Understanding and Automating Graphical Annotations on Animated Scatterplots

Danqing Shi, Antti Oulasvirta, Tino Weinkauf, Nan Cao

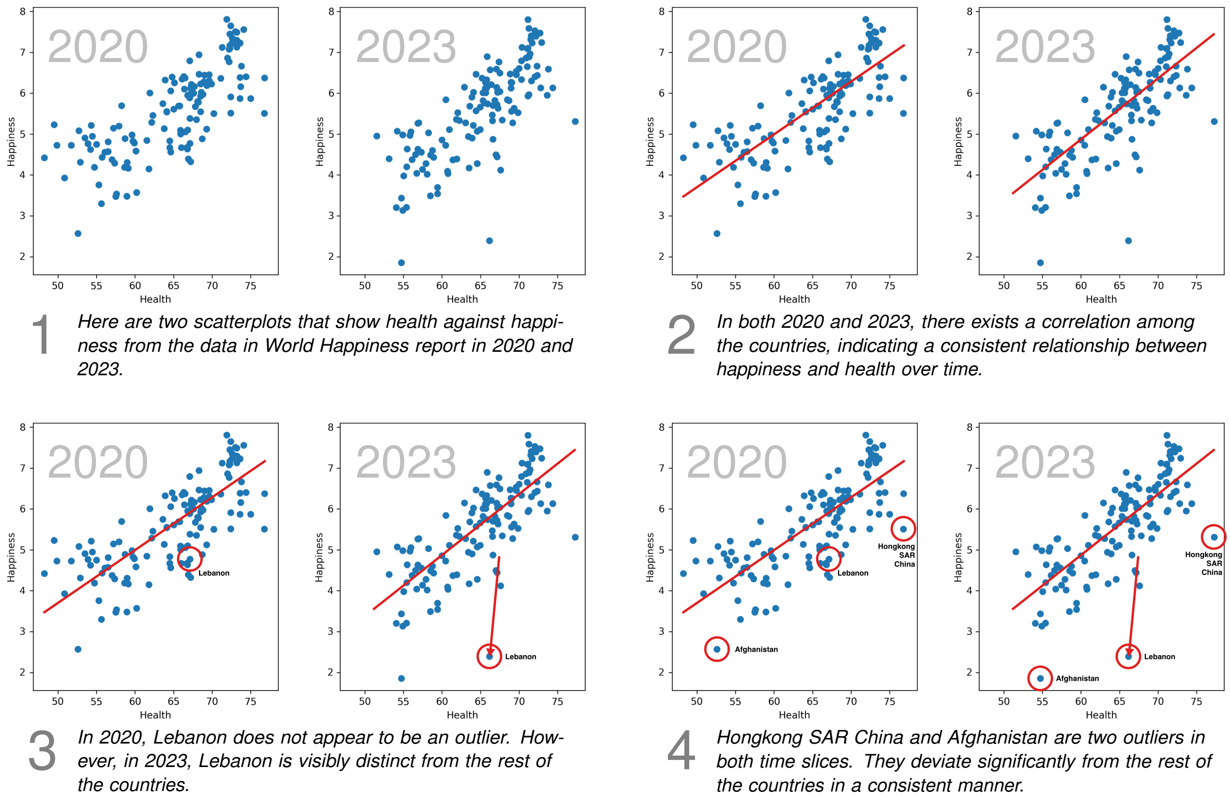


Fig. 1: We present a method for automatically annotating scatterplots for communicating key information in data. The graphical annotations are added one by one with captions, creating a narrative that covers the main data insights in an easy-to-follow order. The figures above provide an example case from the World Happiness Report in 2020 and 2023 (refer to Section 5.2 for more details).

Abstract— Scatterplots are commonly used in various contexts, from scientific publications to infographics for the general public. However, not everyone is able to read them, and even experts may struggle to notice some important information such as overlapping clusters or temporal changes. To address these issues, a computational approach for annotating scatterplots has been developed. This approach involves various forms of annotation, including drawing lines to show correlations, circling areas to show clusters, and indicating movement with arrows. The approach is based on a study that identified common annotation strategies used by people to annotate scatterplots. These strategies are distilled into an automated method for generating graphical annotations on scatterplots. The method involves a problem formulation using a Markov Decision Process and a model for making annotation decisions. The model generates step-by-step graphical annotations by analyzing data insights and observing the chart. The final result conveys a narrative that is easy to understand and allows for the conveyance of temporal changes in the data. The study results suggest that the method can generate understandable and functional annotations that are comparable to those created by human experts. This approach can potentially reduce the time and effort required to read scatterplots, making it a useful tool for data visualization novices.

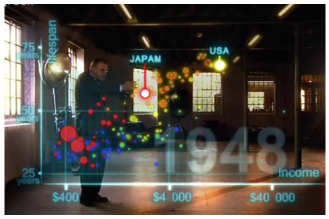
Index Terms—Annotated Visualization, Scatterplot

- *Danqing Shi and Antti Oulasvirta are with Aalto University. E-mail: {danqing.shi, antti.oulasvirta}@aalto.fi.*
- *Tino Weinkauf is with KTH Royal Institute of Technology. E-mail: weinkauf@kth.se*
- *Nan Cao is with Tongji University. E-mail: nan.cao@gmail.com.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

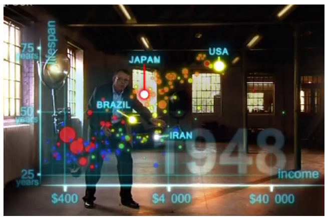
1 Introduction

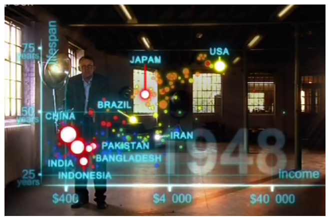
Graphical annotations are visual cues that highlight key information in charts, helping to externalize data facts and make sense of datasets [8, 12, 30]. For instance, lines can be added to show trends or aggregate values in data, and points can be circled to draw attention to specific aspects of data distributions [9, 22]. Well-designed annotations improve the readability of visualizations, guide users' attention, avoid misinterpretations, and allow people to efficiently communicate data facts to the audience [5, 37]. Prof. Hans Rosling's video on socio-economic differences among countries offers a classic example of how graphical annotations can be used to communicate data insights



1 The scatterplot encodes **income per person** (x) and **life expectancy** (y) in the year 1948.

2 *"The United States was in the front. Japan was catching up."*





3 *"Brazil was way behind. Iran was getting a little richer from oil, but still had short lives."*

4 *"The Asian giants, China, India, Pakistan. . . , they were still poor and sick down here."*

Fig. 2: Sequentially added annotations can express patterns in a dataset in a coherent and understandable way to a viewer. Prof. Hans Rosling sequentially annotated an animated scatterplot to describe the socio-economic state of the world after the Second World War [34].

clearly [34]. As illustrated in Figure 2, he annotated notable countries step-by-step to discuss the differences between the countries of the world after the tragedies of the Second World War. This successful video showcases the power of using sequential graphical annotations that can be easily followed by the general audience.

However, it is still an open question how users with data skills add graphical annotations to visualization charts that fit with the chart and communicate the relevant information effectively. It leads to a further challenge that annotations have to be done case by case, which depends on humans. Although existing visualization tools [33] support advanced features for annotation, the current process is manual or semi-automated, and depends on the skill level of the user. This is not an easy task for people without "visualization literacy" [5]. As the number of data insights in the chart grows, the annotation process becomes even more tedious. We believe that a better understanding of graphical annotations and a computational method can assist people in creating effective annotations, making the process less dependent on user expertise and reducing the amount of work needed.

In this paper, we propose a computational approach that generates graphical annotations, such as lines, arrows, and circles, in a meaningful order to convey a narrative about the data. Each of the annotations can enhance visualizations by highlighting and explaining a certain noteworthy feature in the data. Rather than overwhelming the viewer with all annotations at once, our approach adds them one at a time, accompanied by captions, for clarity (see Figure 1). Our study focuses on scatterplots, which are one of the most commonly used types of statistical charts that display rich visual patterns. We specifically consider animated scatterplots in our study, as they allow us to analyze insights not only in a single scatterplot but also in the relationship between different time periods. This is particularly useful in narrative visualization, as exemplified by Hans Rosling's video.

To enable this approach, we first gained empirical knowledge about how people with data skills use annotations. We conducted a research study with twelve students from a university who possessed data analysis skills and had prior experience in analyzing data using Excel or R and displaying data through statistical charts. This study is similar to the previous research on student annotations [31] but focuses on annotating animated scatterplots and analyzing strategies. Through our analysis, we were able to understand the preferences and strategies that people use when annotating animated scatterplots.

We formulate annotation as a sequential decision-making problem, model it as a Markov decision process (MDP) and solve it using Reinforcement Learning (RL). While a rule-based approach is transparent and easy to understand, we chose RL because RL models can adapt to diverse conditions without the need for manual rule additions and maintain a certain level of explainability. Moreover, RL is more suitable for interactive use than online combinatorial optimization because once an RL policy has been trained offline, it can be executed in real-time. We then designed a model architecture that analyzes data, observes charts, and considers past annotations to generate new ones. We use a reinforcement learning algorithm to train the policy of the model to solve the annotation problem by considering statistical insights, narrative order, and visual presentation.

This computational approach suggests applications in daily work with data. We demonstrate its usefulness through an application case with real-world datasets to show how general people can benefit from it. With the help of automated graphical annotations, people can quickly gain insights from complicated scatterplots without requiring any visualization skills. To evaluate the effectiveness of our technique, we conducted a formal user study with 60 participants. We compared the sequential annotations generated by our policy with baseline approaches. The results show that the sequential annotations generated by our approach are comparable to human ground truth and better than other baselines.

To summarize, this paper has made three main contributions:

- A formative study to gain insights into the strategies used for drawing graphical annotations on scatterplots to present data.
- A computational method based on reinforcement learning that automates the graphical annotations on scatterplots. This includes a problem formulation that uses Markov Decision Process and a model architecture.
- An application case and a user study to evaluate the usefulness and effectiveness of the proposed approach.

2 Related Work

This section provides an overview of previous research on annotations in visualization, authoring tools, and automated tools.

2.1 Annotations in Visualization Research

Annotations are graphical or textual additions that are associated with one or more existing visualization elements [30]. They are used to highlight key information in a chart and can assist data analysis [22] as well as reporting observations for further analysis [17]. According to Heer and Shneiderman [17], annotations can be both textual and graphical. They also mentioned that an annotation can be made "data-aware" when realized as selections. Kong and Agrawala [24] classified annotations into two types: annotations associated with marks and freehand annotations. Additionally, annotations can be incorporated into the taxonomy of visual cues, including textual additions such as summary statistics, tooltips, labels, and glyphs such as shapes, brackets, and arrows [23]. Annotations can be used in narrative argumentation with data [23, 37]. A visual annotation, such as highlighting, can be used to draw the viewer's attention to a specific area of the chart [28]. Choe and Lee [10] identified eight annotation types to assist general users in communicating data insights and investigated their usage frequency. Their statistical results showed that labels and shapes were the most commonly used annotations, followed by the trend line, color, and value line. Ren *et al.* [33] surveyed 106 annotated visualizations to characterize a design space of annotations in visual data-driven storytelling. Their design space distinguished four forms of visual annotation (text, shapes, highlights, and images) as well as four annotation target types (data items, structural chart elements, coordinate spaces, and prior annotations). In this paper, we focus on scatterplots, one of the most commonly used statistical charts with rich visual patterns. Annotations in scatterplots reduce the user's cognitive load and make communication more efficient [20]. Specifically, our study focuses on graphical annotations on scatterplots that can directly emphasize and explain information.

2.2 Authoring Tools for Annotation

Manual annotation is typically required for charts and visualizations. Commercial visualization software provides basic annotation features for convenience. For instance, Tableau has three types of annotations: annotating selected marks, specific points, and areas in the view. Matplotlib allows users to add textual annotations with an arrow pointing to the given position. The research community has developed several annotation tools for visualization. Kong and Agrawala proposed a system that applies user-selected annotations to existing chart bitmaps [24]. Ellipsis is a system that combines a domain-specific language for data story authoring with dynamic annotations [36]. ChartAccent is a more expressive tool that generates manual and data-driven annotations by utilizing a palette of annotation interactions [33]. SmartCues provides multitouch interactions to assist users in constructing queries and generating data-aware annotations [42]. Vodar [41] augments visualization charts with interactive annotations to facilitate interpretation. AutoClips [38] incorporates annotations in chart animations to explain insights more clearly. Recently, a tool has been proposed for annotating line chart images to identify potentially misleading visual elements [14]. Despite providing users with complete control, these authoring tools require users' expertise to decide which data items to annotate and which annotation methods to use. This paper examines the annotation practices of human users and proposes a computational method to automate this task.

2.3 Semi-Automated and Automated Tools for Annotation

Semi-automated tools are designed to simplify and streamline the process of annotation authoring and data presentation. For instance, Contextifier [18] is a tool that generates an annotated line chart from a news article, which is a process similar to text-to-visualization. NewsViews [15] is an automated news visualization system that produces annotated maps without requiring human designers. VisAnnotator [25] is another text-to-annotation process that takes a chart image and a text description as input to automatically annotate chart images based on the given text. These tools automate the annotation process but require a textual description. Roslingifier [40] is a semi-automatic storytelling system that assists expert presenters in creating Hans Rosling-style animated scatterplots. The authors define a design space for effective data presentations and identify three key techniques: natural language narratives, visual effects highlighting events, and temporal branching for animation playback. Roslingifier provides an interactive interface for semi-automated authoring of data stories. In this work, we aim to provide a fully automated way of interpreting the scatterplot itself, focusing specifically on the automated annotations of scatterplot images.

We are aware of two fully automated approaches for generating annotations for charts: Just-in-time framework [20] and Temporal Summary Images [7]. The first work provides a pipeline for summarizing observable information by identifying all visual features and determining the semantics of each feature. It enumerates all possible patterns according to a score ranking. The second work focuses on temporal visualizations and uses a greedy algorithm to decide whether or not to show an annotation based on density. They approaches provide an end-to-end recommendation, starting with a single visualization chart and ending with an annotated chart. In addition to these works, there is also automated insight generation, also known as visualization recommendation, which can help reduce user burden in exploring and understanding data. Our approach, however, is different from these works as it provides step-by-step annotations for an informative and readable visual presentation. Instead of creating charts from datasets, our approach presents data facts for a specific chart, which serves as an assistant to the visualization.

3 Understanding Annotation Strategies

We conducted a formative study to understand how experienced users with data analysis skills annotate and explain scatterplots.

3.1 Study

The study focuses on understanding how to annotate animated scatterplots to effectively communicate and present data. Unlike a previous study that analyzed student annotations on bar charts [31], this study chooses a similar user group but extends to animated scatterplots and further analyzes strategies. The primary goal of the study is to understand the strategies used for annotation, such as the types of annotations that are chosen intuitively (e.g., circle, arrow, line) and the order in which annotations are drawn. For instance, the study seeks to determine whether experienced users start with outliers or tend to focus on tendencies in the data.

Participants We recruited a total of twelve participants from a university campus, which comprised of eight females and four males. Their ages ranged from 22 to 29 years old. Out of the twelve participants, six of them had a Master's degree, five had a Bachelor's degree, and one was an undergraduate student. All participants reported having experience in data analysis and visualization, including analyzing data with Excel or R and presenting data with statistical charts. Four participants reported that they had the visualization course at the university, and else they learned the analysis skill online or from other classes. None of the participants reported having any visual impairments.

Material The participants were requested to annotate animated scatterplots at various timeslices. Animated scatterplots were chosen to enable the participants to gain insights from both individual scatterplots and the relationship between two scatterplots, which is common in narrative visualization. The study consisted of twelve tasks, each of which presented a scatterplot as an animation, depicting changes in the data (see study interface in Appendix **??**). Two corresponding static images showed the initial and final frames of the scatterplot sequence. Participants were able to use the "draw" function in Microsoft Word to annotate the scatterplot images freely. In addition, a blank space was provided for each scatterplot image to allow participants to record their annotation explanations in a sequential manner. The scatterplots were generated using synthesized data based on the defined design space of scatterplots [35], using the artificial data generator from Scikit-Learn.

Procedure The study was conducted in an office setting, where participants were invited to take part in person. They were instructed to observe a scatterplot animation for each task, think about what happened in the data and highlight relevant graphical annotations. Participants could view the animated scatterplot as many times as needed to discover insights. They were encouraged to consider effective ways of annotating the scatterplots to communicate the insights to the audience. After each annotation task, participants described and explained their annotations in sequence. Upon completing all tasks, participants summarized their overall annotation strategies. The study concluded with a brief demographic survey. On average, participants spent 51 minutes (STD = 9.6) on the study, receiving 15 euros as compensation.

Data Processing A total of 144 tasks were completed. Each observation included a sequence of annotations made by a participant to explain insights in a scatterplot, resulting in a total of 350 annotations. The author's team was responsible for labeling all 350 annotations. The labeling process involved characterizing the annotations along two dimensions: 1) data insights and 2) annotation methods. To achieve this, the insights were informed by the low-level components of analytic activity [3] and a taxonomy of data facts [39]. Additionally, statistics on frequently used annotation methods [10] were used.

3.2 Results and Analysis

3.2.1 Usage of Annotations

Based on our analysis, we found that the participants primarily annotated five types of data insights. These included correlation (26.0%), cluster (17.7%), outlier (18.0%), distribution (18.0%), and temporal change (20.3%). In statistics, correlation refers to the degree of association between two variables. Clusters, on the other hand, are groups of points that are spatially close to one another. Outliers are data points that are located far away from other data points. Distribution shows the pattern of how data points are spread out in the data space. Finally, temporal change refers to the movement of data points between two scatterplots at different times.

During our analysis, we collected statistics on the types of annotations used by participants to annotate their insights. We identified five common types of annotations: circle, line, arrow, rectangle, and freehand draw. On average, each participant used 3.33 different types of annotations throughout the tasks. For all insights except distribution, the most frequently used annotation type was used at least 80% of the time to describe the insight. Figure 3-A shows the preferred annotation methods for each data insight. Circle was the most commonly used annotation, with 43% usage and was used by every participant. It was mainly used for showing clusters (90.3%) and outliers (90.5%). Line was the second most commonly used annotation, primarily used for annotating correlations (84.6%). To show the temporal changes of data points, participants tended to use arrows to show the direction and distance. Sometimes, participants used rectangles or freehand drawings to present findings, especially to show the shape of the data distribution.

3.2.2 Order of Annotations

We analyzed the strategies used by participants for drawing annotations. We discovered the order of annotated insights, which are illustrated in the Figure 3-B. Here are some of the major findings: The most noticeable insight for participants was correlation, which was usually annotated at the beginning of the analysis, and less frequently towards the end. The observation of clusters was another prevalent insight in the data. Participants tended to annotate it primarily at the beginning of their analysis and less frequently at subsequent stages. The next most frequently annotated insight was related to the overall distribution of the entire dataset. This was often noted in the middle of the analysis, after an initial insight. Participants generally noted this insight with a more evenly distributed frequency across their annotations. It was observed more frequently in the middle of their analysis. Outliers were annotated with a frequency comparable to the previous insights. However, they were the only category of insight that occurred more frequently towards the end of the stated insights. Additionally, we found that half of the participants ended the annotation process in two steps, and most of the rest ended in three steps. In 76 tasks, participants made at least three annotations. Only a few participants continued to add more annotations after three steps.

The trends for the types of annotations used are not as clear as those for the insights gained from the data analysis (see Figure 3-C). The usage of circles and lines is more frequent in the initial stages of the analysis compared to the later stages. Arrows are used consistently throughout the analysis, particularly to highlight changes over time. Additionally, the use of rectangles and freehand drawings remains relatively constant over time, as they are typically used for annotating the overall distribution of the data, often at a later point in the analysis.

Our analysis identified the four most widespread strategies used by experienced users:

Overview first, then details. (66%) The first strategy studied was the dominant one. The idea behind this strategy is similar to the visual information-seeking mantra, which is “overview first, zoom and filter, then details-on-demand.” Participants who used this strategy annotated all the overall insights before annotating any details or movement-related insights. An example of this strategy is illustrated in Fig. 4-b, where circles were used to show data clusters and correlation, followed by circling individual outliers. Participants’ comments also revealed similar summaries, such as P10’s strategy: "*I start by looking at the clearest grouping of points in the data (if any). I then check how the points move between these groups and if there are any similarities/differences. Lastly, I look at what sort of outliers/exemptions there would be to these grouping/transformations between the graphs.*" P5 also commented: "*First, I marked the most salient patterns (linear associations or clustering of data points, mostly). These I did for both plots. After the most salient ones, I kept a closer look at the movement of data points and potential outliers.*"

Tracing temporal changes. (16%) Participants also like to annotate temporal changes from animation (Fig. 4-c). In this strategy, participants started by annotating movement before overall or detail-related

A) The frequency of graphical annotations regarding the insights

	correlation	cluster	outlier	distribution	change
line	77	5	0	9	1
circle	4	56	57	27	5
arrow	7	0	6	0	59
rectangle	0	0	0	12	4
freehand	3	1	0	15	2

B) Annotated insights in the order

C) Annotation methods in the order

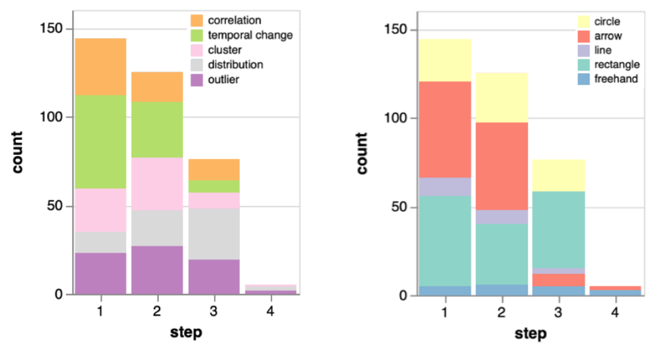


Fig. 3: This figure illustrates statistics in the study: distributions of annotation methods for different insights and the frequency of annotated insights and used methods in different steps of the annotation process.

insights. This includes instances where users only annotate movement insights. For instance, P3 suggested annotating the overall animation, such as moving direction, and using arrows to indicate it. Participants also reveal a preference for annotating the movement of outliers and clusters, with one even annotating an outlier before it becomes one (“*Pay attention to this point. It will become an outlier in the next frame.*”). Additionally, there was a participant who drew shapes on empty space to indicate distribution and temporal change, rather than the data points themselves. An example of this was a participant who noticed that there were dots before the animation, but after the animation, all dots moved around and left a blank space (“*There were dots before animation, but after animation, all dots moved around and left this blank space.*”).

Details in the middle. (11.1%) The strategy involves three steps: First, the participants annotate an overview insight. Second, they provide detailed insight. Third, they conclude with another overview insight. For instance, P2 did this strategy in one task. Initially, he annotated an overview pattern to demonstrate that data points almost form a perfect correlation. Then, he drew an arrow and a circle indicating that one point moves from the middle to the bottom right corner, not following the others in the correlation. Finally, he drew several arrows to show the general movements of all points. Based on our observations, we noticed that this strategy is often used in conjunction with the preference to annotate temporal change in the middle.

Others. (6.9%) This last part includes unusual strategies. In some trials, participants began by annotating detailed insights, while in others, they only annotated detailed insights. Additionally, there was one instance of “overview in the middle”. This category is not given a separate name as it only appeared a few times in the study results.

3.3 Sequential Annotations

Participants annotate scatterplots by following a step-by-step manner to communicate insights in a sequential order, which can result in various possible permutations of sequential annotations. They take into account the existing annotations and how a new annotation would impact future decisions. Even when working with the same data, different sequences of annotations can significantly alter the meaning of data presentation. For example, one participant annotates the trend line and major clusters to illustrate the overall change in the data pattern and then highlights the detailed outliers (Figure 4-b). This annotation sequence conveys that the general change of the data is from a correlation to three clusters,

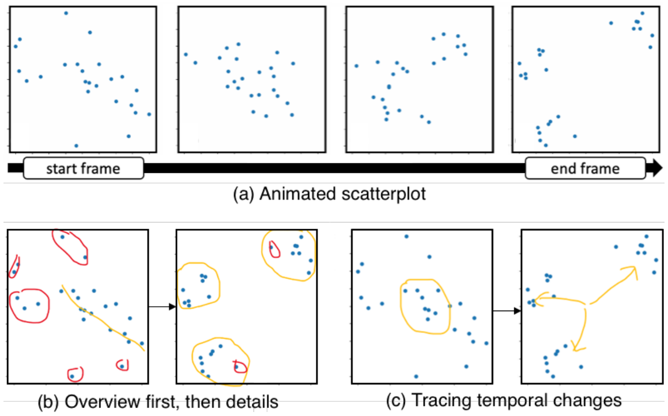


Fig. 4: (a) Participants were shown animated scatterplots that progressed from a starting frame to an ending frame. Two different strategies were used to show the scatterplots: (b) *Overview first, then detail*: This involved drawing yellow circles on the clusters in the right scatterplot, followed by straight lines to show the linear association in the left. Finally, little red circles were drawn to indicate outliers; (c) *Tracing temporal changes*: A circle was drawn to illustrate the initial position of the points, followed by arrows to show that they moved in three different directions.

and some outliers need attention during this change. On the other hand, another participant may focus on a different aspect by annotating the distribution of the initial data points first, and then drawing arrows to show the movement of these data points (Figure 4-c). This annotation sequence conveys a different meaning that the data points are initially distributed in the center of the area. However, they move in different directions with the change of time. Of course, there are many other options available. As the number of insights and annotation methods increase, the possible permutations of sequential annotations will increase factorially ($\sum_{n=0}^{N} n!$).

Based on this finding from the formative study, we formulate the annotation process as a sequential decision-making problem, assuming annotations for a data presentation are added step-by-step. Furthermore, we consider keeping all annotations in sequential order to maintain the narrative flow of the short-term presentation, just like the participants did in the study. Compared with replacing the annotation by removing the previous ones, adding new visual information over the previous one has a lower transition cost, which is preferred by the audience [21]. By sequentially preserving annotations, viewers can follow a coherent narrative that shows the evolution of insights over time. It enhances comprehension and storytelling in data visualizations.

3.4 Design Implications

We derive the following implications from the analysis to the design of automated sequential annotation:

- **Informative insights.** Most annotations present meaningful statistical insights, even if the information is obvious. Annotations serve as a way to communicate data and note observations, and their objective is to make charts more meaningful. They are not only used to analyze data but also to highlight and emphasize intended messages.
- **Understandable order.** Most experienced users prefer an order that presents annotations in a general-to-specific structure, following the "Overview first, then details" strategy. When there are differences in patterns between two scatterplots, "Tracing temporal changes" are also frequently considered.
- **Clear visual presentation.** Experienced users prefer certain annotation methods to be used on corresponding insights in order to avoid ambiguity. They also avoid cluttering the view with too many annotations, stopping after three annotations to prevent the information from becoming hard to read.
- **Text explanations.** Text descriptions are provided to explain graphical annotations, which can help interpret insights and emphasize messages. With explainable captions, annotations can be communicated better to general users.

4 Automatically Annotating Scatterplots

This section describes the automated annotation approach that relies on the design implications from the empirical study. We begin by formulating the annotation problem and then introduce a model that solves this problem.

4.1 Problem Formulation

We view step-by-step annotations as a sequential decision-making process on visualization [19,45], and hence, we use the Markov Decision Process (MDP) to formulate the chart annotation problem. The MDP accurately represents the decision problem and also connects the theoretical problem with practical algorithms. Figure 5 illustrates the agent-environment interactions. In this framework, the agent is an annotator who takes the observation of the environment as input, which in this case is the scatterplots on the right. The agent then produces annotations to add to the chart. This process is iterative, as the chart in the environment updates after adding the annotation. The annotator represents the environment as a state, makes decisions, and receives a reward signal from the environment to get feedback on its performance.

Formally, we can define the MDP as a tuple $< S, A, R, T >$:

- $S = \{s_1, s_2, \cdots, s_m\}$ is the state space. Each state s_t is a feature vector that represents the environment. It includes statistical information and visual presentation of the scatterplot, and captures historical information of past annotations.
- $A = \{a_1, a_2, \cdots, a_n\}$ is the action space. Each action a_t determines the addition of a specific annotation associated with a data insight. The action space is constant based on supported annotation types and insights.
- R is a reward function $R(s_t, a_t)$ that quantifies the improvement of the annotated scatterplot after taking action a_t.
- T is the transition function that facilitates the transition from one state to another. In the current setting, the transition function is deterministic and known in advance. The state of the chart changes by adding a new annotation.

The policy π is the core of the annotator. It takes the current state s_t as input and produces an action a_t as output. Based on the MDP model, the problem of chart annotation is to find an agent model with an optimal policy π^* that maximizes expected episode rewards by taking a sequence of actions.

4.1.1 State

When designing the state representation, we take into account both the underlying data and the visual representation of a scatterplot. Since the scatterplot and data size can be quite large for large datasets, we need to compress the environment to support efficient decision-making [2]. The data information we consider is the statistical insights based on the given two columns instead of the entire dataset, while the visual information we consider is the encoded feature vector from the pixels of the scatterplot. Moreover, we need to consider historical information since we present sequential annotations as the final result. Therefore, we encode all past annotations into a history feature vector to guide future decisions. Instead of observing the entire dataset, the state to the policy is an abstract feature represented by a feature vector that concatenates the features extracted from the three components of the annotator model: Data, Chart, and Annotations (Section 4.2).

4.1.2 Action

An action is a decision to add a graphical annotation. The process of taking action is similar to how users interact during manual annotation using visualization tools such as Tableau and ChartAccent [33]. First, users decide if they want to add an annotation. Next, they select a

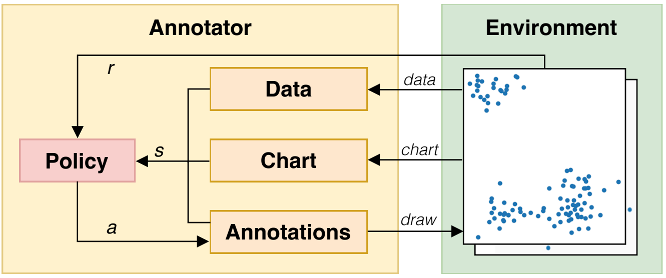


Fig. 5: In agent-environment interactions, the annotator uses three modules to represent the state (s): a data module for analyzing data insights, a chart module for perceiving visual presentation, and an annotations module for recording past annotations. The annotator takes an action (a) to draw an annotation on the scatterplot and receives a reward (r) as feedback from the environment.

target, and finally, they determine the type of annotation they want to add. This process conceptualizes an annotation as a discrete action $a := (continue, insight, method)$ The *continue* option is a binary decision determining whether to stop the annotation process early. The *insight* and *method* are one-hot vectors that refer to statistical insights and the type of annotation with a given target. We have covered annotation methods from our formative study in the current action space. If there are other effective choices from future studies, we can easily extend the action space to include them.

To help viewers understand the annotations better, we have included captions below the scatterplot that sync with the step-by-step annotations. Each caption describes the focus of the annotation and the insights based on the statistics. We have analyzed the user's explanations and included three different temporal changes for each statistical insight. These changes are represented by the terms "Birth" and "Death" which signify the appearance or disappearance of an insight from the first scatterplot to the second one. The term "Continuation" is used for consistent insights in both scatterplots. We have provided the initial captions in Table 1 using a template-based method. These captions can be further improved by manual revision or automatically refined by advanced language models such as GPT4/ChatGPT.

4.1.3 Reward

Reward function R is a key part of the chart annotation problem that guide the agent to annotate scatterplots for effective communication. As annotations are presented sequentially to the viewers, we give a positive reward to the annotation, which is placed in a right place and in a right order to produce a meaningful message. Based on the design implications from the study, we involve three major factors when designing the reward for adding annotations on scatterplots, including making the chart informative, ensuring a proper display order, and providing a clear visual presentation. Formally, when performing an annotation action a_t to a visualization state s_t, we define the reward function as the weighted sum of insight, order, and visual reward: $R(s_t, a_t) = w_1 \cdot r_{\text{insignt}} + w_2 \cdot r_{\text{order}} + w_3 \cdot r_{\text{visual}}$, where the weights can be adjusted to modify the preference of each objective. In our implementation, we use the same weight for all three components.

Insight Reward r_{insignt}. Annotations that provide statistical insights are highly valued as they reduce ambiguity and cognitive load for users. Agents will be rewarded for annotating insights featured by the data module. The Pearson's coefficient [11] is used for analyzing correlation, DBSCAN [13] is used for analyzing clusters to identify multiple clusters and compact distribution (1 cluster), and the Local Outlier Factor (LOF) algorithm [6] is used for analyzing outliers. Additionally, we also offer insight rewards for annotations on the target that have statistical insight on another time slice. The insight reward component is calculated using the formula $r_{\text{insignt}} = IS_{insight}(D(a_t))$, where $D(a_t)$ is the data target of the annotation, and $IS_{insight}(\cdot)$ gives a reward of 1 when the statistical insight is detected by the method, otherwise, it is 0. For instance, the insight reward of correlation will be 1 if the absolute

Insight	Change	Caption Templates
Outlier	Birth	In the (first scatterplot), (point) does not appear to be an outlier. However, in the (second scatterplot), this (point) is visibly distinct from the rest of the (data).
	Death	In the (first scatterplot), an outlier stands out. However, it shifts closer to other data in the (second scatterplot).
	Continuation	The (point) is an outlier in both time slices. It deviates significantly from the rest of the (data points) in a consistent manner.
Correlation	Birth	In the (first scatterplot), there is no apparent correlation. However, in the (second scatterplot), a correlation emerges, indicating a relationship between (x) and (y).
	Death	In the (first scatterplot), a correlation exists among (points) , indicating a relationship between (x) and (y). However, it disappears in the (second scatterplot).
	Continuation	In both the (first and second scatterplot), there exists a correlation among the (points), indicating a consistent relationship between (x) and (y) over time.
Cluster	Birth	In the (first scatterplot), the (points) form *m* distinct clusters. However, in the second plot, the number of clusters increases to *n*.
	Death	In the (first scatterplot), the (points) form *m* clusters. However, in the (second scatterplot), the number reduces to *n*.
	Continuation	The same clusters are visible in (first and second scatterplots). There is a consistent relationship between the variables over time.
Distribution	Birth	In the (first scatterplot), the (points) between (x) and (y) are spread out. However, in the (second scatterplot), the distribution becomes more compact.
	Death	In the (first scatterplot), the (points) have a compact distribution. The points are located closely together. However, in the (second scatterplot), the distribution becomes more scattered.
	Continuation	In both the (first and second scatterplots), the (points) have a compact distribution.

Table 1: These caption templates are used to explain annotations to the audience. Text in parentheses can be replaced with meaningful words corresponding to the data, such as column name and cell value.

Pearson correlation is over 0.5 with significance less than 0.05.

Order Reward r_{order}. Users will be presented with sequential annotations step by step, and it is important to organize them in a proper display order to guide viewers [19]. To accomplish this, we have implemented a positive reward term for annotating a specific information a_t after a general one a_{t-1}. For example, starting with an overall correlation and then moving on to a specific outlier. In addition, we also reward insights into temporal changes that are grouped together following the "Tracing temporal changes" strategy. The order reward component is calculated using the formula $r_{\text{order}} = IS_{order}(D(a_t) \in D(a_{t-1})) + IS_{temp}(a_t, a_{t-1})$, where $IS_{order}(\cdot)$ is 1 if $D(a_t) \in D(a_{t-1})$ holds true and IS_{temp} gives 1 if a_t and a_{t-1} share the same target but annotate in temporal order.

Visual Reward r_{visual}. For a clear and effective visualization, it's important to use a method that has a high co-occurrence rate with the insight. It's equally important to avoid cluttering the charts as well. Overusing annotations can make it harder to understand the chart and obscure the data [43]. Therefore, we add a negative cost for every annotation made by the agent. If the annotation has overlap on the same data target, there will be an additional cost. The visual reward component is calculated using the formula $r_{\text{visual}} = -\frac{|D(a_t) \cap D(s_t)|}{\max(D(a_t), D(s_t))} - \frac{|D(a_t) \cap D(a_{t-1})|}{\max(D(a_t), D(a_{t-1}))}$, where the first term computes the overlap of the data scope between the current annotation and previous annotations, and the second term computes the overlap between the current and last one.

4.2 Annotator Model

We build a reinforcement learning-based annotator model to add optimal sequential annotations. We highlight major reasons to use the reinforcement learning (RL) approach rather than others. First, given a scatterplot, there are various possible permutations of sequential annotations when considering what to annotate. Compared to other techniques, RL has been proven to be effective in complex sequential decision-making problems [27]. Next, real-world datasets are diverse. Data may change the chart design completely [35]. When there are only a few actions to consider, a rule-based approach is a good choice as it is transparent and easy to understand. However, if there are numerous options to consider across different data cases, writing a rule set that covers every possibility becomes challenging. As the state and action spaces expand, the cost of creating such a rule set would significantly increase. Compared to exhaustive online algorithms like combinatorial optimization, RL can be pre-trained offline so that real-time performance is much faster. Furthermore, it can also involve deep learning architecture, which has been proven effective and efficient in capturing rich semantic features of scatterplots [29].

In order to improve the effectiveness of chart annotation and increase the rewards, we utilize information from various sources such as data statistics, visual presentation, and past annotations, which are highly valued by experienced users. For this purpose, we have designed the Annotator model comprising of four main modules. These modules include a data module to analyze the data, a chart module to observe the chart, an annotations module to keep a record of past annotations, and a policy module that helps optimize annotation decisions to maximize rewards for each episode.

Data The data module is responsible for extracting pairwise features by computing statistical results from the underlying data of the scatterplot. Our implementation of the data module analyzes insights such as correlation, distribution, clusters, and outliers. To analyze correlation, we use Pearson's coefficient [11]. For analyzing clusters, we use the DBSCAN clustering algorithm [13] which identifies multiple clusters and compact distribution (1 cluster). To analyze outliers, we use the Local Outlier Factor (LOF) algorithm [6]. We represent each possible insight using a binary feature to determine if there is a strong statistical insight in a single scatterplot. Additionally, we use a categorical feature to indicate temporal changes such as none, birth, death, or continuation in two scatterplots. Finally, we concatenate all the features as the output of the data module.

Chart The chart module aims to compress the visual information of annotated scatterplots into hidden vectors. These vectors contain all the necessary information required to display an annotated scatterplot, including data points and annotations. The chart image is represented as a $(64,64)$ tensor, similar to the previous pixel representation of scatterplots [29]. To transform the image tensor into the target feature vector, a multi-layer CNN encoder is used. The encoder is trained using a convolutional auto-encoder architecture [4]. To make the chart module focus on the data points rather than the blank background, we optimize the MSE loss function by setting a weight ratio that assigns more weight to positions that have data points.

Annotations To ensure that we capture historical information, we encode past annotations as an additional feature. We use the auto-encoder architecture to encode existing annotations into a latent vector, and then reconstruct the same sequence of annotations. To accomplish this, the loss function is the reconstruction loss that minimizes the binary cross-entropy between the input and output. As shown in the figure, this module does not directly obtain information from the environment. Instead, it takes input from inside the annotator model.

Policy We use a reinforcement learning approach to learn the policy of the model. The agent updates the policy's parameters iteratively to aim for a better episode reward by interacting with the environment. The datasets we used in the training process are the synthetic datasets generated by Scikit-Learn and open-source real-world datasets from PyDataset and VegaDatasets. Our algorithm of choice is Proximal Policy Optimization (PPO), which is a state-of-the-art actor-critic policy gradient algorithm that consists of two actor and critic neural networks. We used two-layers fully-connected networks for both the policy and the value network. PPO is selected for its sample efficiency and ease of tuning. In the implementation, we utilized Stable Baselines 3 library's implementation of PPO.

5 Application Case

This section showcases the usefulness of our approach by presenting an application case with real-world datasets. To illustrate, let's consider a hypothetical user named Sally, who is a journalist. Sally wants to investigate two well-known public datasets namely, Gapminder [16] and the World Happiness report [44]. She aims to learn more about global development, but has limited experience in visual analysis. To assist her in learning from the data, Sally decides to use the automated annotation approach.

5.1 Gapminder 1977-2007

Sally begins her investigation by examining the Gapminder dataset [16], which provides comprehensive data on global development. The dataset contains a wealth of information on various variables related to the economy, health, education, and other aspects of development. It covers a large number of countries over an extended period, making it an excellent resource for Sally's research.

Sally's main focus is to see the relationship between life expectancy and GDP. She selects two years, 1977 and 2007, to analyze the changes that occurred before the economic crisis of 2008. The chart reveals a pattern between life expectancy and GDP within the Gapminder dataset, as shown in the first panel of Figure 6. Once Sally creates the scatterplot, she uses the annotator's help to interpret the scatterplot.

In 2007, the annotator identified and annotated three clusters from left to right and bottom to top in the second panel of Figure 6. Sally conducted a further investigation of the data points in these clusters and found that the first cluster, located at the bottom left, consisted of countries with low GDP and varying life expectancies, mainly from Africa. The second cluster, located in the middle, included countries with relatively low GDP but significantly high life expectancies, primarily from Asia and the Americas. The third cluster, situated at the top right, featured countries with higher GDP and longer life expectancies, predominantly from Europe.

The annotator then identify more detailed findings (third panel of Figure 6). In particular, the annotator circled a country in 1977 with an extremely high GDP and traced its movement to its corresponding position in 2007. Sally investigated this data point further and found out that the country was Kuwait. This Asian country experienced a significant increase in GDP in the 1970s as it was one of the world's largest oil producers and exporters. The annotator also circled Saudi Arabia, which had the second-highest GDP in 1977 but became a medium value in 2007. However, life expectancy in Saudi Arabia increased significantly. Lastly, the annotator circled Gabon, which had a relatively high GDP but low life expectancy in 1977. In 2007, the situation in Gabon slightly changed.

After that, Sally continues to inspect the Gapminder data by changing the measurements to GDP and population. She wants to analyze the changes in the pattern between GDP and population in 1977 and 2007 (Please refer to Appendix. **??** for the visualization results.). In 1977, she found most countries had low populations and low GDP, as most of the points are located in an annotated compact distribution. However, by 2007, there were gaps in GDP between countries despite the populations mainly remaining low. There were two clusters of countries: the left cluster still kept low GDPs, while the right cluster had achieved much higher GDPs. China and India were two significant outliers in both 1977 and 2007, with significantly larger populations than other countries. Their GDPs had increased substantially over the 30 years, but they still had similar GDPs to the first cluster in 2007.

5.2 World Happiness 2020-2023

Sally conducts an analysis of the latest World Happiness reports [44] in the second case. The World Happiness reports are annual publications produced by the United Nations Sustainable Development Solutions Network. Sally's objective is to investigate the relationship between happiness and health.

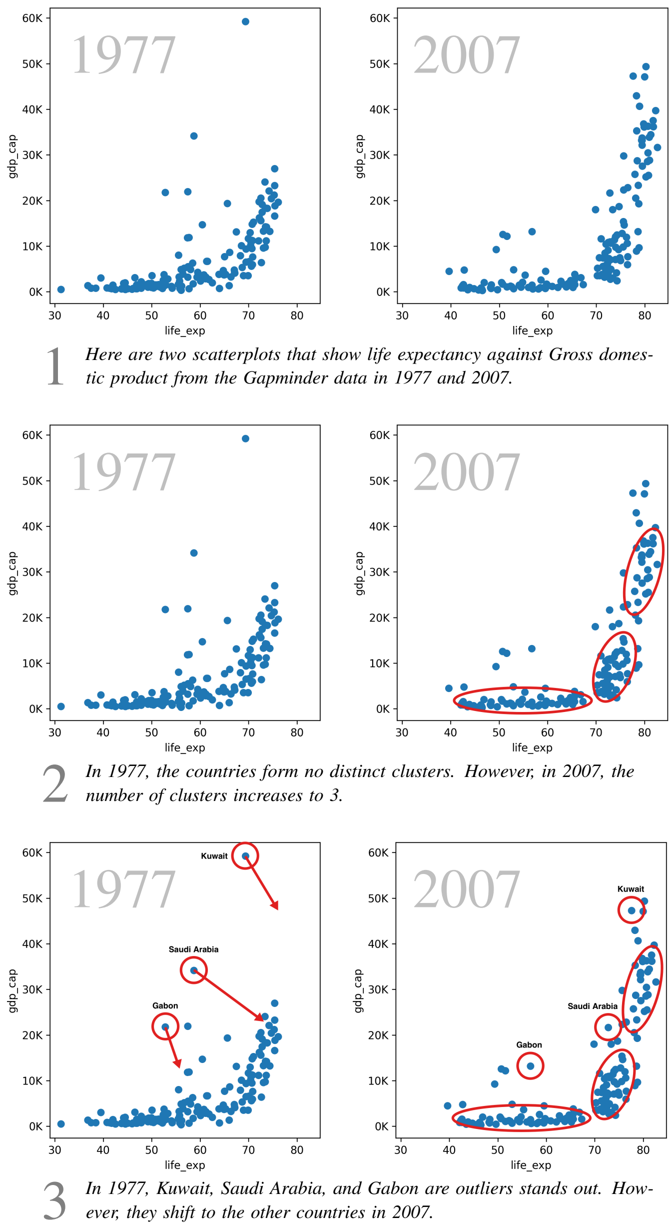


Fig. 6: These figures show automated sequential annotations on scatterplots from Gapminder 1977 and 2007 (Section 5.1).

She wants to explore the World Happiness Report in 2020 and 2023, which are two years at the start and end of COVID-19. The dataset consists of 146 rows, each representing a country with information such as its rank, region, happiness score, and other survey data (e.g., Freedom to make life choices, Healthy life expectancy, Perceptions of corruption, and so on). Sally creates two scatterplots by encoding happiness score and healthy life expectancy as x and y axes respectively, for the two years (the first panel of Figure 1). She then used the help of the automated annotator to provide some initial annotations.

In the second panel of Figure 1, the annotator draws two lines to demonstrate a strong correlation between health and happiness in both years. These annotations suggest that good overall life satisfaction is closely linked with good health, which can lead to a higher life expectancy.

Then, the annotator marks the birth of an outlier that occurred between the two years. As shown in the third panel of Figure 1, Lebanon's ranking in the World Happiness Report has experienced significant fluctuations. In the report of 2023, Lebanon's rank dropped to 145th out of 146 countries from its previous rank of 91st in 2020. Lebanon is annotated in both 2020 and 2023 to help with tracking the birth of an outlier in the data. Sally, in her investigation of the country, discovered several factors that contributed to this decline, including political instability, economic crises, and social unrest. Lebanon has been grappling with a severe economic crisis in recent years, with soaring inflation, unemployment, and poverty rates, which have directly contributed to the lower happiness score.

Apart from Lebanon, the annotator has identified two other points as outliers, as shown in the fourth panel of Figure 1. The one on the right side is Hong Kong S.A.R. of China, which has a moderate happiness score but a high health score. The other marked point is Afghanistan, which ranks last in the report. Afghanistan has consistently remained among the lowest-ranking countries, owing to its ongoing conflict, political instability, poverty, and lack of access to basic services and infrastructure.

Given these results, Sally can add her preferred annotations to modify and improve the story, thanks to the sequential annotations. For instance, she highlights Finland, which was consistently ranked as the happiest country in both 2020 and 2023, even though it was not detected as the automated insights. She marks this point in both timeslices at the beginning. The following automated annotations will not include a correlation line that does not follow a general-to-specific order, but will directly annotate the change of outliers. As the previous annotations could affect future decisions, the RL-based approach shows its advantage of adapting the user's preferences.

5.3 Summary

To sum up, the automated annotation approach can significantly reduce Sally's effort in understanding data and assist her in interpreting graphs. The annotator provides sequential annotations and Sally records valuable results and related information about the annotated chart. Then, she proceeds to inspect other parts of the data by changing the chart. This procedure continues in a similar fashion. With the help of the annotator, Sally can improve her workflow efficiency and quickly focus on noteworthy information.

6 User Study

We conducted a user study to evaluate the effectiveness of our approach, comparing the sequential annotations generated with our method against those created by a human and two baseline approaches.

6.1 Study

Participants We recruited 60 participants (31 females) aged 19 to 55 years ($M = 27$, $SD = 6$) via the Prolific platform. Only participants with normal or corrected-to-normal vision were involved. After our introduction, all participants confirmed their understanding of the scatterplot annotation methods. Each participant received two pounds in compensation, and the study took an average of 15 minutes ($SD = 7$) to complete.

Study Setup To prepare for the study, we used real-world datasets [1] to generate ten scatterplots as study resources. A data analyst, who has three years of experience in visualization and data analysis, was invited to manually design the sequential annotations for each scatterplot as the human baseline. Additionally, we designed two baseline approaches: 1) A naive policy that randomly adds possible annotations regardless of the significance of the insight, and 2) A rule-based approach that uses an insight-driven policy to identify significant data insights, but randomly determines the order in which annotations are added. In each of the ten study trials, we presented participants with a scatterplot followed by four sequential annotations in animated GIF format. The order of these four sequences was shuffled. The GIFs displayed annotations one after the other.

Procedure In the study, participants were introduced to annotation methods through simple examples before starting the experiment. Then, they should confirm whether they understood these concepts or not. Once they were ready, they began to complete ten trials. They then completed ten trials, rating each animated GIF on its informative and

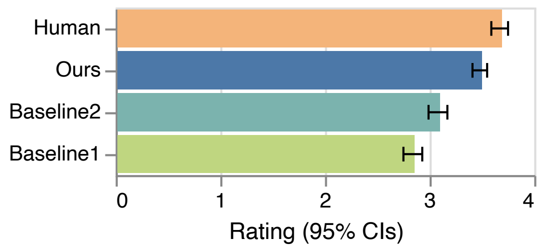


Fig. 7: Quantitative comparison of the mean ratings and 95% confidence intervals (left) among four strategies and (right) between annotations with and without captions.

clear presentation using a 5-point Likert scale. After each trial, they were asked to describe what features made some sequential annotations better than others. In the end, they completed a demographic survey.

6.2 Results and Analysis

We analyzed a total of 2,400 ratings and 60 comments from the participants. The rating results of our study are presented in Figure 7. Overall, our policy and the human user generated annotations significantly higher average ratings with 95% confidence intervals than the two baseline approaches. The statistical results show a significant difference ($p < 0.05$) between ours and Baseline 2 (paired t-test: t=-8.87, pvalue=7.97e-18, df=599, effect size: 0.36), as well as ours and Baseline 1 (paired t-test: t=-13.47, p-value=2.57e-36, df=599, effect size: 0.55). Our policy are highly preferred by the participants over baseline approaches that only consider data insights in scatter annotations. The human expert performed slightly better than our approach, but there was no significant difference (paired t-test: t=-1.76, pvalue=0.08, df=599, effect size: -0.07). The study also revealed that Baseline 2, powered by the insight significance, was significantly better than Baseline 1 (paired t-test: t=4.62, p-value=4.73e-6, df=599, effect size: 0.19), which uses random policy, indicating that significant statistical insights were preferred by the viewers.

The participants' comments revealed the reasons behind their ratings. They tended to give higher scores to sequential annotations that had more understandable data information, such as "*My ratings were whether the animation helped me understand the data*" and "*the more features that were present on the animation the higher I tended to rate the animations.*" Additionally, they liked annotations that showed the visual patterns of the scatterplot. For instance, "*If the lines/circles/points helped to understand the pattern of the graph, then I rated them higher.*" Furthermore, some participants mentioned that visual presentation plays a vital role in their ratings. For example, one participant said, "*I care about the amount of annotations used. If a lot of them are used, it can become confusing, which results in a lower rating.*" Moreover, several participants mentioned the importance of display order. They preferred the general-to-specific order to understand the graphical annotations. One participant stated, "*I liked the animations that presented overall information before moving on to pointing out specific examples on the plot. It is natural and easy to follow and understand the patterns.*" Another participant said, "*I preferred to see the trend line come up first in an animation, before any arrows or circling. This gives more context to the point you might be highlighting.*"

In summary, the qualitative feedback we received was in line with the design implications we took into consideration that informed by our previous study. Our approach, which is guided by these design implications, achieved human-level annotations in rating quality and showed significant improvement compared to two baseline approaches.

7 Discussion and Future Works

Generalizability. The study presents a promising computational approach to annotate scatterplots for data presentation. However, the current study has its limitations in generalizability in terms of diversity and expertise, as it involves only a subset of student participants from the same university. Nonetheless, this study provides valuable preliminary evidence that can pave the way for more comprehensive evaluations in the future. It provides sufficient evidence to demonstrate the feasibility of the approach. We have included all the results of the formative study as supplemental material to support future related research. To further improve the approach, a larger-scale user study and professional annotated visualizations [32] are necessary for future research.

Another challenge to generalizability is the diversity of real-world datasets. The current approach relies on synthetic datasets and online public datasets for training. However, the distribution of the training data may limit the ability to apply the approach to more complex datasets that are outside the distribution of the training data. We conducted tests on complex synthetic datasets and found that our approach may overlook some key findings as the number of data points and insights increases. We presented a failure case in the Appendix. One solution to this issue is to train the model on a larger and more diverse training dataset. However, obtaining all types of data takes time and may not be feasible. Additionally, while tabular data is a common data type, researchers need to find ways to apply their approach to textual data, spatio-temporal data, time-series data in future works.

Telling Expressive Stories. The current method focuses on two specific timeslices in animated scatterplots, which is a simplified approach to analyzing more complex temporal trends. In the practice of data storytelling, it may be beneficial to include multiple timeslices to create a compelling narrative that covers temporal changes more thoroughly. To achieve this, authors may use different narrative structures to make the time-oriented data story more engaging [26]. To incorporate this style, the current approach can be extended to support multiple timeslices. This can be done by allowing users to sequence the order of timeslices and then annotate each pair of them using the current approach.

Integrating Large Language Models. As LLMs become more commonly used to summarize and navigate complex text sources, they have the potential to help users explore data through visualizations. It is beneficial because they can extract insights from data beyond specialized chart type-specific methods. This could lead to a general-purpose annotator that can be interacted with using natural language. Code Interpreter, a tool from ChatGPT, is an example that demonstrates potential for automatically analyzing data and could complement the annotation approach. Moreover, LLMs can be useful in providing common knowledge that may not be available in the dataset. For instance, everyone knows that Kuwait is located in the Middle East, but automated methods may not recognize this unless it is already in the dataset. Threfore, incorporating LLMs as a means of providing common knowledge in the future can significantly enhance the effectiveness of the approach, expanding its usage scenarios.

8 Conclusion

This paper introduces a computational approach that graphical annotations can be automatically generated for interpreting scatterplots. Through a formative study with twelve university students with experience in data analysis and visualization, we advance the body of knowledge on what constitutes scatterplot annotation as well as provide insights on the processes involved to add annotations on scatterplots. Our technical contribution lies in learning a annotator model allows observe essential information about the annotation process, including the data, the chart, and its past annotations, and then take actions to sequentially add annotations that help with effective communication. We evaluated the approach via an application case and a controlled user study. The results demonstrate that the approach can generate understandable and functional annotations, and it can be useful in real-world data application for general users.

Acknowledgments

This work was supported by the Research Council of Finland (flagship program: Finnish Center for Artificial Intelligence, FCAI, grants 328400, 345604, 341763; Human Automata, grant 328813;Subjective Functions, grant 357578).

REFERENCES

[1] Vega datasets. https://github.com/vega/vega-datasets, 2022. [Online; accessed 20-Mar-2023]. 8

[2] D. Abel. A theory of state abstraction for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 9876–9877, 2019. 5

[3] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pp. 111–117. IEEE, 2005. 3

[4] P. Baldi. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pp. 37–49. JMLR Workshop and Conference Proceedings, 2012. 7

[5] K. Börner, A. Bueckle, and M. Ginda. Data visualization literacy: Definitions, conceptual frameworks, exercises, and assessments. *Proceedings of the National Academy of Sciences*, 116(6):1857–1864, 2019. 1, 2

[6] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000. 6, 7

[7] C. Bryan, K.-L. Ma, and J. Woodring. Temporal summary images: An approach to narrative visualization via interactive annotation generation and placement. *IEEE transactions on visualization and computer graphics*, 23(1):511–520, 2016. 3

[8] R. Chang, C. Ziemkiewicz, T. M. Green, and W. Ribarsky. Defining insight for visual analytics. *IEEE Computer Graphics and Applications*, 29(2):14–17, 2009. 1

[9] Y. Chen, S. Barlowe, and J. Yang. Click2annotate: Automated insight externalization with rich semantics. In *2010 IEEE Symposium on Visual Analytics Science and Technology*, pp. 155–162. IEEE, 2010. 1

[10] E. K. Choe, B. Lee, et al. Characterizing visualization insights from quantified selfers' personal data presentations. *IEEE computer graphics and applications*, 35(4):28–37, 2015. 2, 3

[11] I. Cohen, Y. Huang, J. Chen, J. Benesty, J. Benesty, J. Chen, Y. Huang, and I. Cohen. Pearson correlation coefficient. *Noise reduction in speech processing*, pp. 1–4, 2009. 6, 7

[12] C. Demiralp, P. J. Haas, S. Parthasarathy, and T. Pedapati. Foresight: Recommending visual insights. *Proceedings of the VLDB Endowment*, 10(12), 2017. 1

[13] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, and U. M. Fayyad, eds., *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pp. 226–231. AAAI Press, 1996. 6, 7

[14] A. Fan, Y. Ma, M. Mancenido, and R. Maciejewski. Annotating line charts for addressing deception. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2022. 3

[15] T. Gao, J. R. Hullman, E. Adar, B. Hecht, and N. Diakopoulos. Newsviews: an automated pipeline for creating custom geovisualizations for news. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 3005–3014, 2014. 3

[16] Gapminder foundation. https://www.gapminder.org, 2023. [Online; accessed 20-March-2023]. 7

[17] J. Heer and B. Shneiderman. Interactive dynamics for visual analysis. *Communications of the ACM*, 55(4):45–54, 2012. 2

[18] J. Hullman, N. Diakopoulos, and E. Adar. Contextifier: automatic generation of annotated stock visualizations. In *Proceedings of the SIGCHI Conference on human factors in computing systems*, pp. 2707–2716, 2013. 3

[19] J. Hullman, S. Drucker, N. H. Riche, B. Lee, D. Fisher, and E. Adar. A deeper understanding of sequence in narrative visualization. *IEEE Transactions on visualization and computer graphics*, 19(12):2406–2415, 2013. 5, 6

[20] E. Kandogan. Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 73–82. IEEE, 2012. 2, 3

[21] Y. Kim, K. Wongsuphasawat, J. Hullman, and J. Heer. Graphscape: A model for automated reasoning about visualization similarity and sequencing. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pp. 2628–2638, 2017. 5

[22] Y.-S. Kim, N. Henry Riche, B. Lee, M. Brehmer, M. Pahud, K. Hinckley, and J. Hullman. Inking your insights: Investigating digital externalization behaviors during data analysis. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces*, pp. 255–267, 2019. 1, 2

[23] H.-K. Kong, Z. Liu, and K. Karahalios. Internal and external visual cue preferences for visualizations in presentations. In *Computer Graphics Forum*, vol. 36, pp. 515–525. Wiley Online Library, 2017. 2

[24] N. Kong and M. Agrawala. Graphical overlays: Using layered elements to aid chart reading. *IEEE transactions on visualization and computer graphics*, 18(12):2631–2638, 2012. 2, 3

[25] C. Lai, Z. Lin, R. Jiang, Y. Han, C. Liu, and X. Yuan. Automatic annotation synchronizing with textual description for visualization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020. 3

[26] X. Lan, X. Xu, and N. Cao. Understanding narrative linearity for telling expressive time-oriented stories. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2021. 9

[27] Y. Li. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, 2017. 7

[28] J. Liang and M. L. Huang. Highlighting in information visualization: A survey. In *2010 14th International Conference Information Visualisation*, pp. 79–85. IEEE, 2010. 2

[29] Y. Ma, A. K. Tung, W. Wang, X. Gao, Z. Pan, and W. Chen. Scatternet: A deep subjective similarity model for visual analysis of scatterplots. *IEEE transactions on visualization and computer graphics*, 26(3):1562–1576, 2018. 7

[30] T. Munzner. *Visualization analysis and design*. CRC press, 2014. 1, 2

[31] M. D. Rahman, G. J. Quadri, and P. Rosen. A qualitative evaluation and taxonomy of bar chart annotations. *Poster Proceedings at IEEE VIS*, 2, 2022. 2, 3

[32] M. D. Rahman, G. J. Quadri, and P. Rosen. Exploring annotation strategies in professional visualizations: Insights from prominent us news portals. 2023. 9

[33] D. Ren, M. Brehmer, B. Lee, T. Höllerer, and E. K. Choe. Chartaccent: Annotation for data-driven storytelling. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 230–239. Ieee, 2017. 2, 3, 5

[34] H. Rosling. 200 countries 200 years 4 minutes. https://www.youtube.com/watch?v=jbkSRLYSojo, 2010. [Online; accessed Feb-2023]. 2

[35] A. Sarikaya and M. Gleicher. Scatterplots: Tasks, data, and designs. *IEEE transactions on visualization and computer graphics*, 24(1):402–412, 2017. 3, 7

[36] A. Satyanarayan and J. Heer. Authoring narrative visualizations with ellipsis. In *Computer Graphics Forum*, vol. 33, pp. 361–370. Wiley Online Library, 2014. 3

[37] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE transactions on visualization and computer graphics*, 16(6):1139–1148, 2010. 1, 2

[38] D. Shi, F. Sun, X. Xu, X. Lan, D. Gotz, and N. Cao. Autoclips: An automatic approach to video generation from data facts. In *Computer Graphics Forum*, vol. 40, pp. 495–505. Wiley Online Library, 2021. 3

[39] D. Shi, X. Xu, F. Sun, Y. Shi, and N. Cao. Calliope: Automatic visual data story generation from a spreadsheet. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):453–463, 2020. 3

[40] M. Shin, J. Kim, Y. Han, L. Xie, M. Whitelaw, B. C. Kwon, S. Ko, and N. Elmqvist. Roslingifier: Semi-automated storytelling for animated scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 3

[41] A. Srinivasan, S. M. Drucker, A. Endert, and J. Stasko. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE transactions on visualization and computer graphics*, 25(1):672–681, 2018. 3

[42] H. Subramonyam and E. Adar. Smartcues: a multitouch query approach for details-on-demand through dynamically computed overlays. *IEEE transactions on visualization and computer graphics*, 25(1):597–607, 2018. 3

[43] E. R. Tufte and P. R. Graves-Morris. *The visual display of quantitative information*, vol. 2. Graphics press Cheshire, CT, 1983. 6

[44] World happiness report 2023. https://worldhappiness.report/ed/2023, 2023. [Online; accessed 20-March-2023]. 7

[45] H. Wu, D. Shi, N. Chen, Y. Shi, Z. Jin, and N. Cao. Visact: a visualization design system based on semantic actions. *Journal of Visualization*, 23:339–352, 2020. 5