Diverse Interaction Recommendation for Public Users Exploring Multi-view Visualization using Deep Learning

Yixuan Li, Yusheng Qi,Yang Shi, Qing Chen, Nan Cao, Siming Chen*

Abstract—Interaction is an important channel to offer users insights in interactive visualization systems. However, which interaction to operate and which part of data to explore are hard questions for public users facing a multi-view visualization for the first time. Making these decisions largely relies on professional experience and analytic abilities, which is a huge challenge for non-professionals. To solve the problem, we propose a method aiming to provide diverse, insightful, and real-time interaction recommendations for novice users. Building on the Long-Short Term Memory Model (LSTM) structure, our model captures users' interactions and visual states and encodes them in numerical vectors to make further recommendations. Through an illustrative example of a visualization system about Chinese poets in the museum scenario, the model is proven to be workable in systems with multi-views and multiple interaction types. A further user study demonstrates the method's capability to help public users conduct more insightful and diverse interactive explorations and gain more accurate data insights.

Index Terms—Interaction Recommendation, Visualization for public education, Mixed-initiative Exploration

1 Introduction

Visualization systems have been recently introduced in public cultural centers, museums, and websites, accessible via touch screens, computers, and mobile phones for public education, data journalism, and data storytelling. In the past ten years, museums and other types of public education scenarios have made considerable efforts to create large and advanced visualization systems for audiences to browse and explore [32]. For example, the EMDialog visualization system [31], the Qichang Dong Artwork exhibition from Shanghai Museum [1], and the Living Liquid system in the Exploratorium [37] are popular multi-view visualization systems. They all show how interactive visualization systems can help public education by enhancing users' engagements in exhibitions and helping them learn through hands-on practice and free exploration. Visualization is a reliable tool to present socio-cultural, political, scientific, or other topics to public users for its ability to promote a deeper and broader understanding.

However, visualization systems constructed of multiple linked views and interactions types are often complicated. Public users without expertise in datasets, visualization systems and data analysis can hardly take full advantage of the system. As a result, public users might conduct limited exploration in a confined area or conduct random explorations wandering around the system. In either case, the exploration is unsatisfied and does not fully utilize the system's capabilities.

To solve this problem, various researchers focus on the interaction recommendation of visualization systems [23, 24, 41]. These works laid the foundation for personalized interaction recommendations and inspired our work. However, most of the current research is limited to single-view visualization systems or only consider public user's current step's interaction information to determine the following interaction recommendation, which is unable to support visualization systems with multiple views and interaction types.

This paper proposes a solution to the interaction recommendations for public-user-oriented interactive visualization systems with multi-views and multiple interaction types. Using deep learning methods, our model aims to recommend interactions for users' subsequent system exploration instantly. We model the interaction behaviors and visual states with the Long Short-Term Memory (LSTM) network. Interactions and visual states are modeled quantitatively and converted into numerical vectors as inputs to the deep learning model. The model evolves by capturing users' actual interactions to fine-tune the deep-learning model. To evaluate our method's performance, we create an interactive visualization system about Chinese poets' biographical data in a museum context. With the collected interaction and visual-states logs, we train the model for prediction and then conduct a user study for evaluation. Our results show that the model works effectively in visualization systems with multi-views and multiple interaction types. Further analysis shows the model's capability to help public users conduct more insightful interactions and facilitate diverse exploration styles with the help of data filters and multiple LSTM models.

Altogether, we made the following contributions.

- **Numerical Modeling of Interactions:** We propose a method that models interaction logs from different views and interaction types into vectors that share the same structure. Visual states in different views are also mapped into vectors with similar methods.
- **Interaction Recommendation Model in Visualization Systems:** To the best of our knowledge, we are the first to propose a mixed-initiative approach for across-view and across-interaction type interaction recommendations based on users' historical interaction logs and the corresponding visual state in visualization systems. This method can provide real-time, diverse, and insightful interaction recommendations, thus improving the system's usability.
- **An Application Scenario:** Our approach is proven to effectively solve the pains faced by public users-oriented interactive systems in public education scenarios such as museums. This model enables public users to gain more knowledge in these scenarios, thereby improving the accessibility of visualization and enhancing the education capabilities of public education institutions.

2 Related Work

Related work is discussed from the following aspects: interaction log analysis in visual analysis, provenance meta-analysis, and interaction recommendations for data exploration.

2.1 Interaction Log Analysis in Visual Analysis

Interaction occurs at all stages of the visualization process, transforming raw data into graphical insights [22]. As the most critical element in visual analysis and exploration, it determines users' overall experience [31]. The analysis of the interactions in visualization systems has become particularly important for conducting user studies, improving users' exploration experience, and evaluating visualizations [26]. For instance, In Brown et al.'s work [11], users' interactions with the game Waldo were used to understand users' personalities. Sukumar

- *Y. Li, Y. Qi and S. Chen are with School of Data Science, Fudan University. S. Chen is the corresponding author. E-mail: simingchen@fudan.edu.cn.*
- *Y. Li is now also with UC Santa Barbara .*
- *Y. Shi, Q. Chen and N. Cao are with Tongji University.*

Manuscript received 31 March 2022; revised 1 July 2022; accepted 8 August 2022. Date of publication 26 September 2022; date of current version 2 December 2022. This article has supplementary downloadable material available at https://doi.org/10.1109/TVCG.2022.3209461, provided by the authors.
Digital Object Identifier no. 10.1109/TVCG.2022.3209461

and Martinez investigated how to analyze users' interactions to help users better understand visualizations of their personal data [47]. Nobre et al. developed reVISit system for evaluating users' performance in complex interactive visualizations for a better design [40].

Interaction logs are often classified into two types: low-level logs, such as mouse movement information and eye movement data, and high-level logs, such as brushing and zooming [26,59]. The ways of analyzing interaction logs are often divided into qualitative analysis, such as analyzing the interaction patterns from visualizations of interaction logs [7,42,54] and quantitative analysis, such as converting interaction into numerical forms and using machine learning for data analysis [51]. Blascheck et al. quantitatively analyzed users' mouse movement and eye movement data to study how people interact with a visualization [8]. Guo et al. used interaction data to predict users' cognitive abilities during their visualization usage [19]. Guo et al. [28] used both qualitative and quantitative methods to analyze interaction logs in visual analysis systems for understanding how interactions guide insight generations. Each interaction action is defined as the tuple of the action name, the component this action applied to, and the timestamp of this action.

Inspired by these previous work, we propose a new method that captures users' high-level and semantic interaction logs and adopt a quantitative-qualitative combining method to map them into a unified parameter space for subsequent recommendations. Therefore, interaction logs across-views and across interaction types can be well defined.

2.2 Provenance Meta-Analysis

Provenance Meta-Analysis [49] is to present an ensemble of interaction provenance by converting interaction logs into high-dimensional vectors, to understand users' behaviors, evaluate visualization, and conduct model steering [57]. The work of ModelSpace presented a pipeline that converts the interaction into a high-dimensional vector and then visualizes the vector to show the interaction trails [12].Walchshofer et al. improved the method to transform users' interaction provenance into numerical vectors by capturing the application states after each interaction [49]. However, the structure and length of provenance vectors largely depend on the charts and data, making it hard to be adopted in complex visualization systems. Our inspiration is drawn from the previous methods to represent interaction data as an array. As an improvement, we bring up a brand-new way for encoding interaction provenance with inspirations drawn from declarative visualization grammars, which shows strong capabilities to represent interactions in visualization systems with multiple views and interaction types.

Satyanarayan et al. developed the Reactive Vega, which aimed to build declarative primitives to interaction. The input interaction events were parameterized and decomposed into JSON-based semantics [45]. Vega-Lite managed to decode interaction events into a parametric space with a higher-level structural language, making it more convenient and less time-consuming to define an interaction [44]. Each interaction log is interpreted by a specific set of parameters, ensuring a one-to-one correspondence between the interaction and the decomposition result.

Our method studied how Vega and Vega-Lite define signals and then constructed our pipeline to represent the interactions as JSON-structure data regarding Vega's declaring format. After that, we used a Context-Free Grammar (CFG) [20] to convert JSON-structure data into a parse tree and then vectors to facilitate quantitative analysis.

2.3 Interaction Recommendations for Data Exploration

Visual narrative or visual storytelling leverages visualization to tell stories for users [25, 34]. Researchers have investigated how to support storytelling from the interactive analysis process [17]. Recent work shows that interaction guidance is essential for helping users find functionality in visual interfaces and better explore the system [46].

Boy et al. defined suggested interactivity as methods that indicate an area can be interacted with by directing users' attention from the rest of the interface [10]. Ceneda et al. characterized interaction guidance along the knowledge gap of the user, the input and output of the guidance generation process, and the degree of guidance that is provided to users [15]. Zhou et al. built a model to represent the focus of a user's analysis through his past interaction behaviors [60]. Willett et al. created a software structure that added visual cues to the interface to suggest interactions [53]. In Falcon [39], Moritz et al. modeled a user's session with a client-side state and contributed prefetching techniques to eliminate latency for brushing interactions. Most of the work addressed mixed-initiative interaction approaches [3], i.e., both machines and users decide the best-suited interaction at each turn.

Generating interaction recommendations is another hot topic in this scope. Common modeling method including *Pattern Analysis* and *Probabilistic Models*. *Pattern Analysis* detects patterns in user preferences or interaction logs. Milo and Somech presented REACT that generates next-action suggestions by generalizing multiple abstract actions executed in similar n-contexts [38]. Dabek and Caban [21] construct a grammar-based model for generating interaction suggestions based on common patterns of user interaction logs. In VisGuide [13], Cao et al. modeled user's preferences using an online-learning method to recommend comparison and drill-down charts to help users do both hierarchical structuring and parallel structuring data exploration. The Markov model is a common probabilistic model in this topic. Cetintemel et al. [16] built a first-order Markov chain to predict users' interaction with the database. Battle's work [5] created an outgoing transition from each state for every interaction the user can make in the interface using an *n*-th order Markov chain. Wall et al. [50] defined a Markov model of "unbiased" interactive behavior in Scatterplots, in which each combination of data point and interaction type comprises a state. Ottley et al. [41] introduced a hidden Markov model for predicting user's attention by combining visual features in the visual graphs and history interaction behaviors. Deep learning models are also widely used in this field. Fan and Hauser [23] introduced a convolutional neural network to estimate users' selection in scatterplots by evaluating how data distributes in visualization. Li et al. [35] derived a model for predicting interaction actions in vertical menu selection based on the Long-Short Term Memory (LSTM) network. Wu et al. [55] presented MultiVision, a mixed-initiative system for recommending charts based on Siamese and LSTM network structure to help users create analytical dashboards. Embeddings of data columns in visual charts are fed into a bidirectional LSTM model for further charts recommendations. Similarly, we apply LSTM-based models to solve the sequence prediction question but instead of focusing on charts, we focus on interactions.

In our method, we numerically encode interaction logs and visual states across views and feed the embeddings to LSTM models for interaction recommendations with the mixed-initiative approach addressed. Different from the previous work, our method can present interaction recommendations in systems with multiple views and interaction types.

3 Overview

This work aims to provide interaction recommendations to public users during their exploration of visualizations with multiple linked views. To demonstrate the necessity and value, we first present a motivation scenario. Then we outline the design requirements and research problems, providing an overview of Sec. 4 and Sec. 5.

3.1 A Motivating Scenario

The interactive visualization system has become an essential part of the public education digitization process [31]. In this work, we aim at facilitating users' interactions in interactive visualization systems targeting public users. "Public users" are defined as museum-goers with different education background and ages, have little background about the topic and data analysis but are curious to explore for insights, and do not have a specific targeting goal to achieve when using the systems. We can assume that museum-goers' background is generally in line with the definition of our target user. They may have their own insights, but know little about how to make complex interactions. Users with professional backgrounds are not within our target users. Assumptions of our target users are supported by interviews and existing literature [9, 37]. We select the museum as a motivating scenario.

3.1.1 Interview with A Museum Domain Expert

To understanding the needs of our target scenario and generate reasonable design goals, we first talked with a prestigious expert from the museum informatics and public education field with over 20 years of experience constructing visualization systems for education in many museums. First, she recognized the rising importance of visualization systems in museums because these systems could trigger the audience's own enthusiasm for acquiring new knowledge through hands-on practice instead of teaching the audience a specific principle.

She also pointed out some current problems.The most critical one would be the low utilization. Many museum visitors either don't interact with the systems, or just randomly click and leave. The main reason is that audiences do not know how they can interact with the system without guidance. She believes there is an urgent need to solve current problems and help public users to explore visualization systems better.

3.1.2 Interviews with Public Users

We also conducted interviews with two students that qualified our target users' profiles. Since visualization systems are still not very common in museums, we asked about their experience in using interactive computer systems. Firstly, they showed willingness to play with these systems because of their attractiveness. Secondly, they would spend more time when the topics were very appealing and the systems were easy to use; otherwise, they just click around and leave. Thirdly, they mentioned the discomfort of learning to use a system on their own in museums. Also, they prefer to spend less than 10 minutes with one system.

3.1.3 Challenges in Current Practice

Researchers have also recognized the problems in the usage of interactive visualization systems in museums. Slingsby et al. found that novice users often failed to discover or used more advanced interaction functionality [46]. Hinrichs et al. illustrated that since visitor' limited visiting time and the systems' competitions with other exhibits under the scenario of informal learning environments, it is hard for visitors to fully explore these systems without guidance [31].

Through literature studies and interviews, we conclude the challenges in the current practice and construct a scenario for clearer illustration together with this domain expert. A public user is visiting an exhibition about Chinese poets. A digital screen is provided for the audience to explore a visualization system, displaying the travel trajectory and poetry of several Chinese poets. However, due to the following circumstances, her exploration does not go so smoothly.

Audience's Background Knowledge Lacking Facing a system with multiple views and interaction types, this user has to decide which interaction type to adopt and what data aspects to explore. However, this decision relies on the user's background knowledge and data analysis capabilities. As illustrated in [37] and [9], museum audiences usually arrive without this "pre-requisite" knowledge. Therefore, it is hard for the visitor to initiate their exploration in the system.

Multi Views and Interaction Types Due to the multiple views and the lack of knowledge, the audience may tend to concentrate on one single view and one interaction type, thus adopting limited interactions to explore in a single view and ignoring other views and data. This results in monotonous data exploration and underutilization of the interactive visualization system.

No Facilitation Most exhibitions are designed to be used without external facilitation [37]. Therefore, the audience would only refer to the systems' instructions or guidelines when they encounter obstacles, which are usually non-real-time, and entry-level. Such instructions may not provide continuous assistance and or consider users' actual interactions, making it hard to improve the user's interaction experience.

It can be concluded that users feel hard to construct interactions in visualization systems without proper guidance and instructions. As a result, facilitating users' exploration in the visualization systems has become a significant priority [31]. We primarily focus on designing a model to generate interaction recommendations for public users. We addressed mixed-initiative interaction approaches by that both the model and users decide the best-suited interaction at each round.

3.2 Design Goals

Sec. 3.1 illustrates the necessity to provide public users with subsequent interaction recommendations in interactive visualization systems with multiple views and interaction types. We then outline features that provided recommendations should require based on findings from our application scenario.

G1: Real-time The **No Facilitation** feature requires the system to provide users with continuous and **real-time** interaction recommendations to ensure users' using experience. When users face multi-view visualization systems, we believe the most straightforward solution is to recommend interactions right after one interaction is made. Our model should support providing the recommendations of the next interaction immediately after one interaction is made. This can give the users clearer interaction guidance and enhance users' interaction experience.

G2: Diverse The characteristic of **Multi Views and Interaction Types** of the systems reflects that the recommendations should support across-views and various types interactions. This can not only help users conduct more in-depth explorations that approach the data from diverse perspectives but also help the system be fully utilized.

G3: Insightful The users' **lack of pre-requisite knowledge** suggests that the system offers **insightful** recommendations that can lead to valuable data insight, to assist users in exploration. The insightful interaction recommendations can lead to valuable exploration that can deepen the user's understanding of the topic she explores.

3.3 Research Problems

Our design goal is to provide real-time and diverse interaction recommendations for public users during their exploration of the system and inspire them to obtain more accurate data insights. In order to propose interaction recommendations that support the above requirements, we analyze our task and break down the research problems as follows.

R1: How to encode the interaction behaviors? We need to bring up a mechanism that can encode interaction logs of multiple interaction types and object views. This is the basis of further study.

R2: How to construct a model for recommending interactions? This problem decides the structure of our predicting model and greatly influences our method's performance.

R3: How to quantitatively calculate the degree of diversity and insightfulness? We need to quantitatively define the degree of diversity and insightfulness to filter the collected training data and use these as parts of the evaluation metrics.

R4: How to present our real-time interaction recommendations to the users during their exploration? It decides our method's usage and whether our method can combine human and machine intelligence.

Corresponding solutions are carefully illustrated in the following Sec. 4 and Sec. 5.

4 Model Framework

Based on the research problems in Sec. 3.3, we present a model that can give users real-time interaction recommendations when exploring multi-view visualization systems. From literary research [41,52], we concluded that users choose their interactions based on the previous behaviors and the information displayed by the visualization chart. An **interaction log** is one record that contains what type the interaction is, what data it interact with and on which view it happens. Based on this conclusion, we decide to encode the interaction logs and visual states numerically as the input and use deep learning methods to make interaction recommendations based on these.

The workflow presented in Fig.1 is constructed with several parts: an interaction converter for modeling interactions, a visual states converter for modeling visual states, and a recommendation generation model for recommending next interactions. A user is shown a visualization system with multiple views and interaction types. Different interaction recommendations will be presented in the UI after the users' current interaction. The user can choose to follow the recommendations or not. The system captures users' actual interaction for further model training. Following subsections introduce how each step works.

4.1 The Interactions Converter (R1)

To define the encoding of interaction logs, we need to decide how to categorize interaction types and map the logs to numerical space. In this section, we focus on solving **R1**.

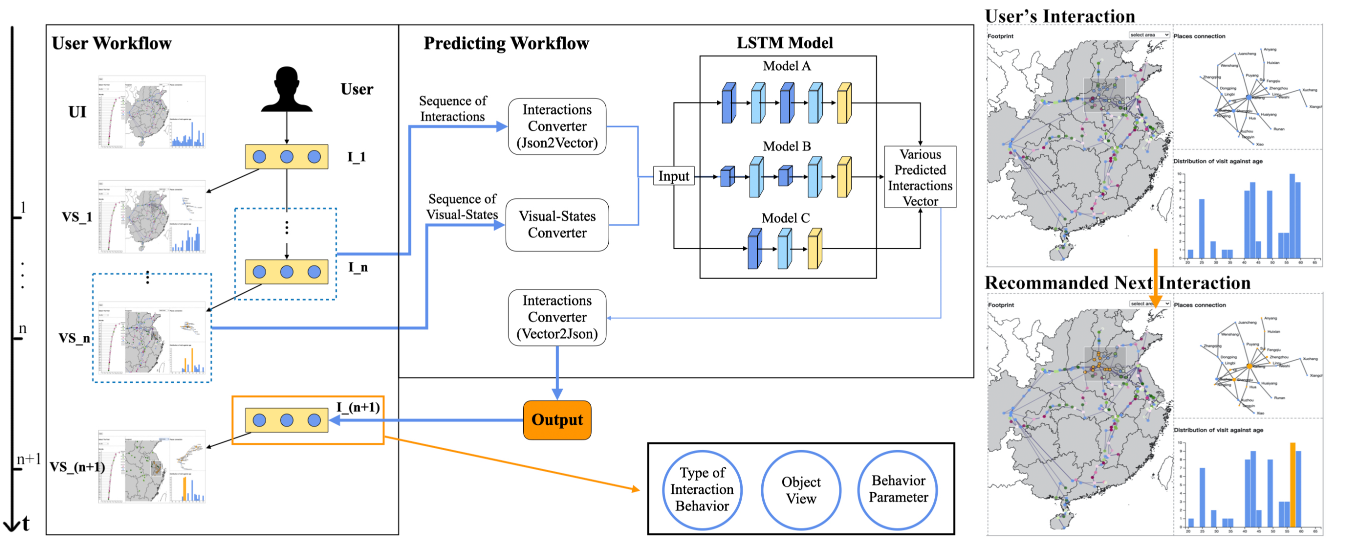


Fig. 1. A high-level illustration of our workflow for recommending interactions. In the user's workflow, each time the user completes an interaction, the system collects the interactions and visual states and then transfers them to the converters in a sequence. In the predicting workflow, the pre-trained LSTM models receive inputs and then output various predicted interaction vectors. After being converted to JSON data, the corresponding recommended interactions will be presented to the user. The two UIs on the right represent one user's interaction of brushing in the Map View, and the system gives the recommended interaction of selecting one column in the Histogram view for the next step.

First, we define a discrete-time index t to distinguish each **interaction log** I_t. At the start of a user's exploration, $t = 0$. Each time the user explores, t increases accordingly, thus forming an interaction sequence of user i $S_i = [I_0, I_1, \ldots, I_k, \ldots]$.

4.1.1 Interaction Type

There are many widely recognized interaction taxonomies for visualization systems [4, 18, 53]. To make our method more general and universal, we develop a taxonomy of **interaction logs** based on the interaction classification method in [58]. Three interaction types that are high-level and semantic are covered and listed in Table. 1.

Table 1. Interaction Types and Parameters

Type	Definition	Parameters
Select	Choosing interesting points, such as click, brush, etc.	The coordinate range and the index of the content (if any)
Filter	Querying data marks with conditions, such as filtering specific channels	The conditions and scope of its filtering
Abstract / Elaborate	Showing an area of information more or less detailed such as zoom and pan, etc.	The relationship between the area to be explored and full view

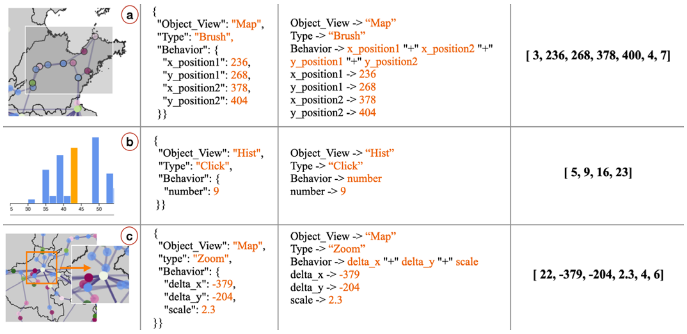
4.1.2 Interaction Parameters

In Reactive Vega [45], the authors defined specific parameter spaces for different types of interactions for interpretation. Inspired by this work, we determine a set of parameters $P_T = [p_1, p_2, \ldots, p_n]$ of each interaction type T in Sec. 4.1.1. Different interaction types have different sets of P_T, and one P_T has various values for each time interaction log. The last column in Table. 1 shows the details.

4.1.3 Modeling Interactions Numerically

We first convert the interaction logs into JSON-structure data, then to numerical vectors for further study. This section introduces this pipeline.

We proposed a rule that transforms each **interaction log** to a JSON structure of (*Interaction Type, Object View, Interaction Parameters*), which can define interaction data from across-view and across interaction types. Then we apply an embedding method to the JSON data using a Context-Free Grammar (CFG) representation [20]. This representation helps to transform data in different JSON structures to parsed trees based on rules extracted by CFG from the dataset. Under this method, data that has different JSON structures can be converted into the same rule-governed format since the method is conducted over the whole dataset. The parsed-tree consists of the following keys:



```
{
 "Object_View": "Map",
 "Type": "Brush",
 "Behavior": {
  "x_position1": 236,
  "y_position1": 268,
  "x_position2": 378,
  "y_position2": 404
 }}
```

```
Object_View -> "Map"
Type -> "Brush"
Behavior -> x_position1 "+" x_position2 "+"
y_position1 "+" y_position2
x_position1 -> 236
y_position1 -> 268
x_position2 -> 378
y_position2 -> 404
```

[3, 236, 268, 378, 400, 4, 7]

```
{
 "Object_View": "Hist",
 "Type": "Click",
 "Behavior": {
  "number": 9
 }}
```

```
Object_View -> "Hist"
Type -> "Click"
Behavior -> number
number -> 9
```

[5, 9, 16, 23]

```
{
 "Object_View": "Map",
 "type": "Zoom",
 "Behavior": {
  "delta_x": -379,
  "delta_y": -204,
  "scale": 2.3
 }}
```

```
Object_View -> "Map"
Type -> "Zoom"
Behavior -> delta_x "+" delta_y "+" scale
delta_x -> -379
delta_y -> -204
scale -> 2.3
```

[22, -379, -204, 2.3, 4, 6]

Fig. 2. Examples the encoding process of (a) Brush, (b) Click, and (c) Zoom. These interactions are examples of interactions in Table. 1. They are converted to JSON representations, then to CFG Parsed Tree, and then to numerical vectors based on the rules generated by CFG Tree.

- **Object View**: Indicating the object view of the interaction log.
- **Type**: Indicating the type of the interaction log.
- **Behavior**: Indicating the parameter set corresponding to the interaction type that can uniquely define an **interaction log**.
- **Behavior Parameter Value**: Indicating the corresponding value of the parameters in **Behavior**.

We use a new mechanism to express both quantitative and qualitative variables in the same vector. For qualitative variables **Type**, **Object View**, **Behavior**, the attribute is represented by the index of the corresponding value; for a quantitative variable **Behavior Parameter View**, its value is directly filled in the corresponding position of the vector. Through this method, we can map each interaction log to a unique numeric vector. Taking the Brush operation in Fig.2 (a) as an example, the user performs a brush operation in Map view, starting at point (236, 268) and ending at point (378,400). In this case, the first attribute value '3' indicates that the 2nd-5th attributes are the Behavior Parameter Values. The 6th attribute value '4' represents the view 'Map' and the 7th attribute value '7' indicates that the interaction type is 'Brush'.

4.2 The Visual States Converter (Continuing R1)

Every interaction in the visualization system, in many cases, will cause the visual states in each view to change, thereby bringing new data insights to the users. These visual states and data insights may affect the user's next interaction.

We consider what information can represent the status of the current visualization, and decompose the data distribution information on a

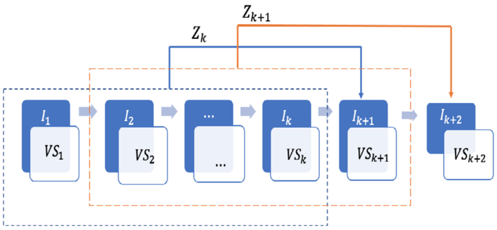


Fig. 3. The Data Structure of LSTM Model. The first k {Interaction, Visual States} are sequenced together to form an input value Z_k, and the corresponding label value is the next interaction I_{k+1}

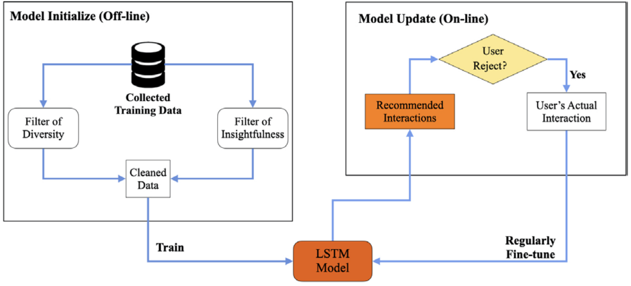


Fig. 4. Model Initialization and Model Updating. The training data is first cleaned by two filters and then fed into the LSTM model for training. This process is model initialization and is conducted offline (before the recommendation process). Model Updating is conducted during the recommendation process by capturing users' actual interactions and then using this data for regularly fine-tuning the pre-trained LSTM model.

view into low-level visual channels to form a quantitative information collection $VS = (C_1, C_2, ...C_k)$. [6] defined the visual channels of marks on a visual chart as the color, position, size, shape, etc., which is able to describe any visual charts [14]. We believe this information set VS affects the user's choice of the next **interaction log**. Thus we generate the parameter space of encoding visual states based on the law in [6]. For instance, a Map View with Point marks distribute on it, the recorded visual state is stored in JSON format with keys (channels) {Points' coordinates, Points' color, Points' size}. We then use the tSNE [48] method on each channel of each view for dimension reduction. Each view is represented by a numerical vector and the visual states in the systems are represented numerically. The numerical vector is attached to the corresponding interaction for model training.

4.3 An LSTM Model Approach (R2)

In this section, we focus on solving **R2**. We compared the LSTM model, Hidden Markov Model (HMM) and Transformer-based network, which are popular in sequential data analysis. HMM only considers the impact of the timestamp $t-1$ at t and does not consider the historical interaction sequence. Also, the transition matrix of visual states is emphasized in the HMM, but this is not what we need to consider, thus increasing the model's complexity. The transformer-based network lacks modeling of the temporal dimension in the sequences, which is important information in our task. The complexity of the Transformer-based model also is unwanted in our scenario. LSTM shows significant advantages among them because it considers the user's historical interaction data, imitate the user's memorizing and forgetting habit and is more convenient and straightforward for experts in museums and other scenarios to train.

Taking all these into account, we choose the LSTM model. Fig.3 shows how the model works. We use the time-shifted window approach to process the data and set the window's length as b, i.e., the last b records are used to predict the next interaction log. Multiple models with various parameters are provided to improve the diversity of our method's results (**G2**). The LSTM model is fed with the input sequence $\{Z_j; j = n-b+1, n-b+2, \cdots, n\}$ and then outputs a vector of the recommendations for each step. After decoding this vector, we could obtain recommended interactions, which are later presented to the user.

4.4 Model Initializing and Continual Learning (R3)

Instead of using a static model, we divide our model training into two parts, as shown in Fig.4.

4.4.1 Model Initializing

The training data is first cleaned by two filters of diversity and insightfulness to ensure the data quality required by **G2** and **G3**. Therefore, the recommending results can also have this nature because of that the Kullback-Leibler divergence [33] between the distributions of the recommended results and training data should be small. We compute the entropy of users' interaction types and views to measure the degree of diversity and keeps the interaction logs only if the entropy is higher than a threshold. The insightfulness filter regards a series of interactions as insightful if it involves interactions within the insightful data areas. The insightful areas are selected on different views where data are distributed in clusters. For instance, on Map View the insightful areas cover the east, south, and central parts of the map. Based on the above illustration, we employed the following two equations to calculate the $Degree-of-Diversity$ (DoD, ranging from 0 to 1; a higher DoD is closer to 1), $Degree-of-Insightfulness$ (DoI, ranging from 0 to 1; a higher DoI is closer to 1). Here we focus on solving **R3**. The two filters are controlled by a pair of weight-tuning parameters. A higher weight refers to stronger filtering. These weight-tuning parameters are pre-setted by system managers based on the preferred needs for diversity and insight. For instance, a larger DoI is suggested to obtain a preference for insightfulness if the museums would like visitors to have more task-oriented exploration. And a larger DoD and a smaller DoI can be set to obtain a preference for diversity if more fun is wanted.

After the filtering, the cleaned data is fed into the LSTM model above and the training will start.

$$DoD = Entropy\ of\ interactions'\ distribution\ of\ each\ Interaction\ Type \tag{1}$$

$$DoI = \frac{\#Predictions\ that\ within\ the\ insightful\ data\ area}{\#Predicting\ results} \tag{2}$$

4.4.2 Model Updating

To keep high-quality performance over time, our model achieves continual learning by supplementing a steady stream of new training data.

Training data explicitly calls out attributes in the dataset that represent fundamental facts in the real world, which is continually changing. [29]. Therefore, a steady stream of new training data is necessary for the model's performance. We recognize this issue and develop a continual learning pipeline regarding this.

As Fig.4 shows, once a user rejects the provided recommendations, we consistently collect the actual interactions and use them to fine-tune the pre-trained model regularly. For example, a user rejected the recommendations at time t and made his own interaction log I_t, then we would feed $[(I_{t-b+1}, ..., I_t]$ as a new piece of data to the model updating (the width of the LSTM's time-shifted window is b). We design this strategy because when the user rejects the recommendations, it means the recommendations are not satisfying and the user's actual interactions can help correct the models' predicting results. In this way, we realize the model continues learning without manual labor.

This strategy can also work as a cold start. When transformed to a new visualization system, our model can first generate random recommendations and collect the data once users reject the recommendations. With the gradually incoming data, the model gets fine-tuned several times and will gradually improve its performance.

4.5 Presenting the Predicted Results (R4)

This section focuses on solving **R4**. The predicted vector results generated by the LSTM model first need to be transformed into the hierarchical interaction log through the Interaction Converter using similar rules as described in Sec. 4.1.

The visualization system captures and analyzes the prediction results, and feeds back recommendations on the view. The recommended interaction position or area will be highlighted on the predicted Object View. For example, if the predicted result is brushing, the recommended brushing area will be highlighted in the corresponding object view. This presenting method is selected because it is straightforward for the audience to understand the recommendation when it is shown on the interface around the data marks that are recommended to interact with.

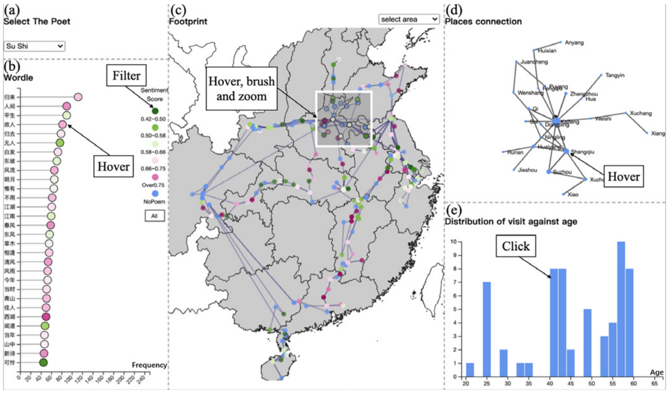


Fig. 5. The Interface of the System. (a) Select one poet to explore. (b) Wordle View. (c) Map View. (d) Force Directed Graph View. (e) Histogram View.

5 An Illustrative Example

This section proposes an interactive visualization system in a museum scenario as an example to implement and test our method completely.

5.1 A Multi-view Visualization System

We selected four different poets in Song Dynasty [2]. The data contains the Spatio-temporal trajectory data of when and where a literati was created and the content. To present the poets' life from different angles, and to meet our method needs (i.e., multi-view, multi-interaction types), we designed the following visualization system (Fig.5).

For the poetry data, we use the TF-IDF algorithm [43] to calculate the keywords in each poet's works and apply sentiment analysis [36] to each work's. For each keyword, we calculate the average of all poems' sentiment value that this keyword appears in; for each location, we calculate all the poems' average sentiment value in that location. We draw the trajectories in chronological order on the map. The points' color is based on its sentiment value. The force-directed map shows the connection between various locations. A location with a larger radius means the author visited this location more often.

This system consists of five views including (a) Poets button for selecting poets to explore. (b) Wordle View that presents keywords with their sentiment value and frequency. It supports interaction types of Hover and Filter. The x-axis represents the frequency of each word. (c) Map View that shows the places a poet visited. It supports Hover, Brush, and Zoom. (d) Force Directed Graph View showing the connection between the locations the poet visited in the selected area. It supports Click. (e) Histogram View that shows the frequency of poets' visits in the selected area at different ages. It supports Click. The x-axis represents the age. Views are connected through interactions like select, hover, zoom, etc. We also compile a set of data insights from the dataset with the guidance of a domain expert in Sec. 3.1.1. as a validation for our model's performance of insightfulness.

5.2 Modeling Users' Interaction and Visual States

Table 2. Interaction Types

Taxonomy	Type	Object View	Behavior
Select	Hover	Wordle	Coordinates of points
	Hover	Map	Coordinates of points
	Hover	Force-Directed	Coordinates of points
	Click	Histogram	Index of columns
	Brush	Map	Coordinates of bounding boxes
Abstract	Zoom	Map	Scale, Δ_x and Δ_y
Filter	Filter	Map	Value

We have incorporated seven different interaction logs types based on different views into the system. According to the method in Sec.4.1, we define each **Behavior** of each type of interaction, and then map the interaction data into a numeric vector. The interaction types and parameter sets P_T are listed in Table. 2. The numerical vector's structure after mapping is [**Behavior, Parameter values, Object view, Type**].

We also obtain the corresponding visual states generated during each interaction. The channels we collected for each interaction view are listed in Table. 3. The collected visual states are then transformed into low-dimension vectors using the method illustrated in Sec. 4.2.

Table 3. The Visual States of Each View Type

View Type	Mark Type	Visual Channel
Wordle	Points	Points' coordinates and color
Map	Points	Points' coordinates and color
Force-Directed	Points	Points' coordinates and radios
Histogram	Columns	Columns' index and height

5.3 Preparing Predicting Models

Our method uses deep learning to predict the user's next interaction. Firstly, we need to collect interaction data generated under purposeful and logical data exploration conducted by public users after understanding our system. Then, through the data processing method mentioned above, the data is converted into a numerical format. We construct the training dataset based on these generated numerical vectors.

5.3.1 Training Data

47 participants that satisfied the definition of "public users" as defined in Sec. 3.1 are invited for data collecting. There are 21 women and 26 men with ages from 18 to 27. All the participants have finished or are in the process of college education. After 5 minutes of reading about the manual of system, they spent 9 minutes exploring the system in Google Chrome browser on their own computers. During the exploration, we collected each user's interaction data and the visualization states.

The weighted combination of a diversity filter and an insightfulness filter is applied to the training data to ensure the quality. The weight of two filters can be modified according to actual needs as a trade-off of the need for diversity and insightfulness. In this example, we set the two weights both to 0.5, meaning that the diversity and insightfulness are equally important. After mapping the cleaned data using the method in Sec.5.2, we obtained 4,716 interaction log vectors.

5.3.2 Model Training

The sequence composed of the current interaction and the previous b-1 interaction logs is the input (b is the width of the time-shifted window as defined in Sec. 4.3). When a user performs an interaction, the sequence will be passed into these models simultaneously and generate results. Based on experience, we took the window width from 3 to 5 and used different network structures to obtain 3 models.

We used several techniques to improve the model.Firstly, we normalized each variable before training. Secondly, since both Type and Object View are qualitative variables, after the predicted result was inversely normalized, we took the values from the value range of Type and Object View that were closest to the actual predicted results as the predicted result. The prediction vector generated by the above method was transformed into structured interaction data.

We used Bidirectional LSTM (BiLSTM) to build the model [27]. This structure traverses the input sequence in two directions to make predictions based on past information and uses future information to modify the prediction results, improving the effectiveness of the prediction compared with LSTM. Different choices of hyper-parameter values, including the sliding window's width, the number of BiLSTM layers in the neural network, and the number of cells in each BiLSTM layer, are tuned and compared for better results.

5.3.3 Performance Results

The collected data set was divided into the training and testing set at a ratio of 4:1. We compared the training duration, the accuracy and loss value of the training and testing set, and the mean square error of each model. The three models with the best prediction effects were selected as the final models. The results are shown in Table.4. After the comparison and selection, we retrained the models using the entire dataset and obtained three prediction models for further usage.

Table 4. Model Performance

Model	Structure	Loss	Valid-Loss	Acc	Valid-Acc	RMSE	Valid-RMSE	Trianing Duration
A	BiLSTM(100)+Dropout(0.2)	0.0142	0.0151	0.6915	0.6702	0.1188	0.1229	15s/epoch
B	BiLSTM(50)+Dropout(0.2)+BiLSTM(50)+Dropout(0.2)	0.0118	0.0128	0.7132	0.7001	0.1036	0.1133	26s/epoch
C	BiLSTM(100)+Dropout(0.2)+BiLSTM(100)+Dropout(0.2)	0.0090	0.0127	0.7406	0.7155	0.0952	0.1122	25s/epoch

Table 5. Impacts on DoD, DoI and accuracy of filters and model updating

Model Types	DoD	DOI	Accuracy
Filters(NO), Update (NO)	0.329	0.659	0.6814
Filters(Yes), Update (NO)	0.776	0.873	0.6656
Filters(NO), Update (Yes)	0.663	0.947	**0.7242**
Filters(Yes), Update (Yes)	**0.838**	**0.963**	0.7151

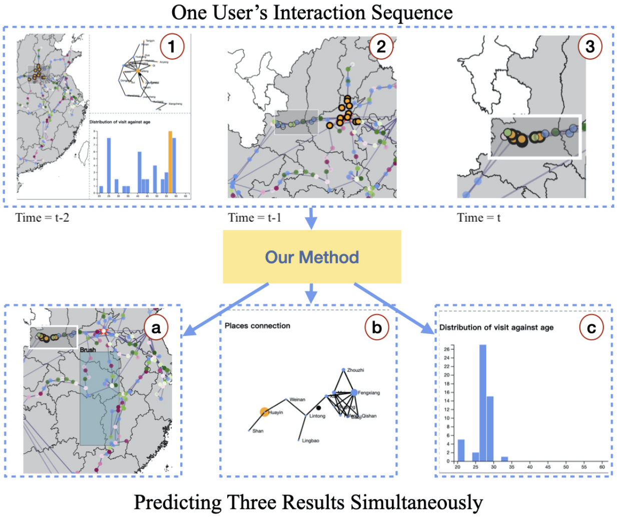


Fig. 6. One example of recommendation results of one input sequence of interaction logs. ①click a column on the Histogram View②brush an area on the Map View③zoom into an area on the Map View, compose a sequence of user interactions and the corresponding visual states. ⓐbrush on the Map View ⓑclick on the Force-Directed Viewⓒclick on the Histogram View are the recommendations given by our model.

5.3.4 Impact of the Weighted Filters and Model Updating

To evaluate the weighted filters and the model updating, we conducted an ablation experiment to compare the models' performance. We trained four different sets of models (with or without filters and with or without model updating), and used these to make predictions of the dataset we collected in Sec. 5.3.1. We then calculated the *DoD* and *DoI* using the equations in Sec. 4.4.1. The results are listed in Table. 5. It shows that the filters of diversity and insightfulness significantly increased the results' *DoD* and *DoI*, but decreased the accuracy very slightly, which measures the difference between the users' actual interactions and the recommendations, because the filters decreased the size of the training dataset by filtering out unsatisfying data. The results justify the effectiveness of the two filters. The model updating method also increased the results' accuracy.

5.4 Visual Exploration With Recommendations

After data collection and model building, we obtain three models for interaction recommendations. This allows our method to provide users with three different recommendations for the next interaction choice. Fig.6 shows an illustrative example of the three different recommendation results according to one input interaction log.

For each round, the visualization system makes and presents predictions and feeds back recommendations for the next operation. The user could choose to follow one of the recommendations or freely operate. Both previous users and the current users' interaction logs impact the provided interactions in two ways. Firstly, the pre-trained model generates recommendations based on this user's previous two interaction logs. Secondly, the model updating strategy helps the pre-trained model fine-tune itself based on the new cases for a better predicting result. It reflects the concept of human-machine coordination in our method.

6 Evaluation

In order to evaluate the effectiveness of our method, we further conducted a controlled user study and a set of in-depth interviews.

6.1 A controlled study

A controlled study was conducted for the investment of evaluating our method's performance.

6.1.1 Method

Participants were invited to use our visualization systems and complete a questionnaire at the end of the evaluation.

Participants 42 Participants were selected based on the target users definition in Sec. 3.2 with backgrounds ranging from computer science, data science, mathematics to museum study, psychology, and philosophy, and with ages ranging from 18 to 30 (M = 22.45, SD = 3.00). 31 have little experience in data analysis and 11 have basic data analysis knowledge less than 1 year. They all reported visiting 3 or more times in the last six months and having interests in Chinese poets. All the participants in this study are different from those in Sec. 5.3.1.

Participants were randomly assigned to three groups, of which 14 people without the interaction recommendation function as a control group (**Group A**), 14 with interaction recommendations randomly generated as another control group (**Group B**), and the remaining 14 people explored the visualization system with interaction recommendations generated by our model as the experimental group (**Group C**).

Procedure and Tasks

We first gave participants some digital exhibits about the poets to imitate the background knowledge provided by exhibits in museums. Then, we briefly introduced the visualization system. After that, participants explored our system on their own within 8 minutes. After that, a questionnaire about the poets was sent to participants to test data insights they gained from the exploration. The questions were selected from the data insights we constructed as mentioned in Sec. 5.1, including six correct insights (Creation and Trajectory Characteristic Questions 1-3), and two fake insights (Questions 4). Participants were asked to rate out of 5 their degree of agreement of the statements. For Group B and C, the questionnaires also contained some 5-point Likert Scales questions about users' experience toward the recommendations. A questionnaire sample and detailed procedure record are attached to the supplementary material.

6.1.2 Results and Analysis

Effect on Interaction Diversity We obtained 1368 interaction logs from Group A, 1660 from Group B, and 1480 from Group C. Then, we calculated the entropies of different interaction types (4 in total), interaction object views (4 in total), and interaction behaviors (7 in total) of each participant's interaction logs. A bar chart is presented in Fig.7 (c) to illustrate the average entropy of each criteria using the following formula $\bar{H}_{GroupX} = \frac{\sum_i (H_i)}{Size\ of\ GroupX}$. As shown in the Fig.7(c), the values of user interactions in group C are significantly greater than those in group A in terms of Interaction Type, and significantly greater than those in groups A and B in terms of Object View and Interaction Behavior. It illustrates our approach is able to provide recommendations with different interaction types and object views, thus increasing the diversity of user interactions, which satisfies our G2.

Also, we captured the coordinates of the interaction logs on the interface and drew a heat-map to compare their distributions. As Fig.8 displays, users' interaction range in Group C is broader, especially on

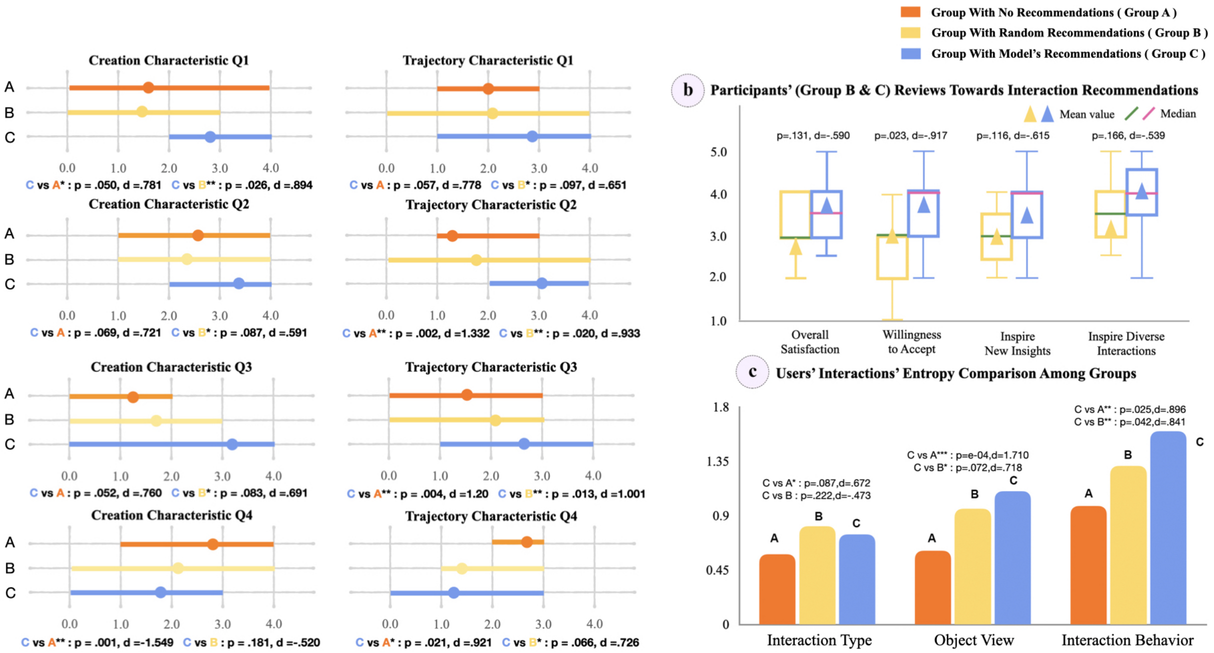


Fig. 7. Data Insights. Questionnaire Rating (a) shows the distribution of the acceptance degree of insight statements, with the results of paired t-tests' p-value between our model and two control groups (*, ** and *** denote $p < .05$, $.01$ and $.001$, respectively). The higher degree of acceptance of the correct statement (Creation and Trajectory Characteristic Q1-Q3), the better the data insight obtained by the user. 5-Points Likert Scales (b) shows users' reviews of random recommendations and our model's recommendations. Bar Chart (c) shows user's interactions' entropy in the aspect of interaction types, object views, and interaction behaviors among three different groups, along with the results of t-tests.

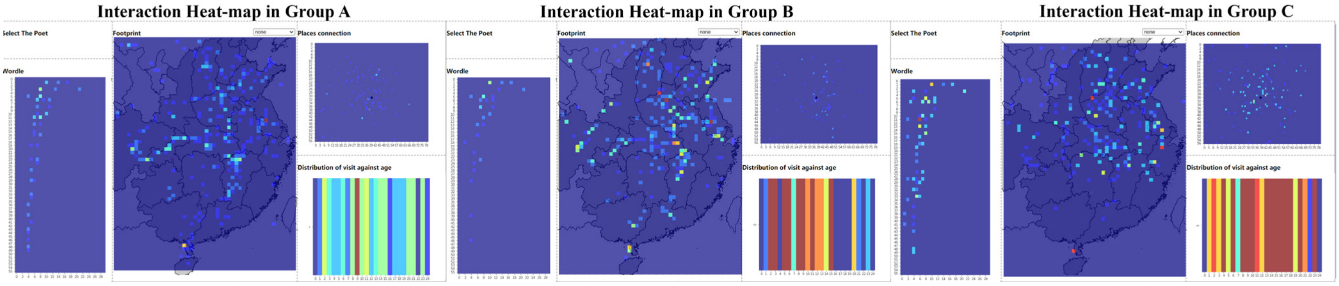

Interaction Heat-map in Group A
Interaction Heat-map in Group B
Interaction Heat-map in Group C
Select The Poet
Footprint
Places connection
Wordle
Distribution of visit against age

Fig. 8. Interaction Diversity Comparison. Interaction Heat-map shows the distribution of interactions on the interface in three groups. It is notable that the distribution of the interactions is broader in Group With Our Model's Recommendation, especially on Force-directed View and Wordle View.

the Force-directed View and Histogram View, compared with Group A and B. The distribution in Group B is broader than that in Group C on the Map View. This could be caused by the insightfulness filter, which may reduce the diversity to some degree reflecting the trade-off between diversity and insightfulness.

Effect on Obtaining Data Insights To evaluate whether our method could guide participants to obtain more data insights, we analyzed the questionnaire result. A higher acceptance of the correct perceptions and a lower acceptance of the incorrect ones means more accurate insights gained from the system. Fig. 7 (a) shows the distribution of the agreement of insight statements. Participants in Group C showed more agreement to good insights (Questions 1-3) and less to pseudo-insights (Questions 4) than those in Group A and B. The results have been verified by two paired t-tests between Group A & C and Group B & C. This shows our model's ability to help participants obtain a more insightful understanding of the target topic, satisfying our **G3**.

Usability The 5-point Likert Scales rating focuses on the acceptance rate of the interactive system's recommendations (**Reasonablity**), whether the system can guide participants to obtain more data insights (**G3**), whether the system can inspire users to adopt various interactions (**G2**) and the satisfaction of the system's usage. The averaged results are shown in Fig.7 (b). The acceptance rate is 52.58% in Group B and 72% in Group C, which shows that the model's recommendations are reasonable. Also, participants in both groups reported that recommendations could inspire them to adopt various interactions types. Group C users with score= 3.92 showed more agreement of this statement compared to Group B (score = 3.43). Group C also reported a score of 3.8 to our systems' performance, much higher than that in Group B. When recommendations were turned down, some thought the recommendations could be distracting when they wanted to explore using similar actions around a specific area. Some would reject the recommendations when they could not figure out what outcomes they could get after interactions. Possible solutions are discussed in Sec. 7.

Run Time Based on the deep learning framework, our model can generate recommendation results immediately. The time interval from the user's current interaction to the display of recommendations is 0.8s, which is close to no waiting time, ensuring the **G1**.

6.2 In-depth Interview

In this session, in-depth interviews with several participants were conducted for further opinions on our method.

6.2.1 Method

We invited three participants and conducted interviews about their attitudes towards the exploring experience.

Participants We invited a 26-year-old male student in the field of data science, a 21-year-old female student in the field of museum study, and the domain expert in Sec. 3 to evaluate our system.

Procedure and Tasks The study is divided into an observation part and an interview part. For the first part, participants follow procedures in Study 1 with one researcher around. After the experiment, a semi-structured interview was conducted for additional feedback.

6.2.2 Results and Analysis

Participants reported spending about 90s to get familiar with the systems and 8.5 minutes for an in-depth exploration. According to their claims, their exploration interest changed during the usage of the system, from the poet's activities in the capital to the poet's trajectory and the relationship between his creative style and the places. Participants hoped to obtain multiple insights on these topics through exploration.

Reasonability The participants' acceptance of interaction recommendations accounted for 70%, 55% and 59%, that is, on average, users have the possibility of 61% to accept the provided recommendations. One participant reported that although there were some repetitive recommendations, more than 65% of the system's interaction recommendations were reasonable, meaningful, and coherent with previous interaction behaviors. One participant reported that, since she found many of the recommendations were reasonable, she sometimes struggled to choose one from the three recommendations provided at the same time. She suggested the recommendations be provided in priority.

Diversity The interaction recommendations were very diverse (in more than 190 interactions, 54 interaction recommendations are on three different views, 68 recommendations are on two different views), and not intermittent (78% of recommendations include views based on the current interaction). The expert also agreed that the recommendations are helpful to explore more detailed information from a broader aspect. But it was sometimes difficult for her to immediately understand why a recommendation was made when the previous and current recommendations were very different. She suggested our method provide more explanation. Another participant experienced a similar issue, but he chose to trust the recommendations and would pick the recommendations to explore another area. In his case, he showed a relatively higher degree of trust with the system compared with his judgment, which is a limitation that also discussed in other previous work [30] [56].

Insightfulness The participants recognized that the recommendations could lead users to more data insights. One participant gave an example. When he selected the poet's travel experience in the eastern coastal area and wished to explore further, the system recommended he explore the Wordle View and filter different map point colors. This made him realize that he could pay more attention to the emotional characteristics of works in the region, which led him to draw new data conclusions. The domain expert also recognized this point but further suggested that some simple explanations of the visualization results could be provided after one interaction was conducted.

Usability All participants liked the current way how the system interacted with the users and how the recommendations were provided. One participant pointed out that the recommendations provided by our method were easy to follow and deepened his understanding of how to interact with visualization systems. The domain expert pointed out that our method is able to give users the freedom to explore what they like without museums' need to design a curation to teach the knowledge. However, one participant suggested involving the "UNDO" feature in the system since it reflected the users' second thoughts. This suggestion might be further developed in future work.

7 Limitations and Discussion

The illustrative example shows that our system can model interaction behaviors and visual states and make recommendations to facilitate and inspire users' next step interaction. The evaluation results suggest that our method can help public users explore complex visualization systems by combining human and machine intelligence together. However, we admit that this method does not come without any limitations.

Firstly, explanations about why the recommendations were generated and how to interpret the visualization can be provided when giving recommendations. We have tackled the functionality side of the provided recommendations (reasonability, diversity, and insightfulness). The next step would be focusing on interpretability and usability to further help the results be more understandable and acceptable.

Secondly, although we have designed strategies to mimic the environment of museums by presenting virtual exhibits to participants in the in-lab evaluation, a user study under a real scenario would be more persuasive. Also, participants should cover students from middle and primary school, and adults with lower degrees to better fit the distribution of museum visitors. And in the museum, the audience interact with most visualization systems with a touch screen, but in our experiment, they used a mouse to make interactions. A more rigorous evaluation will be conducted within the real museum environment for future work.

Thirdly, there is still room to improve the model even though our model is able to provide diverse and insightful interaction recommendations now. One way would be to collect more training data. Another approach would be to integrate different methods of parsing visual charts and extracting data insights, such as auto insights. In future work, we will try to experiment with other methods of analyzing visual states and compare their effects on the model's performance.

We admitted that showing one case in our paper limits the generalization and extensibility of our method. In this work, we emphasized more on giving a clear explanation and evaluation of our designed workflow. To reduce this limitation, we have thoroughly discussed the implementation and evaluation of this case. Moreover, the modeling updating strategy mentioned in Sec. 4.4.2 as a cold start and the encoding strategy reduce the cost of transforming the method to a new system. In the future, we would like to compact our method into a toolkit that can be added to the existing visualization systems to improve the reusability of our methods in more systems under the public education scenarios.

The trade-off between diversity and insightfulness deserves further discussion. Since every user might exhibit homogeneous exploration behaviors, different users show diverse exploration patterns that our method can learn. "There are a thousand Hamlets in a thousand people's eyes", "insightful" focused on helping audiences obtain a "Hamlet" after exploration, while "diverse" focused on "how each individual audience obtain a distinct Hamlet in their way". Therefore, the relationship between diversity and insightfulness is actually the unity of opposites. Better ways to deal with this relationship can be added, such as dynamically modification of filters' weights according to users' exploration tasks and interaction patterns. When users' tasks are clear, the insightfulness filter has a higher weight and vice versa. In future work, a module for recognizing users' tasks might be added, and the recognized users' tasks would be combined with interactions log, visual states and the weighted filters for generating predictions.

Recommendations provided to professions should be different. Domain experts might have clear tasks to achieve during their usage, so the recommendations should be more task-oriented. Also, since exports have more background knowledge, providing interaction hints or adding triggers to start recommending might be more helpful. New strategies are required to provide the recommendations to expert users.

8 Conclusion

This work introduces a method for making real-time interaction recommendations to users when exploring interactive visualization systems with multiple views and interaction types. This method adopts a deep learning structure to learn the patterns in historical users' interaction paths and corresponding visual states and make subsequent interaction recommendations to the current user based on the learned patterns. The results of an illustrative example and the user study found that the method is capable of giving real-time recommendations to public users when they are exploring interactive visualization systems. The recommendations generated by our method are proven to be reasonable, diverse, and insightful, which is consistent with our design motivation and effectively solves the tasks faced by novice users-oriented interactive systems in public education scenarios. We believe our method can contribute to interaction studies and visualization accessibility, helping visualization exploration better support public users.

Acknowledgments

This work is supported by Shanghai Municipal Science and Technology Major Project (No. 2018SHZDZX01, 2021SHZDZX0103), General Program (No. 21ZR1403300), Sailing Program (No.21YF1402900) and ZJLab. This work is also supported by NSFC No. 61972278 and No.62061136003.

REFERENCES

[1] Dong qichang artworks exhibition. https://www.shanghaimuseum.net/museum/dongqichang/index.html. Accessed: 2022-06-30.

[2] Souyun map of poems in tang and song dynasty. https://sou-yun.cn/poetlifemap.aspx. Accessed: 2021-02-08.

[3] J. Allen, C. Guinn, and E. Horvtz. Mixed-initiative interaction. *IEEE Intelligent Systems and their Applications*, 14(5):14–23, 1999. doi: 10.1109/5254.796083

[4] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, p. 15. IEEE Computer Society, USA, 2005. doi: 10.1109/INFOVIS.2005.24

[5] L. Battle, R. Chang, and M. Stonebraker. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data*, pp. 1363–1375, 2016.

[6] J. Bertin. *Semiology of graphics*. University of Wisconsin press, 1983.

[7] T. Blascheck, M. John, K. Kurzhals, S. Koch, and T. Ertl. Va2: A visual analytics approach for evaluating visual analytics applications. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):61–70, 2016. doi: 10.1109/TVCG.2015.2467871

[8] T. Blascheck, L. M. Vermeulen, J. Vermeulen, C. Perin, W. Willett, T. Ertl, and S. Carpendale. Exploration strategies for discovery of interactivity in visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 25(2):1407–1420, 2019. doi: 10.1109/TVCG.2018.2802520

[9] K. Börner, A. Maltese, R. N. Balliet, and J. Heimlich. Investigating aspects of data visualization literacy using 20 information visualizations and 273 science museum visitors. *Information Visualization*, 15(3):198–213, 2016.

[10] J. Boy, L. Eveillard, F. Detienne, and J. Fekete. Suggested interactivity: Seeking perceived affordances for information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):639–648, 2016. doi: 10.1109/TVCG.2015.2467201

[11] E. T. Brown, A. Ottley, H. Zhao, Q. Lin, R. Souvenir, A. Endert, and R. Chang. Finding waldo: Learning about users from their interactions. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1663–1672, 2014. doi: 10.1109/TVCG.2014.2346575

[12] E. T. Brown, S. Yarlagadda, K. A. Cook, R. Chang, and A. Endert. Modelspace: Visualizing the trails of data models in visual analytics systems. In *Proceedings of the Machine Learning from User Interaction for Visualization and Analytics Workshop at IEEE VIS*, 2018.

[13] Y.-R. Cao, X.-H. Li, J.-Y. Pan, and W.-C. Lin. Visguide: User-oriented recommendations for data event extraction. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. Association for Computing Machinery, New York, NY, USA, 2022. doi: 10.1145/3491102.3517648

[14] S. Card, J. Mackinlay, and B. Shneiderman. Information visualization. *Human-computer interaction: Design issues, solutions, and applications*, 181, 2009.

[15] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit, and C. Tominski. Characterizing guidance in visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):111–120, 2017. doi: 10.1109/TVCG.2016.2598468

[16] U. Cetintemel, M. Cherniack, J. DeBrabant, Y. Diao, K. Dimitriadou, A. Kalinin, O. Papaemmanouil, and S. B. Zdonik. Query steering for interactive data exploration. In *CIDR*, 2013.

[17] S. Chen, J. Li, G. Andrienko, N. Andrienko, Y. Wang, P. H. Nguyen, and C. Turkay. Supporting story synthesis: Bridging the gap between visual analytics and storytelling. *IEEE Transactions on Visualization and Computer Graphics*, 26(7):2499–2516, 2020. doi: 10.1109/TVCG.2018.2889054

[18] M. C. Chuah and S. F. Roth. On the semantics of interactive visualizations. In *Proceedings IEEE Symposium on Information Visualization '96*, pp. 29–36, 1996. doi: 10.1109/INFVIS.1996.559213

[19] C. Conati, S. Lallé, M. A. Rahman, and D. Toker. Comparing and combining interaction data and eye-tracking data for the real-time prediction of user cognitive abilities in visualization tasks. *ACM Trans. Interact. Intell. Syst.*, 10(2), May 2020. doi: 10.1145/3301400

[20] A. Cremers and S. Ginsburg. Context-free grammar forms. *Journal of Computer and System Sciences*, 11(1):86–117, 1975.

[21] F. Dabek and J. J. Caban. A grammar-based approach for modeling user interactions and generating suggestions during the data exploration process. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):41–50, 2017. doi: 10.1109/TVCG.2016.2598471

[22] E. Dimara and C. Perin. What is interaction for data visualization? *IEEE Transactions on Visualization and Computer Graphics*, 26(1):119–129, 2020. doi: 10.1109/TVCG.2019.2934283

[23] C. Fan and H. Hauser. Fast and Accurate CNN-based Brushing in Scatterplots. *Computer Graphics Forum*, 2018. doi: 10.1111/cgf.13405

[24] C. Fan and H. Hauser. Personalized sketch-based brushing in scatterplots. *IEEE Computer Graphics and Applications*, 39(4):28–39, 2019. doi: 10.1109/MCG.2018.2881502

[25] A. Figueiras. Narrative visualization: A case study of how to incorporate narrative elements in existing visualizations. In *2014 18th International Conference on Information Visualisation*, pp. 46–52. IEEE, 2014.

[26] D. Gotz and M. X. Zhou. Characterizing users' visual analytic activity for insight provenance. *Information Visualization*, 8(1):42–55, 2009.

[27] A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005.

[28] H. Guo, S. R. Gomez, C. Ziemkiewicz, and D. H. Laidlaw. A case study using visualization interaction logs and insight metrics to understand how analysts arrive at insights. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):51–60, 2016. doi: 10.1109/TVCG.2015.2467613

[29] R. Hadsell, D. Rao, A. A. Rusu, and R. Pascanu. Embracing change: Continual learning in deep neural networks. *Trends in Cognitive Sciences*, 24(12):1028–1040, 2020. doi: 10.1016/j.tics.2020.09.004

[30] J. Heer. Agency plus automation: Designing artificial intelligence into interactive systems. *Proceedings of the National Academy of Sciences*, 116(6):1844–1850, 2019.

[31] U. Hinrichs, H. Schmidt, and S. Carpendale. Emdialog: Bringing information visualization into the museum. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1181–1188, 2008. doi: 10.1109/TVCG.2008.127

[32] M. S. Horn, B. C. Phillips, E. M. Evans, F. Block, J. Diamond, and C. Shen. Visualizing biological data in museums: Visitor learning with an interactive tree of life exhibit. *Journal of Research in Science Teaching*, 53(6):895–918, 2016.

[33] S. Kullback and R. A. Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[34] S. Latif, S. Chen, and F. Beck. A deeper understanding of visualization-text interplay in geographic data-driven stories. *Computer Graphics Forum*, 40(3):311–322, 2021. doi: 10.1111/cgf.14309

[35] Y. Li, S. Bengio, and G. Bailly. Predicting human performance in vertical menu selection using deep learning. p. 1–7, 2018. doi: 10.1145/3173574.3173603

[36] B. Liu et al. Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666, 2010.

[37] J. Ma, I. Liao, K.-L. Ma, and J. Frazier. Living liquid: Design and evaluation of an exploratory visualization tool for museum visitors. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2799–2808, 2012. doi: 10.1109/TVCG.2012.244

[38] T. Milo and A. Somech. Next-step suggestions for modern interactive data analysis platforms. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 576–585, 2018.

[39] D. Moritz, B. Howe, and J. Heer. Falcon: Balancing interactive latency and resolution sensitivity for scalable linked visualizations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2019.

[40] C. Nobre, D. Wootton, Z. Cutler, L. Harrison, H. Pfister, and A. Lex. revisit: Looking under the hood of interactive visualization studies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2021.

[41] A. Ottley, R. Garnett, and R. Wan. Follow the clicks: Learning and anticipating mouse interactions during exploratory data analysis. In *Computer Graphics Forum*, vol. 38, pp. 41–52. Wiley Online Library, 2019.

[42] A. Ottley, H. Yang, and R. Chang. Personality as a predictor of user strategy: How locus of control affects search strategies on tree visualizations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, p. 3251–3254. Association for Computing Machinery, New York, NY, USA, 2015. doi: 10.1145/2702123.2702590

[43] R. Rehurek and P. Sojka. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*. Citeseer, 2010.

[44] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization*

and Computer Graphics, 23(1):341–350, 2017. doi: 10.1109/TVCG.2016.2599030

[45] A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):659–668, 2016. doi: 10.1109/TVCG.2015.2467091

[46] A. Slingsby, J. Dykes, J. Wood, and R. Radburn. Designing an exploratory visual interface to the results of citizen surveys. *International Journal of Geographical Information Science*, 28(10):2090–2125, 2014. doi: 10.1080/13658816.2014.920845

[47] P. T. Sukumar, G. J. Martinez, T. Grover, G. Mark, S. K. D'Mello, N. V. Chawla, S. M. Mattingly, and A. D. Striegel. Characterizing exploratory behaviors on a personal visualization interface using interaction logs. *EuroVis 2020-Short Papers*, 2020.

[48] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[49] C. Walchshofer, A. Hinterreiter, K. Xu, H. Stitz, and M. Streit. Provectories: Embedding-based analysis of interaction provenance data. 2020.

[50] E. Wall, A. Arcalgud, K. Gupta, and A. Jo. A markov model of users' interactive behavior in scatterplots. In *2019 IEEE visualization conference (VIS)*, pp. 81–85. IEEE, 2019.

[51] M. O. Ward and J. Yang. Interaction spaces in data and information visualization. In *VisSym*, pp. 137–145, 2004.

[52] J. Wenskovitch, M. Zhou, C. Collins, R. Chang, M. Dowling, A. Endert, and K. Xu. Putting the "i" in interaction: Interactive interfaces personalized to individuals. *IEEE Computer Graphics and Applications*, 40(3):73–82, 2020. doi: 10.1109/MCG.2020.2982465

[53] W. Willett, J. Heer, and M. Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129–1136, 2007. doi: 10.1109/TVCG.2007.70589

[54] K. Wongsuphasawat, J. A. Guerra Gómez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. Lifeflow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, p. 1747–1756. Association for Computing Machinery, New York, NY, USA, 2011. doi: 10.1145/1978942.1979196

[55] A. Wu, Y. Wang, M. Zhou, X. He, H. Zhang, H. Qu, and D. Zhang. Multivision: Designing analytical dashboards with deep learning based recommendation. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):162–172, 2021.

[56] T. Wu, M. Terry, and C. J. Cai. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. *arXiv preprint arXiv:2110.01691*, 2021.

[57] K. Xu, A. Ottley, C. Walchshofer, M. Streit, R. Chang, and J. Wenskovitch. Survey on the analysis of user interactions and visualization provenance. In *Computer Graphics Forum*, vol. 39, pp. 757–783. Wiley Online Library, 2020.

[58] J. S. Yi, Y. a. Kang, J. Stasko, and J. A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, 2007. doi: 10.1109/TVCG.2007.70515

[59] J. Zhao, M. Glueck, S. Breslav, F. Chevalier, and A. Khan. Annotation graphs: A graph-based visualization for meta-analysis of data based on user-authored annotations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):261–270, 2017. doi: 10.1109/TVCG.2016.2598543

[60] Z. Zhou, X. Wen, Y. Wang, and D. Gotz. Modeling and leveraging analytic focus during exploratory visual analysis. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2021.