changed to 0.975. For TRPO and DQN the actor network additionally takes as input the past movie ratings. Also in this case, the models were trained for 1.6M steps with the default configuration of SB3, and all parameters are default except for gamma, which is changed to 0.975.

F.1 Genre preference

In this section we present the result of the user genre preference statistic of the top-5 recommendations for A2C, our best performing RL model. Each user in the training dataset has both preferred and disliked movie genres. The trained RL model recommends a list of top-5 movie recommendations for each user. The recommendations are classified into three categories: *liked* (movies matching preferred genres and excluding disliked ones), *disliked* (movies with disliked genres and no preferred ones), and *neutral* (remaining recommendations). Again, we can see that the majority of recommendations fall into the *liked* category.

We observe from Figure 7 how A2C is the model that learns better our environment. Moreover, as shown in Figures 8 to 11, the RL recommender model is able to learn the dynamics of the user’s genre preferences, primarily recommending movies that fall into the user’s favorite genres. It should be noted that recommending a *neutral* movie can be a valid strategy, especially if it is a highly praised or outstanding movie.

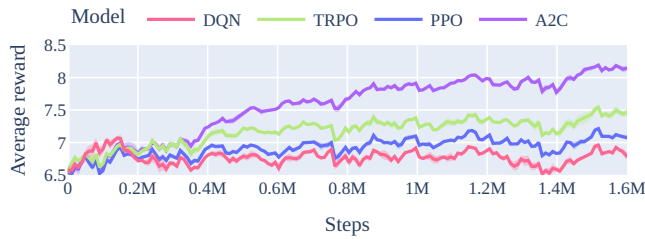


Figure 7: Training plot of various RL models. The y-axis displays the average reward from evaluation samples.

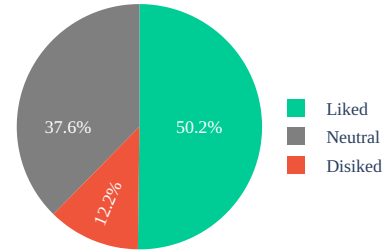


Figure 8: User genre preference statistic of top-5 movie recommendations generated by the RL model, trained with A2C.

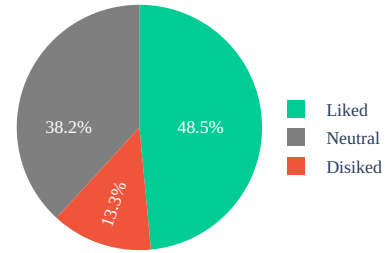


Figure 9: User genre preference statistic of top-5 movie recommendations generated by the RL model, trained with TRPO.

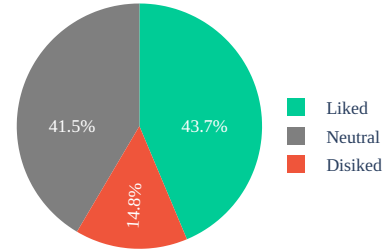


Figure 10: User genre preference statistic of top-5 movie recommendations generated by the RL model, trained with PPO.

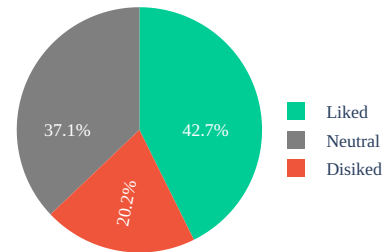


Figure 11: User genre preference statistic of top-5 movie recommendations generated by the RL model, trained with DQN.

Table 15: At the top we show the personal interest of Max. At the bottom, we show the title and genres of the first 5 recommended movies.

Name	Max
Liked genres	thriller, documentary, fantasy, crime
Disliked genres	romance
Recommended movie	Genres
The Walk	adventure, drama, thriller, history
The Testament of Dr. Mabuse	crime, mystery, thriller
The Island	action, thriller, science fiction, adventure
Captain America	action, adventure, science fiction, war
The Count of Monte Cristo	adventure, drama, history

Table 16: At the top we show the personal interest of Max. At the bottom, we show the title and genres of the first 5 recommended movies.

Name	Ava
Liked genres	drama, science fiction, animation, adventure
Disliked genres	romance, fantasy, crime, comedy
Recommended movie	Genres
Inception	action, science fiction, adventure
The Walk	adventure, drama, thriller, history
Spartacus	history, war, drama, adventure
Exodus: Gods and Kings	adventure, drama, action
Cliffhanger	action, adventure, thriller
Atragon	science fiction, action, adventure, fantasy

Table 17: At the top we show the personal interest of Max. At the bottom, we show the title and genres of the first 5 recommended movies.

Name	Maya
Liked genres	drama, science fiction, documentary, comedy
Disliked genres	horror, animation
Recommended movie	Genres
The One-Armed Swordsman	action, drama
The One I Love	romance, comedy, drama
In the Heart of the Sea	thriller, drama, adventure, action, history
Detachment	drama
Amadeus	history, music, drama
Margaret Cho: I'm the One That I Want	comedy

F.2 Recommended movies examples

For a set of random users, we interact with the trained RL model, in Tables 15 to 17 with embedding dim 32, and show the first 5 recommended movies.

Table 18: List of Large Language Models tested on the environment. iterations/seconds are computed for all models using GPTQ and Exllama on a RTX3090, and A100-40GB for Llama-2-70B.

Model name	Size	Contex length	iterations/s (in our env)
GPT-4	1760B	8k / 32k	API ratelimit dependent
GPT-3.5	175B	4k / 16k	
Llama-2-Chat	70B	4,096	1.6
	13B		5
	7B		6
Vicuna-v1.3	33B	2,048	3
	13B		5
	7B		6
Vicuna-v1.5	13B	4,096	5
	7B		6
Mistral	8x7B	4,096	2
	7B		6

F.3 Environment performance

In Table 18 we show, for different LLMs, the size and performance (iterations in seconds) of our environment.

G LIST OF HOBBIES AND PROFESSIONS

For the data see our code:

- Hobbies: environment/users/datasets/hobby_list.csv
- Hobbies for childrens: environment/users/datasets/hobby_children.csv
- Professions: environment/users/datasets/job_list.csv