ShiftScope: Adapting Visualization Recommendations to Users' Dynamic Data Focus

Sanad Saha*
Oregon State University
sahasa@oregonstate.edu

Nischal Aryal*
Oregon State University
aryaln@oregonstate.edu

Leilani Battle
University of Washington
leibatt@cs.washington.edu

Arash Termehchy
Oregon State University
termehca@oregonstate.edu

ABSTRACT

Visualization Recommendation Systems help users discover important insights during data exploration. These systems should understand users' exploration behaviors and goals to suggest relevant visualizations. However, users' mental models constantly evolve as they learn more about their data or as their personal or organizational goals change, leading to shifts in their data focus. Current systems do not adapt to these changes; therefore, they may inevitably suggest irrelevant visualizations over time. Thus, we introduce ShiftScope, an interactive system that recommends personalized visualizations while adapting to users' conceptualization of data. ShiftScope utilizes a dual-agent reinforcement learning framework, where one agent adapts to evolution in data focus and collaborates with the other agent to recommend the best visualizations to satisfy users' current and future exploration needs.

KEYWORDS

Exploratory Data Analysis, User Modeling, Reinforcement Learning, Personalized Visualization Recommendation

ACM Reference Format:
Sanad Saha, Nischal Aryal, Leilani Battle, and Arash Termehchy. 2018. ShiftScope: Adapting Visualization Recommendations to Users' Dynamic Data Focus. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

As datasets continue to grow, data-driven discovery becomes paramount for solving pressing societal and business challenges. This interactive process involves users exploring new datasets to uncover crucial and interesting insights [1, 7]. It is particularly challenging for users performing open-ended exploration tasks with vague exploration goals that give them minimal guidance on their objectives and how to achieve them. Thus, users do the hard lifting of generating hypotheses, analyzing the data, and deciding how to proceed with the exploration problem (Figure 1). As users gain insights from their data interactions/queries over time, what they want to learn changes, dictating a shift in users' data focus.

Visualization recommendation (VisRec) tools alleviate the insight discovery process for users by suggesting and personalizing visualizations suited to their intent or task[1, 5]. The predominant methods for inferring user intent involve building a taxonomy of intents, i.e., tasks, and predicting which of these predefined tasks the user will switch to next. Then, visualization tools provide recommendations for what to explore next. However, task taxonomies are too brittle to account for spontaneous tasks that may appear during data exploration. Additionally, current VisRec systems fail to account how the user's understanding of the data, and thus, their data focus evolves over time, causing the underlying models and task taxonomies to quickly become outdated.

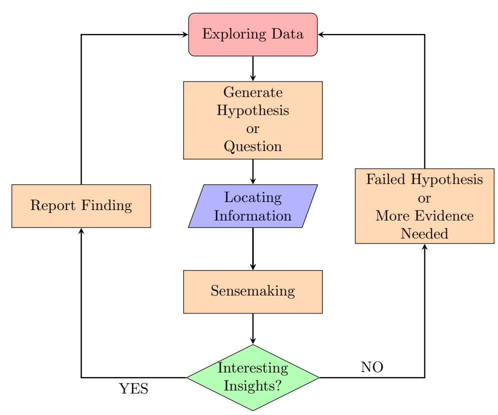

Figure 1: Example of an exploration approach

During exploration, users may learn about the data and change their exploration strategies to be more effective. For instance, users may revise their prior beliefs [14], shift their focus to different data attributes [9], and change the queries they submit to the system [8]. Similarly, users may deepen their understanding of the data through visualization recommendations at different steps of exploration (Figure 1), bringing the user to a new goal or insight [3]. The following scenario illustrates how users *shift their data focus*, i.e., change their data selection and transformation preferences, based on what they learn over time.

<u>*Scenario:*</u> Alice and Bob are exploring the Birdstrikes dataset [15] containing records of wildlife strike incidents with aircrafts. Alice

*Both authors contributed equally to this research.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

is exploring opportunities to improve aviation safety and reduce the losses incurred by airlines. Bob is researching the current safety of wildlife behavior and migration patterns. Both explore the same visualizations for their first three interactions at time t, $t+1$ and $t+2$ (see Figure 2). Interaction t reveals the number of strikes per month, $t+1$ displays the total strikes at different times of the day (e.g., Day, Dusk, Night), and $t+2$ depicts total strikes in different phases of flight (e.g., Climb, Approach).

However, Alice and Bob start to deviate at time $t+3$. Alice analyzes the Total **Cost** incurred for strikes happening at different phases of flight whereas Bob investigates the number of strikes involving different **Sizes** of wildlife (e.g., Small, Medium, Large).

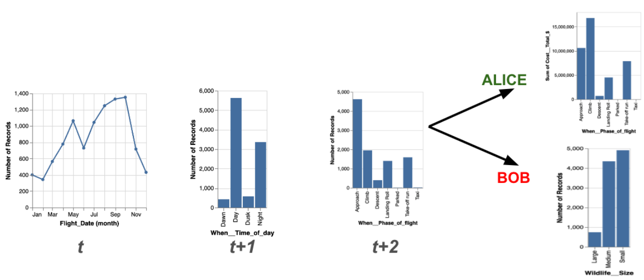

Figure 2: User interaction flow

Shifts in Data Focus: Alice spends more time on *interactions t+2* and *t+3* and learns: 'Although the number of incidents during Approach is higher, incidents during the Climb phase are more costly to the airlines'. Thus, she will focus more on the attribute **Cost**, rather than 'Number of records' in the upcoming phases.

In contrast, Bob spends more time on *interaction t*. He relates the high number of birdstrikes in September and October with birds' migratory patterns. He is not interested in the costs of airlines but rather wants to know which species are more at risk. So, he prioritizes visualizations involving attributes such as 'wildlife_species' and 'wildlife_size' (e.g., *interaction t+3 (Bob)*).

Challenges: Current techniques assume we can pre-train models to learn all relevant tasks and behavioral patterns offline. For example, rule-based [11] or ML-based [5] VisRec systems either statistically measure or utilize learned models to find the best possible visualizations given users' current state. However, data exploration is inherently dynamic. So, these models will fail during tasks they have never seen before, which is quite common during data exploration. So, relying on static methods that recommend visualizations based on common or popular features may not accommodate sudden shifts in users' data focus and aid users' long-term objectives.

Our approach: Rather than trying to model every aspect of the user's learning process, which is out of scope for this paper, we focus on modeling a key aspect of the user's cognition: when they *shift their data focus*. In this way, we move beyond rigid task prediction towards understanding *when* and *how* the user shifts their focus to different parts of the data.

In this work, (1) we model users' choice of attribute for visualization generation as indicators of their data focus during exploration. ShiftScope employs a state-of-the-art reinforcement learning (RL) agent to model users' attribute preferences. Learning from the user's past feedback and exploration operations, this user model agent can adapt and simulate a particular user's future data focus.

(2) We formalize exploratory data analysis using a dual-agent Markov decision process framework. The user agent lets the RL-based recommender agent know about potential shifts in users' data focus. In addition to this information, the recommender agent uses explicit user feedback on past recommendations to recommend six visualizations. Both agents collaborate so that one can accurately model the data shifts, and the other can utilize that to generate visualizations. ShiftScope displays the data focus simulations for the user as visualization recommendations.

2 SYSTEM DESCRIPTION

2.1 Technical Background

2.1.1 Exploratory Data Analysis (EDA). EDA initiates when the user loads a dataset into ShiftScope. It recommends some *default visualizations* to avoid a cold start, i.e., starting the exploration without any prior goal or knowledge. These recommendations are designed to encapsulate interesting starting points derived from statistical properties in the dataset and other users' interactions. It helps the user to start the EDA process as in Figure 1.

In ShiftScope, users can either select a recommended visualization or construct a new one for analysis. This chosen/created visualization plays a pivotal role in shaping both the user's subsequent data focus as it acts as the basis of ShiftScope's future recommendations.

2.1.2 Markov Decision Process (MDP). MDP is a framework for modeling decision-making, where an agent learns to achieve its goals through repeated interactions with an environment (interface). In each interaction, the agent takes action based on its current state (visualization), and the environment responds with a reward (feedback). The agent uses these rewards to update its policy and make better decisions in the future. In MDP, the agent's objective is to find an optimal **policy** (π), which is a function that maps each state to an action that maximizes the expected future reward.

2.2 Dual-Agent Architecture

2.2.1 Formalized MDP. Our dual-agent MDP can be represented using a tuple $< S, \alpha_{\{i \in \{U,S\}\}}, A_{i\in\alpha}, \pi_{i\in\alpha}, \Omega_{i\in\alpha}, \tau, R_{i\in\alpha} >$, where S is a discrete set of states shared by our two agents (α). User-agent ($\alpha = U$) learns evolution in the user's data focus, and the recommender agent ($\alpha = RL$) generates recommendations conditioned on this evolution. A_α represents the set of actions available to each agent. $\tau : S \times A_U \times A_{RL} \times S \longrightarrow [0,1]$ represents the transition matrix, which determines the next state after all agents perform actions on the current state. R_α is the reward received by each agent after performing an action, and what they observe in the environment is Ω_α. Formalizing EDA as an MDP enables us to use state-of-the-art RL algorithms for the decision-making process of our agents.

2.2.2 Defining MDP components in ShiftScope:

States (S): The state corresponds to the current visualization the user interacts with, which is generated using Vega Lite [13] specifications. In this work, we represent the state by converting key visualization information, such as the mark type, selected attributes, attribute types, and applied aggregations, into a one-hot vector.

User Agent's Actions (A_U): User-agent's action involves selecting

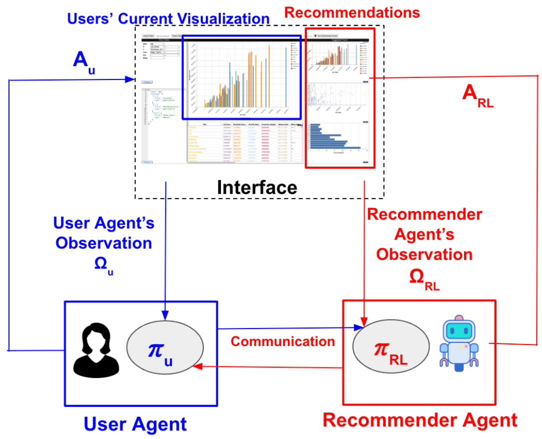

Figure 3: Dual-Agent Collaborative Framework

the attributes for the *X*, *Y*, and *Color* fields.

User Agent's Reward (R_U): The user agent gets rewarded if it can correctly predict attribute preferences, which leads the recommender agent to generate informative visualizations. However, it is challenging to quantify R_U. In EDA, it is not feasible for the user to let us know what data area they would like to focus on after every interaction. However, actions like bookmarks and satisfying-clicks (SAT-clicks) give explicit insight into the usefulness of particular recommendations. Moreover, this feedback is sparse, and to quickly adapt to users' needs, it would be beneficial to get feedback after every step on the recommendations. So, we take proxy measures (e.g., KL-divergence[2]) to determine the effectiveness/interestingness of the recommended attributes and derive R_U.

User Agent's Policy (π_U): ShiftScope focuses on users' evolving data focus, given the current state S, π_U stochastically determines a possible set of attributes. These attributes may generate visualizations that users may find interesting to analyze in future interactions. π_U is updated based on the feedback from the recommender on user agent's performance in detecting data shifts and its own observations from the environment.

Recommender Agent's Actions (A_{RL}): Recommending *six* visualizations to the user. However, on a low level, the recommender needs to predict a set of attributes for these six visualizations. Leveraging information recieved from U ensures the recommendations entail six potential directions where users' data focus may shift.

Recommender Agent's Reward (R_{RL}): The recommender receives rewards (in the form of bookmarks, SAT-clicks, etc.) reflecting users' satisfaction with the recommendations. Furthermore, the recommender considers users' interactions, such as clicks and hovers over visualization items, as forms of reward. Early user feedback can be noisy, so ShiftScope employs parameters to also include default visualizations (subsubsection 2.1.1) which are more likely to provide exploration guidance in early stages.

Recommender Agent's Policy (π_{RL}): Given current state S,π_{RL} stochastically determines the best visualizations to recommend. π_U is updated based on the attribute preferences suggested by the user model, user feedback, and recommender agent's observation (Ω_{RL}).

Observation (Ω): Both the user and recommender agents monitor user interactions within the interface, utilizing observations like clicks, hovers, bookmarks, etc., to refine their respective policies.

2.3 Collaboration and Algorithm

Without collaborating with the user agent, the recommender agent lacks the ability to generate insightful visualizations that align with the user's evolving data focus. If it fails to do so, it cannot garner positive feedback from the user, causing both agents to encounter two significant challenges: firstly, the recommender lacks the necessary guidance to create visualizations that align with the user's specific informational needs; secondly, the user agent renders ineffective, thereby undermining the recommender's capacity to produce recommendations tailored to the user's focused data areas.

In ShiftScope, both the user agent and the recommender agent is modeled using Proximal Policy Optimization (PPO), an actor-critic reinforcement learning algorithm variant. Recent studies have highlighted the advantages of PPO in modeling human learning within collaborative environments[4]. Leveraging available information from the interface and user feedback (subsubsection 2.2.2), the PPO-based user-agent model successfully adapts to users' data focus and lays the foundation for the recommender to recommend adaptive visualizations. Additionally, PPO's popularity in systems that incorporate human feedback is well-established [4].

Effectively generating visual encodings from data is essential for users to comprehend information based on selected attributes. To achieve this, we employ the state-of-the-art tool Draco [10], which is embedded with visualization design knowledge. Draco aids in navigating the visualization design space, thereby enabling the identification of optimal encodings for the selected attributes.

3 DEMONSTRATION

3.1 Dataset and Exploration Tasks

We will begin with an overview of our system, dataset, and exploration tasks. We will utilize two Voyager [15] datasets and 2 open-ended tasks to demonstrate our system.

birdstrikes: The dataset contains incident reports of aircraft (e.g., airplanes) striking wildlife (e.g., deer, birds), with contextual details (e.g., weather conditions, total struck, etc). The dataset is a redacted version of FAA wildlife airplane strike records with 10,000 records and 14 attributes (9 nominal, 1 temporal, 4 quantitative).

Exploration task: Feel free to explore any and all aspects of the data. Use the bookmark features to save any interesting patterns, trends or other insights worth sharing with colleagues. Note the top 5 bookmarks that you found most interesting from your exploration.

movies: The movies dataset contains 3,201 records and 15 attributes (7 nominal, 1 temporal, 8 quantitative). *Exploration task: What kinds of movies will be the most successful movies based on your observations of the data? Summarize the 2-3 characteristics that you believe are most important in predicting their success*

3.2 System Interface

Figure 4 presents our system interface. The top panel (A) provides data imports and exports and provides a quick view of data attributes. The data-panel (B) contains a more detailed view of the dataset. The chart-editor view (C) allows users to steer the exploration. Users can select the data areas to explore. Once users specify the data attributes, the lower panel presents the automatically generated Vegalite specification. Users comfortable with Vegalite can

Figure 4: System Interface

also edit this chart specification. Current-analysis view (D) renders the user-specified visualization. The suggested charts view (E) shows visualizations for data areas relevant to the user's current analysis and allows users to bookmark relevant and interesting charts. Users can click on any of the recommendations and the system automatically populates Vegalite specification and drop-down in (A) and brings the chart to the user's current-analysis view (D) ready for the next exploration step.

3.3 Demonstration Workflow

3.3.1 Interactive System. We will demonstrate 2 versions of our system, both sharing the same interface design (Figure 4) (1) Human-aware system, which incorporates a user model and personalizes its recommendations online (2) offline version of our system with no user model, similar to standard recommendation systems. The participants will be split into two group one using our Human-aware system and the baseline system.

During the exploration tasks, participants will be able to bookmark the recommendations they found useful to complete the tasks. Once the exploration task is completed, participants can report their insights and answers. At the end of the task, we will reveal the identity of their interacted system.

3.3.2 Performance View. To evaluate the effectiveness of our system we will compare average correctness scores of the two groups and overall exploration time. Participants will also get to view a visual flow of their interactions , similar to Fig 2, demonstrating our system's adaptability under diverse exploration behavior and insight needs of different participants. Finally, the participants will get an insider view of the system and examine how the models' policy (π_U,π_{RL}) evolved based on their interactions.

4 RELATED WORKS

Recent research illustrates the necessity of modeling users' exploration strategies [8], prior-belief [14]. We see the use of Bayesian approaches [9] and RL algorithms [12] to model users' data focus. ShiftScope's RL-based user model is aware of users' dynamic shift in data focus. A common theme current VisRec systems adopt is statistically identifying and suggesting the underlying data patterns to users. They generate visualizations based on factors such as *diversity, interestingness, coherency*, etc [1, 2, 5]. However, when utilizing them, they assume these factors have the same contribution, wherein the demand for visualizations based on these elements evolves. Besides, prior studies underscore the need for a framework that combines user interaction history [6] for personalized recommendations and a data-driven approach [2] to facilitate insight discovery and goal formulation. ShiftScope integrates a data-driven approach with effective use of past interactions. VisRec systems such as DashBot [5] and ATENA [2] have popularized the use of Deep-RL to aid users in finding interesting aspects of the dataset. However, these approaches do not directly incorporate users' preferences or learning as reinforcement to guide their future decisions.

REFERENCES

[1] Calvin S Bao, Siyao Li, Sarah G Flores, Michael Correll, and Leilani Battle. 2022. Recommendations for visualization recommendations: Exploring preferences and priorities in public health. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–17.
[2] Ori Bar El, Tova Milo, and Amit Somech. 2020. Automatically generating data exploration sessions using deep reinforcement learning. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 1527–1537.
[3] Leilani Battle and Alvitta Ottley. 2023. What Do We Mean When We Say "Insight"? A Formal Synthesis of Existing Theory. *IEEE Transactions on Visualization and Computer Graphics* (2023), 1–14. https://doi.org/10.1109/TVCG.2023.3326698
[4] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. 2019. On the utility of learning about humans for human-ai coordination. *Advances in neural information processing systems* 32 (2019).
[5] Dazhen Deng, Aoyu Wu, Huamin Qu, and Yingcai Wu. 2022. DashBot: Insight-Driven Dashboard Generation Based on Deep Reinforcement Learning. *IEEE Transactions on Visualization and Computer Graphics* (2022), 1–11. https://doi.org/10.1109/tvcg.2022.3209468
[6] Will Epperson, Doris Jung-Lin Lee, Leijie Wang, Kunal Agarwal, Aditya G Parameswaran, Dominik Moritz, and Adam Perer. 2022. Leveraging analysis history for improved in situ visualization recommendation. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 145–155.
[7] Zhicheng Liu and Jeffrey Heer. 2014. The effects of interactive latency on exploratory visual analysis. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2122–2131.
[8] Ben McCamish, Vahid Ghadakchi, Arash Termehchy, Behrouz Touri, and Liang Huang. 2018. The data interaction game. In *Proceedings of the 2018 International Conference on Management of Data*. 83–98.
[9] Shayan Monadjemi, Roman Garnett, and Alvitta Ottley. 2020. Competing models: Inferring exploration patterns and information relevance via bayesian model selection. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 412–421.
[10] Dominik Moritz, Chenglong Wang, Gregory Nelson, Halden Lin, Adam M. Smith, Bill Howe, and Jeffrey Heer. 2019. Formalizing Visualization Design Knowledge as Constraints: Actionable and Extensible Models in Draco. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)* (2019). http://idl.cs.washington.edu/papers/draco
[11] Aditeya Pandey, Arjun Srinivasan, and Vidya Setlur. 2023. MEDLEY: Intent-based Recommendations to Support Dashboard Composition. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (Jan. 2023), 1135–1145. https://doi.org/10.1109/tvcg.2022.3209421
[12] Sanad Saha, Nischal Aryal, Leilani Battle, and Arash Termehchy. 2023. How Does User Behavior Evolve During Exploratory Visual Analysis? *arXiv preprint arXiv:2312.09407* (2023).
[13] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2016. Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 341–350.
[14] Rajesh Shrestha, Omeed Habibelahian, Arash Termehchy, and Paolo Papotti. 2023. Exploratory Training: When Annonators Learn About Data. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–25.
[15] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2016. Voyager: Exploratory Analysis via Faceted Browsing of Visualization Recommendations. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)* (2016). http://idl.cs.washington.edu/papers/voyager