TU WIEN Informatics

Visualisierungsempfehlung bei der Visuelle Analyse der Abfolge von Sportereignissen

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Media and Human-Centered Computing

eingereicht von

Thibault Borde
Matrikelnummer 12302387

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof.in Mag.a rer.soc.oec. Dr.in rer.soc.oec. Silvia Miksch

Wien, 6. Juli 2025

Thibault Borde	Silvia Miksch

TU WIEN Informatics

Guidance and Visualization Recommendation in Visual Analysis of Sports Event Sequence

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Media and Human-Centered Computing

by

Thibault Borde
Registration Number 12302387

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof.in Mag.a rer.soc.oec. Dr.in rer.soc.oec. Silvia Miksch

Vienna, July 6, 2025

Thibault Borde	Silvia Miksch

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Thibault Borde

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel“ habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, haben ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT- Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 6. Juli 2025

Thibault Borde

Danksagung

Ich möchte mich persönlich bei meiner Betreuerin, Prof. Silvia Miksch, dafür bedanken, dass sie mich unterstützt und mich an diesem Thema arbeiten lässt. Ich bin auch sehr dankbar für alle, die an meinen Studien teilgenommen haben, sowohl Expert:innen als auch Benutzer:innen. Mein herzlicher Dank gilt meiner Heimathochschule, der École nationale supérieure des mines de Saint-Étienne, sowie der Europäischen Union und ihrem Erasmus-Programm für die Möglichkeit, ein Doppeldiplom an der Technischen Universität Wien zu machen. Schließlich möchte ich mich bei meinen Freunden, meinen Hunden und meiner Familie für ihre bedingungslose emotionale Unterstützung bedanken. Merci Beaucoup!

Acknowledgements

I want to personally thank my supervisor, Prof. Silvia Miksch, for supporting me and letting me work on this subject. I am also very grateful for everyone that participated in my studies, both experts and users alike. My dearest thanks to my home university École nationale supérieure des mines de Saint-Étienne and to the European Union and its Erasmus program for offering me the opportunity to do a double-diploma with the Technische Universität Wien. Lastly, I send my thanks to my friends, dogs, and family for their unconditional emotional support. Merci Beaucoup!

Kurzfassung

Viele Bereiche haben begonnen, enorme Mengen komplexer Daten zu sammeln und zu verarbeiten, um ihre sehr unterschiedlichen Prozesse zu verbessern. In vielen Fällen werden Ereignisdaten gesammelt und in Form von Sequenzen zusammengefügt. In den meisten komplexen Fällen können multivariate Ereignisse bis zu Hunderte von Attributen haben, und Sequenzen können Tausende von Ereignissen enthalten, was die Visualisierung solcher Daten zu einer echten Herausforderung macht. Frühere Forschungsbeiträge haben gezeigt, dass solche Visualisierungen möglich sind, aber es bleibt selbst für Expert:innen mühsam, diese Datensätze zu durchforsten, vor allem wenn man nicht weiß, worauf man achten muss. Basierend auf einer neuen Klassifizierung und Anpassung bestehender Visualisierungsansätze und Anleitungsstrategien, schlagen wir in dieser Arbeit eine Lösung vor, wie die Analyse multivariater Ereignisabläufe durch Anleitung und Visualisierungsempfehlungen verbessert werden kann. Dies geschieht insbesondere durch die Vorstellung von *VizREvent*, einer neu entwickelten Anwendung, die sich auf den Bereich der Analyse von Sportereignissen konzentriert. Schließlich zeigen wir durch eine Evaluierung mit Expert:innen und Nutzer:innn der Visual Analytics, dass unser Ansatz die Argumentations- und Fragebeantwortungsfähigkeiten der Expert:innen und Nutzer:innen unterstützt und auch Möglichkeiten für weitere Forschung eröffnet.

Abstract

Many domains have begun collecting and processing enormous amounts of complex data in order to improve their very different processes. In many cases, event data is collected and put together as sequences. For most complex cases, multivariate events may have up to hundreds of attributes, and sequences can contain thousands of events, making the visualization of such data a real challenge. Previous research contributions have shown that such visualizations are possible, but it still remains tedious, even for experts, to go through those datasets, especially when not knowing what to look for. Based on a new classification and adaptation of existing visualization approaches and guidance strategies, we propose in this thesis a solution for how to improve the analysis of multivariate event sequences with guidance and visualization recommendations. In particular, this will be done by presenting *VizREvent*, a newly created application focusing on the domain of sports event analysis. Lastly, through an evaluation with both Visual Analysis experts and users, we show that our approach assists experts' and users' reasoning and question-answering capabilities and also opens up opportunities for further research.

Contents

Kurzfassung **xi**

Abstract **xiii**

Contents **xv**

1 Introduction **1**
1.1 Background . . . 1
1.2 Motivation . . . 2
1.3 Research Questions . . . 3
1.4 Contributions . . . 3
1.5 Thesis' outline . . . 4

2 Literature Review: Establishing the Synergy between the Related Research Fields **5**
2.1 Related Work . . . 5
2.2 Methodology . . . 8
2.3 Classification . . . 9
2.4 Results . . . 15

3 VizREvent: Supporting Event Sequence Visual Analysis with Visualization Recommendation **31**
3.1 Motivation . . . 31
3.2 User Interface & Interactions . . . 31
3.3 Implementation & Architecture . . . 37
3.4 User Scenario Example . . . 42

4 Evaluation **45**
4.1 Overview . . . 45
4.2 Study Design . . . 47
4.3 Visual Analytics Expert Study . . . 48
4.4 Visual Analytics User Study . . . 49

5 Discussion **53**

5.1 On the Synergy between the Research Fields 53
5.2 On VizREvent and its Evaluation 53
5.3 Limitations and Future Research 54

6 Conclusion **57**

Overview of Generative AI Tools Used **59**

List of Figures **61**

List of Tables **63**

Acronyms **65**

Bibliography **67**

CHAPTER 1

Introduction

1.1 Background

Event Sequence (ES) data and its analysis can be found in many different domains. From social media [CCW+18, LLMB19], to manufacturing [SHC+20], to healthcare [vdLWvG+23], to computer systems [XWY+20], [NBJ+21] and many others. *Events* are discrete occurrences characterized by a *timestamp*, a *duration*, *attributes*, and a *label* (often referred to as the Event type). Events with a single attribute are referred to as *univariate*, while those with multiple attributes are called *multivariate* [vdLWvG+23, vdLdFvdEV23].

In the context of basketball, an Event could be a field goal. Its timestamp might represent the minute in the game when it occurred. Attributes associated with this event could include the player who scored it, the coordinates on the court where the shot was taken, and the team that made the field goal. An Event Sequence, or sequence of events, is the list of Events grouped around the same attribute value, which is called the *case* [vdLWvG+23, vdLdFvdEV23]. The list is also ordered around another attribute, or, as is often the case, chronologically. For example, the sales record of a specific car salesman can be considered an ES, with the salesman being the *case* of the sale Event type. Multiple techniques exist to visualize ES [GGJ+22, vdLdFvdEV23, YM24]. However, most of those techniques only allow for univariate visualizations [GGJ+22, vdLWvG+23]. Therefore, limiting the analysis process. Recently, improvements have been made in visually analyzing multivariate ES while maintaining the multivariate perspective (instead of switching to univariate perspectives like it was traditionally done) [vdLWvG+23]. However, managing and going through such complex data still remains tricky, especially when users are not experts, are not familiar with the system, or are not sure what to investigate [vdLWvG+23].

Guidance can be defined as the act of giving advice, information, or instructions to someone to provide help or orientation in the accomplishment of a task or goal (interpretation of

the definitions given in two dictionaries [Mer25, Enc25]). As *guidance* is a broad term, this is only one possible definition. In the context of Visual Analytics (VA), Ceneda et al. [CGM+17] describe Guidance as the resolution of a knowledge gap through a machine-assisted process during the use of an interactive visualization. Moreover, Collins et al. [CAS+18] describe the goals of Guidance to be to inform, to mitigate bias, to reduce cognitive load, to verify conclusions, for training, and finally for engagement. The help provided by ideal Guidance comes in different degrees that should adapt to the user. The adaptation is threefold. First, the help provided (or Guidance degree) needs to evolve according to the user's tasks and goals (e.g., orienting toward a possible path or prescribing a clear target) [CGM+17]. Second, the help needs to be provided in a timely manner, e.g., refreshing the view recommendation after parameters were changed by the user and not before or way after [CAS+18]. Lastly, while the traditional Guidance approach has the user adapt to the machine [SJB+20], mixed-initiative approaches where the machine also adapts to the users are possible and can lead to creating more findings and insights for the user [PMCEA+22].

1.2 Motivation

Categorizing and storing Event data is not considered an issue, but creating effective visualizations to analyze them is. First, Events, even in a specific domain, are very heterogeneous. For instance, two different sports or even categories in the same sport may have different rules, mechanics, or regulatory bodies, which leads to different definitions of Events in the logging process. Events can have low or high dimensionality; they might not have the same attributes, or some of the values could be empty. The time dimensionality also adds to its complexity, as some Events might represent single points in time while others could be intervals [GGJ+22, ZYG+23]. Furthermore, different representations of time, such as in a linear or cyclic way, can further complexify the analysis. Moreover, depending on the intent of the user, we might want to apply various different strategies and techniques even in the same application case [ZSCL25]. As visualizations are also used in critical fields, the time lost in this search process may have very serious and grave consequences. Moreover, relevant views that could have offered important insights to the expert might be missed because of this.

Guidance, and in particular, Visualization Recommendation (VizRec), could be a way to improve the search process. However, it is unclear how to do so efficiently in the context of multivariate ES data and relating to the task (or tasks) at hand. Especially given the fact that, in the literature focusing on Event Sequence Visual Analysis (ESVA), and to the best of our knowledge, the concept of *Guidance* (as established in [CGM+17] and [CAS+18]) is not used at all. The idea of guiding and helping users exists but is not named that way or at all, and the related concepts are not necessarily applied. For instance, similarity metrics are often used to derive non-temporal data attributes and to guide users toward interesting patterns or comparing sequences or Events [DPSS16, DPS+19, YNKJ+25].

To our knowledge, there is no VizRec system that exists for ES data. However, such

systems have been used and are being used in the visual analysis of other data types [WQM+17, MWN+19, QRD+22, ZWZ+24]. Integrating VizRec into the domain of ES data could prove interesting and a fruitful way to improve the search process. Because, as it did in their respective use cases, it could lower the overall complexity of ESVA.

1.3 Research Questions

Having introduced the global context as well as ESVA and Guidance, the main research question of this master thesis is:

> **RQ1:** How can Guidance help in the Visual Analysis of Multivariate Event Sequences?

This research question can be refined into multiple sub-research questions.

> **RQ1.1:** Which types of Guidance have been used in Visual Analytics of Event Sequence?
>
> **RQ1.2:** How can incorporating Visualization Recommendations into Event Sequence Visual Analysis help users in their analysis?
>
> **RQ1.3:** How can the Guidance degrees Orienting and Directing help in the Visual Analysis of Multivariate Event Sequences?
>
> **RQ1.4:** How can Guidance enhance Visual Analysis in Sports Event Sequence Analysis?

1.4 Contributions

To answer the research questions, the thesis will attempt to contribute in the following ways:

- As there are no apparent overlapping contributions between the ESVA, Guidance, and VizRec research fields, this thesis will first attempt to establish the synergy between those fields. Especially by the creation of a classification focusing on the types of Guidance used, the types of Visualizations handled, and the data types supported (chapter 2).
- This thesis highlights the attempt at integrating VizRec into ESVA. That resulted in the creation of the *VizREvent* web application [Bor25], which will also be described by this thesis (chapter 3).
- This thesis will outline the evaluation of *VizREvent* and its approach at integrating VizRec in ESVA through a two-part user study (chapter 4).

1.5 Thesis' outline

This thesis will first establish, in chapter 2, the synergy between the ESVA, Guidance, and VizRec research fields by first presenting related surveys (section 2.1). We then introduce the methodology used in this chapter (section 2.2) as well as the classification used to categorize existing Guidance systems (section 2.3). The classification builds upon taxonomies of previous works and is composed of three main dimensions: **Guidance**, **Visualization** and **Data Analysis**. We then present our findings (section 2.4) of this classification process.

In chapter 3, we first introduce the *VizREvent* application by giving the motivation behind the creation of this application (section 3.1). Second, we highlight its User Interface (UI) as well as list the available user interactions and how they can help users in their analysis process (section 3.2). Then we describe implementation, the architecture, and overall the technology behind *VizREvent* (section 3.3). Lastly, we present a user scenario to highlight how the interactions come into play and how they help users (section 3.4).

Afterward, we focus on how we evaluated the application by using an established two-part survey from previous work in chapter 4. There, we first introduce the methodology and study design (section 4.1 and section 4.2). Then we describe the results of both surveys in section 4.4 and section 4.3.

Lastly, we summarize our overall findings as well as the limitations of this thesis and potential future directions in chapter 5.

With this thesis, we aim to help researchers by providing a bridge between the fields of Guidance, Visualization Recommendation & Event Sequence Visual Analysis.

CHAPTER 2

Literature Review: Establishing the Synergy between the Related Research Fields

2.1 Related Work

In this section, we will provide an overview of surveys and characterizations that are relevant with the topics of Guidance and ES visualization.

2.1.1 Guidance

Ceneda et al., [CGM+17], characterized Guidance along three main aspects. The **Knowledge Gap** corresponds to the information the user needs to make progress. The main goal of Guidance is to bridge this Knowledge Gap to help the user in their analysis. Then there is **Input and Output**. Inputs are the sources of information upon which the Guidance is built. Whereas, Outputs are the *answers* to the user's need and Knowledge Gap, as well as the *means* with which the user is presented the answer. Lastly, we have the **Guidance Degree**, which can be described as the level of interference by the system in the analysis process. There are three levels to this Guidance Degree, **Orienting**, **Directing**, and **Prescribing**.

Prescribing is the highest level of Guidance and is akin to an interactive presentation. An example of this level of Guidance would be the system giving a step-by-step list of instructions to the user to reach its goal. **Directing** is the medium level of Guidance. Directing Guidance, as its name suggests, directs the analysis by presenting the user with a set of alternative possibilities. Directing is strongly tied to recommendations [CGM19] and recommender systems. Finally, we have **Orienting**. Here, the user's mental map is maintained. Orienting relies on visual cues to propose points of interest to the users.

In contrast to directing, the cues presented while orienting are not ranked or ordered in any way. Orienting also relies more on the existing view by embedding itself into it and adding an extra layer of information [PMCEA+22].

The review by Ceneda et al., [CGM19] provides a classification of Guidance approaches. All approaches described in this survey are mixed-initiative approaches, and both dimensions, *System Guidance* and *User Guidance* are defined and then each divided into two categorization axes. The analysis of the system Guidance is divided into *analysis objective*, which corresponds to the goal of the Guidance process, and *Guidance degree*, reusing the characterization described previously [CGM+17]. Then, user Guidance is divided into how the Guidance is inferred from the user (*Guidance inference*) and whether the *Guidance*'s *direction* is toward the past (feedback is given from previous interactions) or the future (the user describes their goal for the next interaction).

Han and Schulz [HS23] expended on the notions of Guidance given by both Ceneda et al., [CGM+17], and by Collins et al., [CAS+18], to produce guidelines through decision support for a generic Guidance design.

The *mixed-initiative* approach is a core concept in many VA applications, [CCI+15]. In such applications, the system can take the initiative when it determines that it can advance the user's analysis. Likewise, in this approach, both the user and the system teach each other and learn from each other in order to improve the analysis process [SJB+20]. This concept is an essential part of Guidance, as the system needs to adapt the degree of Guidance applied [CGM+17] as well as apply the Guidance at the right time [CAS+18, PMCEA+22]. Moreover, Pérez-Messina et al.[PMCEA+22] expanded on the taxonomy of Brehmer and Munzner [BM13]. They incorporated the system's tasks in a mixed-initiative environment and explored how the system can, with the right degree of Guidance, alter the user's current task or perspective.

2.1.2 Recommendation system

In the classification of Guidance degrees, recommendation systems are often offering *Directing* Guidance. In specific conditions, like when the recommendations are not ranked or in the case of overview summaries, recommendation systems are then offering *Orienting* Guidance [CGM19].

Kaur and Owonibi [KO17] divided recommendation systems into four categories based on the strategies by which they recommend visualizations. On the other hand, the survey by Zhu et al. [ZSJ+20], provides an overview of the diverse VizRec systems and categorizes them according to which techniques they use to produce the recommendation. Recommendations can be *Knowledge-based* (also called *Rule-based*), *Data-Driven* (or *Machine-Learning-based*) or *Hybrid* (a combination of both). They also classified the recommendation systems by: the data type they handle, the Machine Learning model they use (if applicable), whether or not the suggestions are ranked, the input they take as well as the output they produce, and finally the design space they handle.

Zeng et al. [ZMD+22] developed an evaluation-focused framework to better compare VizRec algorithms. The comparison is made on how they navigate the visualization design space (e.g., Depth First Search or Breadth First Search), on whether the focus is toward encoding, data query, or hybrid recommendation, and also on how the ranking is done.

Soni et al. [SdRBV24] produced a survey focused on recommendation systems that help users create dashboards. They classified their findings with 8 dimensions: system's origin (name, year, reference), visualization considered, suggestion method, utility functions, layout method, user feedback, User Interface (UI) and evaluation, and finally license. We took inspiration from this survey by extending a large part of their classification to create ours; see section 2.3.

2.1.3 Event Sequence visualization

Guo et al. [GGJ+22] defined a four-dimensional design space for ESVA. The dimensions are the following: *data scale*, *automated sequence analysis*, *visual representation*, and *interaction technique*. These dimensions are then extended by the introduction of five high-level analytical tasks: *summarization*, *prediction and recommendation*, *anomaly detection*, *comparison*, and *causality analysis*. It is important to note that in this context, 'recommendation' does not refer to suggesting visualizations that might be interesting or anything directly related to the visualization process. Instead, as discussed in Guo et al. [GGJ+22], recommendation seems to be more connected to actions that can be taken based on the analysis results, rather than guiding the user during the analysis.

Linden et al. [vdLdFvdEV23] extend this survey by Guo et al. [GGJ+22] by focusing their review on the high-level task of *comparison*. They provide a taxonomy that can be used to classify existing visualization approaches for ES comparison. In this paper, and also in a previous work [vdLWvG+23], they provide the formal definition of Events described here:

Events can defined as a tuple : $e_i = \{a_{i0}, \ldots, a_{ij}, \ldots, a_{im}, t_{is}, t_{ie}\}$, where a_{ij} is attribute m of the Event e_i with an attribute value $a_{ij} \in \Theta_j$. Θ_i is the set of all possible values of the i-th attribute in case i is categorical, or $a_{ij} \in \mathbb{R}$ in case attribute j is numerical. And t_{is}, t_{ie} are the start and end times of e_i. Events are grouped around one attribute (the *case* attribute) and ordered on another attribute (often chronological) to determine the sequences. The formal definition of a *case* is $c_u = \{e_0, \ldots, e_i, \ldots, e_n\}$ where $a_{0m} = \ldots = a_{im} = \ldots = a_{nm}$. Moreover, the sequence belonging to the *case* c_u has the following definition: $s_u = [e_0, \ldots, e_i, \ldots, e_n]$ where $t_{s_i} \leq t_{s_{i+1}}$.

Yeshchenko and Mendling [YM24] also expanded on the survey of Guo et al. [GGJ+22] by focusing on the *visual representation* dimension. They developed the *Event Sequence Visualization Framework* (ESeVis) to give a better understanding and perspectives of the fields of Information Visualization and Process Mining and also to open up new synergies for future research. The framework categorizes papers on the categories of the visualizations they used (the visual representation dimension from [GGJ+22]), but also

on how ES are represented (*instance-based* or *model-based*). We also took inspiration from this survey in the creation process of our methodology; see section 2.3.

A recent work by Zinat et al. [ZSCL25] presents a multi-level framework for ESVA. The framework consists of four levels, which are (from highest to lowest) *Objective*, *Intent*, *Strategy*, and *Technique*. *Objective* represents the high-level goal of the user, e.g., pattern exploration or cohort comparison. Then, to achieve these goals, users form high-level tasks, which are called *intent*, e.g., augment data, configure visualization. *Strategies* are then established to realize those intents. Finally, to accomplish a strategy, multiple *techniques* exist. The granularity offered by this framework allows a better understanding of the tasks in ESVA.

2.2 Methodology

2.2.1 Aims of the Literature Research

We took inspiration from Yeshchenko and Mendling [YM24] for our methodology. Indeed, as they described, both the *Information Visualization* and *Process Mining* (or ES analysis) research fields are similar in what they do but have developed independently of one another. Making the literature connecting both sparse. Here we are facing a similar issue since Guidance and process mining research fields don't seem to have synergy. Similarly to Yeshchenko and Mendling [YM24], this chapter aims at establishing synthesized coherence [GB97] through the categorization of contributions from those two fields. There is synthesized coherence between research fields when links between them can be drawn and lead to further research opportunities. In contrast, progressive coherence describes a cumulative effort by researchers to shape a unified, emerging field. Lastly, fields might develop contradictory conclusions about a topic, which leads to a noncoherence. As mentioned earlier, there doesn't appear to be progressive coherence between Guidance and ESVA, and it is also unclear whether there is noncoherence or synthesized coherence or not.

With this chapter, we aim at categorizing contributions from both fields and establishing synthesized coherence for future research. The main objective of this chapter is to solve **(RQ1.1)** (defined in section 1.3). To make this literature research's aims clearer, we can divide **(RQ1.1)** into multiple sub-questions:

RQ1.1.1: Which degrees of Guidance are applied in Visual Analytics ?

RQ1.1.2: For which visualization types (e.g., Chart-based, Sankey-based) is Guidance used in visual analysis?

RQ1.1.3: Which Event Sequence analysis tasks are best supported by Guidance?

RQ1.1.4: How does Guidance in Event Sequence differ from Guidance in Visualization Recommendation?

2.2.2 Literature Research

A total of 75 papers have been found and analyzed in the writing of this survey. Years range from 2015 to 2024. The search engines and libraries Google Scholar, IEEE, and ACM were used. The following keywords were searched: *Event Sequence*, *Guidance*, *Visual Analytics*, *Survey*, *Event data*, *Visualizations*, *Visualization Recommendation*, *Recommendation*, *Process Mining*. The modern assistive AI tool ResearchRabbit [SY24] was also used, but only for forward and backward searches. Moreover, papers that were included in previous surveys, such as Guo et al., 2022 [GGJ+22]; Linden et al., 2023 [vdLdFvdEV23]; Yeshchenko and Mendling, 2024 [YM24]; or Soni et al., 2024, [SdRBV24] were also considered. From the papers collected, we went through another filtering phase:

- Our primary exclusion criteria was the lack of Guidance or recommendation in a contribution. Meaning that all remaining systems have some sort of Guidance implemented.
- Then, the secondary exclusion criteria was the data type, in particular on implementations that don't handle ES. Systems that satisfied both criteria were kept.
- For comparison purposes and to have additional contributions, systems that only satisfied the main exclusion criteria (i.e., the presence of Guidance but no ESVA) were also kept.
- We also excluded papers that were published before 2015 (ten years old or older).

Out of the 75 papers collected, 59 were excluded and only 16 were kept.

Finally, all papers and their contributions were then categorized following the classification described in section 2.3.

2.3 Classification

To categorize works from both Information Visualization, Guidance and ES, we extend the classification proposed by Soni et al.[SdRBV24]. Indeed, their classification offers a good overview of recommendation systems and therefore of Guidance systems. It also offers a good insight into the methods used to provide such recommendations. The *License* dimension was excluded to not overload the classification, and while it is important, it falls outside the scope of this literature research. Moreover, we extended or modified some dimensions, namely *User Interface* and *Visualization Considered*; more details on that are in their respective parts below. We also added new dimensions to include the concept of Guidance, see the *Guidance Degree* and *Guidance Prominence* dimensions. Lastly, we also included a focus on *Data Input*, which was lacking and which also allows us to establish the link with ES data.

2.3.1 Metadata

Naturally, we included the papers' metadata in the classification. As well as the name and reference, we also added the year of the papers' publication. This is to give insight as to how recent the corresponding contribution is.

2.3.2 Guidance Dimension

Guidance Degree

This dimension was inspired by [CGM+17]. Even though recommendation or suggestion systems are often offering *Directing* Guidance, when the suggested list or items are not ordered or ranked, then it is considered *Orienting* Guidance [CGM19]. This distinction was not really marked in Soni et al., [SdRBV24]. Moreover, the presence of *Prescribing* Guidance is also something that we need to look out for, as it is the most disruptive to the user's analysis and thought process [PMCEA+22]. Both of these facts justify the inclusion of this dimension. The degrees of Guidance are:

- **Prescribing (P)**: Given the current data and tasks of the user, the system computes the most efficient way for the user to reach the end of the analysis process. The result is then displayed to the user.
- **Directing (D)**: Compared to *Prescribing* Guidance, suggestions made here are focused on *directing* the user on multiple possibly interesting paths or targets. The suggestion can be of different quality and can be ranked according to relevant heuristics. *Directing* can be done by either focusing or constraining the research or, oppositely, by broadening the search (which is a rarer approach in practice) [PMCEA+22].
- **Orienting (O)**: this is the lowest degree of Guidance and the only non-disruptive one, in regard to the user's thought process. The goal here is to preserve the user's mental map as much as possible. Contrary to *Directing* Guidance, suggestions are not ranked or ordered.

Guidance Prominence

This dimension was inspired by Han and Schulz [HS23]. Indeed, they introduced another dimension to describe the way Guidance systems visually present answers to the user's Knowledge Gap. The different ways to present them are the following:

- **Implicit** Guidance: alternatives can be constrained or prioritized, e.g., by ordering attributes in lists or hiding unfavorable items. This Guidance is not expressed by additional visual encoding or elements.

- **Embedded** Guidance: visual encodings are used on top of the current view to provide assistance. Examples of encoding are color, animation, highlighting, or shape.
- **Expanded** Guidance: additional views are created for the sole purpose of providing information or further steps (i.e., providing Guidance). For example, the related views provided in Voyager 2 [WQM+17] qualify as expanded Guidance.

Suggestion Method

This dimension was taken from the classification of Soni et al. [SdRBV24]. Here, the way suggestions are produced is described. However, there are many facets to how it's produced.

Firstly, in the case of dashboard or multiple views visualizations, suggestions could take into account the current state of the global visualization. To this end, we differentiate systems on the following aspects:

- **Single and independent (SI)**: suggestions are made for a single visualization at a time, without considering the state of the dashboard.
- **Single and dependent (SiD)**: suggestions are still done for a single visualization at a time, this time with consideration of the overall state of the dashboard.
- **Global optimization (GO)**: a global suggestion of the whole dashboard is done at once. The different views suggested are automatically adapted to each other.

We then differentiate systems by how the suggested visualizations are built and which characteristics are taken into account in the visualization's generation process:

- **Design from User's Initial Actions (UIAs)**: the user is able to choose specific sets of the data upon which the system builds relevant visualization, often according to expert-defined heuristics or domain knowledge.
- **Visualization Query Language (VQL)**: the user defines visual constraints, and the data and objectives are turned into queries. A constraint solver is then used to produce a set of matching views.
- **Statistical properties of the dataset (SD)**; properties like variance, deviation, or distribution are used to rank and recommend visualizations.
- **Behavior-based (BB)**: inferred or direct user actions are collected, possibly from many users. This data is then used to output possibly relevant visualizations. This method typically relies on statistical properties or Machine Learning models.

Lastly, we can classify systems by whether they use Machine Learning or not in the computation of recommendations:

- **Machine Learning (ML)**: uses machine learning techniques such as decision trees, neural networks, or deep learning to compute recommendations.
- **Rule-based (RB)**: expert-defined rules and heuristics are used to compute recommendations.
- **Hybrid (H)**: both Machine Learning techniques and expert-defined heuristics are used at the same time to produce the recommendations.

We added *Rule-based* and *Hybrid* suggestions as they were not in the initial classification, or not by themselves at least. And it could prove insightful to directly see whether a system uses Rule-based or ML or both.

Feedback

This dimension is taken from Soni et al. [SdRBV24]. Systems can use user feedback to improve on the suggestions they make. Here, we distinguish between two main possibilities:

- **Direct feedback (DF)**: users can directly inform the system on whether the recommendations are satisfactory or not.
- **Behavioral feedback (BF)**: actions and choices made by the user are recorded. From this history, the system can infer ways to improve the suggestion it makes.
- **"none"**: the system does not support any kind of user feedback.

Utility function

This dimension is taken from Soni et al. [SdRBV24]. Options suggested by the system might be evaluated and eventually ranked based on a specific criterion. Thus, we consider the following cases for this dimension:

- **insights (IN)**: the system assigns a score or value to its suggestion based on criteria. For example, the system can follow visual constraints and assign a cost value to the suggestion related to how many constraints are not respected.
- **"none"**: when systems do not involve utility functions,
- **"Not Found" (NF)**: if it's not clear whether utility functions were used or not.

Layout

This dimension is taken from Soni et al. [SdRBV24]. Having the correct size for a visualization is important to ensure that the message it conveys is readable. Moreover, if multiple views are in place, the layout also has a significant role. Therefore, we differentiate the following cases:

- **Manual Size (MS)** or **Manual Layout (MLa)**: The user manually adjusts the size and placement of visualizations.
- **Imposed on a Grid (IG)** or **Imposed Size (IS)**: The system enforces a predefined grid layout or specific sizes for visualizations.
- **Optimized Layout (OL)** or **Optimized Size (OS)**: An optimization algorithm determines the size or layout of the visualizations.
- **Predefined Templates (PT)**: Users select templates designed to match the current dataset.
- **"none"**: the system does not support any kind of layout (e.g., a single view visualization).

User Interface

This dimension is expanded from [SdRBV24]. The User Interface (UI) plays a key role in empowering the analysis process. Therefore, we categorize features present in the contributions as:

- **drag and drop (DD)**: the control panels allow the user to DD filters to modify visualizations or the dashboard.
- **selection mechanisms (SM)**: includes selections of mark type, of different channels (e.g., colors, shape), and global and local filters. For a specific visualization or for the global dashboard.
- **Linking and brushing (LB)**: Supports the interaction between multiple visualizations by linking data points and highlighting related elements across views.
- **History and Saving (HS)**: the system offers undo/redo functionalities and/or screenshot or saving current progress.
- **Annotations (AN)**: users are able to annotate or highlight views and items that are of interest.
- **"Not Found" (NF)**: if it's not clear whether the system has specific features in its User Interface.

The features *Linking and brushing (LB)*, *History and Saving (HS)*, and *Annotations (AN)* were not present in the original classification by Soni et al. [SdRBV24] and have been added to display more interesting functionalities of the contributions.

2.3.3 Visualization Dimension

For the visualization dimension, we replaced all the standard and typical visualizations listed in Soni et al. [SdRBV24] by the list of visual representations presented in [GGJ+22] and extended in [YM24]. This choice was made for multiple reasons. Firstly, it improves the readability as there are fewer possible values and less text for cases that encompass a lot of visualization types. Secondly, some categories, like hierarchy-based or Sankey-based, were not included, while the original types listed could fit into the five categories of visual representation. Lastly, by doing that, we create a bridge between the Guidance research field and the ES field, since we can classify Guidance systems as being able to handle the visual representation used in ESVA. The visualizations considered are thus:

- From Yeshchenko and Mendling [YM24], **Matrix-based (MA)**, **Timeline-based (T)**, **Hierarchy-based (H)**, **Sankey-based(S)** and **Chart Based(C)**
- **mark type (MT)**: some systems allow users to choose geometrical primitives like squares, lines, and circles and then generate visualizations based on those.
- **many visualizations (MV)**: multiple visualizations are presented at once.
- we also mention the **total number of visualizations available in the system**. If we did not find this number of visualizations, we indicate it by **"Not Found" (NF)**.

2.3.4 Data Analysis Dimension

In this dimension, we focus on the data analysis supported by the contributions. As our focus is on ES data, so is the focus of this dimension. Be it from the sets of *input data types* and of *analysis tasks* present and supported by the contributions.

Input Data Type

We mainly want to see whether the contributions can already handle ES data, and whether it is univariate or multivariate. While time series data are not ES data per se, they can be modeled after time series [XYF+17, GGJ+22]. In the classification, we consider categorizing the following types of data:

- **Multivariate Event Sequence (MES)**: the system handles the visual analysis of multivariate ES data as defined in subsection 2.1.3

- **Univariate Event Sequence (UES)**: contrarily to MES, univariate sequence data only have one attribute, which is then the *case* and the *type*. They can be formally represented by $e = (\tau, t)$, with τ as the Event type and t as the timestamp.
- **Time Series (TS)**: this type of data differs from ES as, here, the data is captured in a continuous-time domain with regular intervals. Whereas, Events are discrete and can happen at any time [GGJ+22].
- **Other types of data (O)**: as our focus is on whether current contributions can handle ES, we group all other types of data into this category. Examples of other types of data are geospatial data, text data, images or videos, etc.
- If the type of data handled is not clear, we indicate it by **"Not Found" (NF)**.

Analysis Task supported

We want to observe which high-level tasks (or *objectives*) relevant to ESVA, as defined by [ZSCL25], are supported by the contributions. In other words, we want to see which contributions can help us in answering **(RQ1.1.3)**. These tasks are:

- **Anomaly Detection (AD)**: identifying patterns or events that differ from expected behaviors.
- **Stage Progression (SP)**: registering the progression of processes through stages or phases to discover paths, trends, or factors' influences over time.
- **Pattern Exploration (PE)**: exploratory analysis to identify common patterns or recurring sequences in Event data.
- **Prediction & Recommendation (PR)**: Forecasting future events through the modeling of past ES.
- **Cohort Comparison (CR)**: comparing Event, sequences, or cohorts between each other to generate insights.
- **Correlation & Causality(CC)**: Investigating potential causal links between variables or events.
- ***Not Found/Not Applicable* (NF/NA)**; for systems that do not give a clear idea which ESVA user task they enable or if they don't enable any.

2.4 Results

2.4.1 Overview

Voyager 2 [WQM+17] combines manual and automatic specification of view parameters to help in the process of VA. It introduces the concept of *Wildcards* specification that

enables the generation of multiple parallel charts. *Related Views* are also introduced, allowing the user to choose among a recommendation list which visual encoding they prefer for specific attributes.

Sequen-C [MSM+22] enables a multilevel and details-on-demand exploration of ES. Overviews can freely change according to the user's exploration direction. Moreover, to help users in their exploration, the system provides them with alternative optimal numbers of clusters to the default one. Enabling them to change their perspective.

EventThread [GXZ+18] utilizes tensor decomposition and a robust layout algorithm to create summarizations of large-scale and high-dimensional ES data. This, in turn, helps users in their latent thread pattern discovery as well as latent stage analysis; see Figure 2.1.

Multivision [WWZ+22] is a deep-learning-based, mixed-initiative recommendation system for designing and creating dashboards. The system leverages the user's behavior by recording it and logging it. This is then used to iteratively improve the deep learning model, making the recommendation more tailored to the given user.

Dowsing [ZWZ+24] provides user- and task-tailored VizRec for data exploration. To do that, it leverages ML and answer set programming to build a hybrid model for recommendation of views; see Figure 2.4.

EventAction [DPSS16] leverages ES analysis to support users in decision-making and in making action plans. The system enables this by efficiently presenting similar records and thus potentially appropriate actions. A more recent iteration of *EventAction* [DPS+19] improved on this by adding automatic recommendation of action plans.

Calliope [SXS+21] enables users to automatically produce a visual data story from tabular data. The system generates facts, which are a visualization and a caption pair, from the datasets and orders them by their information quantity and statistical significance. The user still has the control to edit, add, or remove facts to tailor the data story to their liking.

Data2Vis introduces information visualization as a *"sequence to sequence translation problem"*[DD18, p. 8]. Here, the visualization production process can be formulated as the answer to the following question: *How can this JSON data be converted to a Vega-Lite specification?* To answer that, Data2Vis trains a deep learning model to encompass this sequence-to-sequence specificity. The model will then generate a set of valid visualizations from a JSON input.

'Engaging Dashboards' by Aksu et al. [AdRORR19] presents an expert-defined decision model for creating Key Performance Indicators (KPI) dashboards.

DashBot [DCCS+21], see Figure 2.3, provides support in iteratively building dashboards (B) panel by panel. The system suggests one panel (A) at a time that can be accepted or denied with or without justification by the user. Recommendations adapt to the direct user feedback and also to the current state of the dashboard.

SpotLight [HRM+23] is an insight-focused recommendation system that automatically discovers insights and insight types from the dataset and ranks them. From this, SpotLight suggests a multiple-view visualization to give an overview of the most relevant insights in the dataset.

Draco [MWN+19] is a VQL that uses logic programming and a constraint solver to provide recommendations of visualizations (in the form of queries). It can then be integrated into other tools such as Vega-Lite[SMWH17] to then have a UI and be displayed. Users can write constraint specifications to add on top of basic rules to tailor the visualizations suggested by *Draco*.
VISID [YNKJ+25] enables users to navigate through multiple granularities to interactively explore and analyze interaction logs.
"Interactive Similarity Builder" - Building similarity measurement for multivariate ES is a complex, tedious, and domain-centered task. To improve this metric-building process, Xu et al. [XSZX22] introduced a VA system for building and evaluating similarity measurements. This building is done using blocks and is verified at three levels of granularity. This system also allows similarity-based exploration of multivariate ES.
SeqCausal is a causality analysis-focused interactive system for multivariate ESVA [JGC+21]. The system allows users to iteratively build a causal model in a mixed-initiative way.
CallFlow [NBJ+21] helps users identify performance bottlenecks in the context of calling context trees of application codes. The analysis is empowered through Sankey diagrams whose layout is optimized according to the current dataset; see Figure 2.2.

The classification of the contribution can be found in Table 2.1.

2.4.2 Guidance Dimension

Guidance Degree

Directing - From the classification, we can observe that *Directing* Guidance is the most common degree used. First, we can group some contributions together [WQM+17, MWN+19, HRM+23, ZWZ+24]. Indeed, all of them follow a similar principle as to what degree and how the Guidance is provided. Each of them has different computation processes, but they all produce a set of visualizations that are ranked according to quality metrics.
For instance, in *Draco* [MWN+19], this metric is the cost value of constraints broken. Recommendations of visualizations are then ordered on this *cost* to then be provided to the user.
In *EventThread* [GXZ+18], the data is transformed into latent threads through tensor decomposition as shown in Figure 2.1. Threads (a) provide a link between stages, events, and entities and, after being computed, are visualized through a line map. The threads are then divided into stages (b) and nodes (c) on the lines, which display the probability of events occurring at this stage. Each node on a line visually represents Events with a high probability of occurring at the specific stage. The node is labeled as the Event with the highest probability, while the other remaining Events are encoded in the nodes using circles. The circles are colored according to the Event type, and the size is relative to the probability of the Event at the given stage. The Events in a node (i.e., the circle visual marks) are ranked according to their probability. The gray background (h) behind

Table 2.1: Classification of the selected contributions.

System		Guidance Dimension						Visualization Dimension	Data Dimension	
Systems	**Year and Reference**	**Guidance Degree**	**Guidance Prominence**	**Suggestio Method**	**Feedbac**	**Utility Function**	**Layout**	**User Interface**	**Visualization Supported**	**Input Data Type**	**Tasks Supported**
Voyager 2	[WQM+17]	D	Expanded	VQL	None	IN	IS, IG, MS, MLa	DD, SM, HS, AN	C, MT, MV (NF)	O	NF/NA
Data2Vis	[DD18]	O	Expanded	ML, SD	None	NF	IS, IG	SM	C, MV (NF)	O	NF/NA
EventThread	[GXZ+18]	D	Embedded	SD, RB	None	None	OS, OL	SM, LB	MA, T, H, C, S, MV (6)	MES	SP, PE
Draco	[MWN+19]	D	Expanded	VQL, H	None	IN	None	None	MT (NF)	O	NF/NA
'Engaging Dashboards'	[AdRORR19]	P	Expanded	RB	None	None	IS, IG	SM	C, MV (13)	O	NF/NA
EventAction	[DPS+19]	D	Embedded	SD, H	None	IN	IS, IG	SM, LB	MA, H, C, MV (NF)	MES	PR, CR
MultiVision	[WWZ+22]	D, P	Expanded	BB, H, SiD, GO	BF	IN	IS, IG, MS, MLa	DD, SM, LB, HS	C, MV (5)	O	NF/NA
DashBot	[DCCS+21]	P	Expanded	BB, SD, ML, SiD	DF	IN	IS, IG	SM	C, MV (NF)	O	NF/NA
SeqCausal	[JGC+21]	D, P	Expanded	SD, ML	DF	IN	IS, IG	SM, HS	MA, H, C, MV (6)	MES	CR, CC
CallFlow	[NBJ+21]	O	Embedded	SD, RB	None	None	IS, IG, OL	SM	C, S, MV (3)	MES	AD, SP
Calliope	[SXS+21]	D, P	Expanded	SD, RB, GO	None	IN	IS, IG	SM, HS	C, MV (12)	O	NF/NA
Sequen-C	[MSM+22]	D	Implicit	SD, RB, GO	None	IN	IS, IG	SM	MA, T, H, C, MV (NF)	MES	PE, CR
'Interactive Similarity Builder'	[XSZX22]	O	Expanded	SD	None	None	IS, IG	DD, SM, HS	C, MV (NF)	MES	PE, CR
SpotLight	[HRM+23]	D	Expanded	SD, H, GO	None	IN	IS, IG	SM, HS	MA, C, MV (NF)	TS, O	NF/NA
Dowsing	[ZWZ+24]	D	Expanded	SD, BB, H, SiD	BF	IN	IG, MS, MLa	DD, SM, LB, HS	C, MV (NF)	O	NF/NA
VISID	[YNKJ+25]	O, D	Expanded	SD	None	IN	IS, IG	SM, LB	MA, C, T, MV (NF)	MES	CR

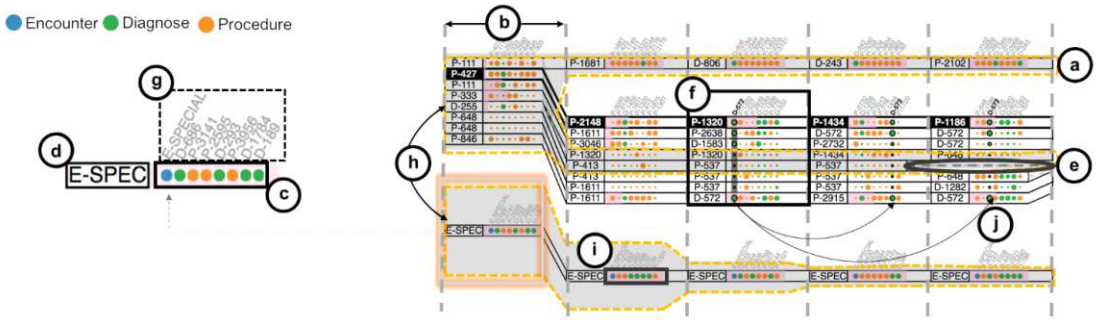

Figure 2.1: Thread Visualization in *EventThread*. Each node (c) that makes up the threads (a) displays the Events that are the most likely to happen. The Events are ranked inside a node, and the node is labeled by the most likely Event (d). Figure taken from [GXZ+18].

the nodes represents the number of Events involved in the current stage or category. All of this makes it a *Directing* Guidance on the system part. Moreover, not all Events in the node are displayed; only the highest Events are displayed, which reinforces the classification of *EventThread* as providing *Directing* Guidance. It should be noted that, by selecting a specific node, all Events belonging to it are then displayed using the same type of encoding.

EventAction [DPSS16, DPS+19] recommends action plans for the user, but, as discussed in subsection 2.1.3, recommendations here are not linked to the visual analysis process but rather suggestions for action as a result of the analysis process. The VA Guidance in *EventAction* lies in how the data is presented. When looking into a specific ES, e.g., the academic record of a specific student, all sequences with high similarity scores are displayed. This is to help the analyst in its decision-making process by easing up comparisons. The Guidance here is *directing*, as we only show sequences that are above a threshold (that can be modified by the user), and all sequences below that are not displayed in the similarity charts. Moreover, the sequences with the most similarity have an additional visual mark to make them stand out from the rest.
Sequen-C [MSM+22] allows users to switch between global overviews by changing the number of clusters considered. The default value for this is automatically computed by the system when the dataset is first imported. However, the system also computes different values of the number of clusters and ranks them according to quality metrics. To change the clustering, users can select a value in a dropdown list that contains all the values computed by the system, ranked according to their quality score. It is therefore also a *Directing Guidance*.

Directing & Prescribing - Some contributions offer both *Directing* and *Prescribing* Guidance. In *Multivision* [WWZ+22], users can get a recommendation list for single views that takes into account the current state of the dashboard. Alternatively, they can

ask the system to directly generate them a whole dashboard with a user-defined number of views. These are respectively *Directing* Guidance and *Prescribing* Guidance. The generation of the entire dashboard could be considered as being *Directing* because it is basically a list of recommended views, and the user can just keep one and dismiss the others. However, at a higher level of granularity, the system outputs what it considers as the best dashboard and doesn't provide alternatives, which is clearly *Prescribing*. This is a good example of how the Guidance degree can change relating to the perspective or task at hand.
Calliope [SXS+21] is similar to *Multivision* in this regard. The data story dashboard can be generated all at once by a simple click, making the dashboard generation process *Prescribing*. However, the user can also edit, add, and remove single views and, thus, can consider the views as single entities and also as *directions* to explore.
In *SeqCausal* [JGC+21], the user can modify at will the model proposed by the system. The related views change when the user modifies the model. However, the system also guarantees that the modified causal model still satisfies quality criteria. The model suggested at each iteration is the result of a *Prescribing* Guidance process. Meanwhile, the model is itself composed of possible causal relationships, meaning that the systems suggest the best potential relationships, and by doing so, is *Directing* the user.

Orienting - Then, we also have systems with *Orienting* Guidance. *Data2Vis* [DD18] is classified as such because the views that the system outputs are not ranked by any means. There is no score or value given to them, and thus there is no real ranking among the several view possibilities.
In their similarity builder, Xu et al. [XSZX22] give overviews of all attributes of the dataset, be it with distributions of the attributes, with dimensionality reduction techniques, or with an Instance View to provide instance-level validation. In the different views, no ranking is done to order the attributes in any way. Therefore, the user's exploration faces no disturbance from that while it is still empowered through this additional information.
Last, *CallFlow* [NBJ+21] provides *Orienting* Guidance. The main view of the system, see Figure 2.2, is a Sankey diagram (a) whose nodes each have an embedded miniature histogram summary of their distribution (e). This small and not cluttered chart provides quick and efficient insight into each node and helps the user decide which node could be of interest. This without disrupting the analysis process of the user.

Orienting & Directing - The Guidance aspect in *VISID* is linked with *Cohort Comparison* [YNKJ+25]. Indeed, one panel of the UI is dedicated to showing the user similarity between sequences. This is done with a combination of a heatmap and dendrogram to give a broad overview (*Orienting*). After the user has selected a sequence to analyze more in depth, the ranked list of similar sequences is displayed to help in the comparison (*Directing*).

Prescribing - Finally, we have the contributions that provide *Prescribing* Guidance.

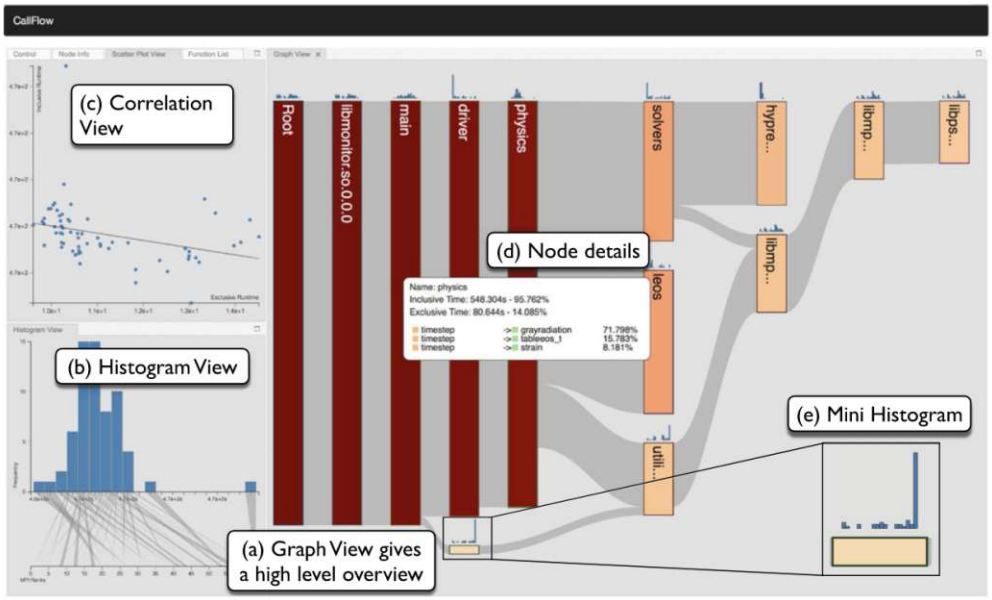

Figure 2.2: Screenshot of *CallFlow*'s interface. Calling contexts of parallel computer processes are visualized through a Sankey-Flow diagram (a). Related views (b, c) help the analyst by giving them an overview of the data. A tooltip (d) enables the user to gather more detail on specific nodes. Exploration of the data is empowered through the presence of mini histograms on each node (e), which enables a quick analysis of the variability. Figure taken from [NBJ+21].

The *'Engaging Dashboard'* by Aksu et al. [AdRORR19] automatically generates a full dashboard composed of multiple views of tabular data. The system does not output a single view one-by-one, so it cannot be used to explore visualizations as it can be done in *Calliope* [SXS+21] for example.
Dashbot's Guidance lies in the fact that the dashboard (B) is built one view, or panel, at a time [DCCS+21]; see Figure 2.3. The current view (A) is automatically generated and *prescribed* according to the given dataset and to the user's behavior history. However, the user can only accept or reject the current view suggestion in (C); there is no list of view suggestions to choose from, which reinforces the *Prescribing* assessment.

Guidance Prominence

Expended **-** Most contributions provide their Guidance by using an *expanded* view or panel, regardless of the degree to which it is applied. For instance, *Voyager 2* [WQM+17] has a view section called *Related View* where the recommendations of views that could be of interest to the user are displayed. All the systems that give out VizRec, like *Voyager 2* [WQM+17], *Draco* [MWN+19], *Dowsing* [ZWZ+24], *Calliope* [SXS+21], *Data2Vis* [DD18], *DashBot* [DCCS+21], *SpotLight* [HRM+23] and the contribution by Aksu et al. [AdRORR19], use expanded views to provide Guidance. For instance, the (A) panel in

Figure 2.3: *Dashbot*'s interface. Users iteratively build a dashboard in (B) by accepting, rejecting, or rejecting with explanations (C) the recommended panel in (A). Taken from [DCCS+21].

Dashbot (see Figure 2.3) or the (D) panel in *Dowsing* (see Figure 2.4).
On the other hand, *VISID* [YNKJ+25], *SeqCausal* [JGC+21] and the *'Interactive Similarity Builder'* [XSZX22] are also using additional panels and views to provide Guidance but are not providing VizRec.
VISID [YNKJ+25] and the *'Interactive Similarity Builder'* [XSZX22] both have a dedicated panel to give overviews of attributes or statistical data. For instance, in *VISID*, for a given ES, the most similar sequences are visualized in a comparison view as well as a similarity matrix to provide insight to the user and eventually direct them to promising sequences.
SeqCausal is a special case, as nearly every view can be used to provide Guidance. As described earlier, the causal model view, where the building of the causal model happens, provides both directing and prescribing Guidance. Moreover, the other views help by providing an overview of the process of building and verifying the causal model.

Embedded - *EventAction* [DPS+19] supports the user in their comparison analysis through colors (a, b & e) and visual marks (e). Indeed, sequences that are above a similarity threshold are colored to express that, and sequences that are an exact match or the ones with the highest similarity have an additional visual mark (e) to visualize that; see Figure 2.5. Likewise, *EventThread* [GXZ+18] embeds colored circles in their line map to provide users with an overview of which Events are the most likely to occur at a given node (c); see Figure 2.1. In the Sankey flow diagram of *CallFlow* [NBJ+21], small histograms are embedded on each node to give a quick overview of the distribution (e). Moreover, in the *histogram view* (b), an additional x-axis below the histogram is present, as well as *shadow lines* that connect the bins of the histogram to this axis; see

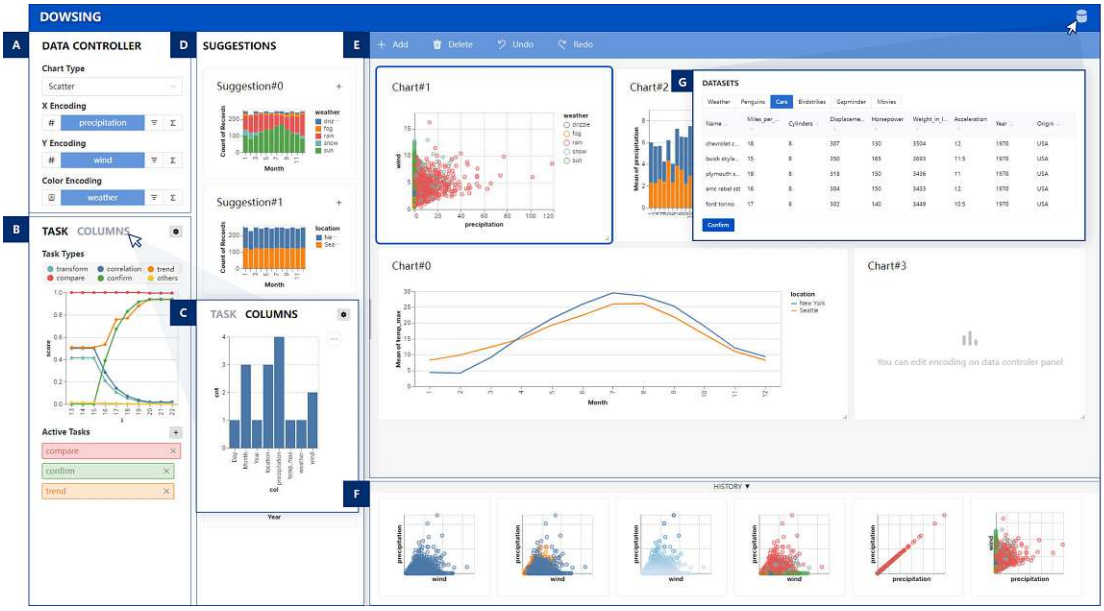

Figure 2.4: Interface of *Dowsing*. Users can build a dashboard on the canvas (E) by adding charts and selecting attributes from the Data Controller (A). Panels (B) and (C) allow users to check and adjust the system's interpretation of their tasks. VizRecs are provided in the *Expanded* window (D). Users can manually place and resize charts on the canvas (E), and check the modification history of the selected chart in (F). Finally, raw data can be analyzed in panel (G). Taken from [ZWZ+24].

Figure 2.2. This allows the user to see an additional distribution of the bins in the histogram, providing even more insights.

Implicit **-** In *Sequen-C* [MSM+22], the Guidance is implicit, as it can only be seen through the sorted dropdown cluster size list. It is therefore not visually encoded.

Suggestion method

Here, we can group the contributions by the computation method they use to produce the suggestions or Guidance.

Rule-based **-** Firstly, we have systems that use knowledge or expert-defined rules or heuristics to generate Guidance. The *'Engaging Dashboards'* by Aksu et al. [AdRORR19] uses an expert-defined decision model for automatic recommendation. Considering a KPI specification input, a decision tree is traversed until a leaf (or direct match) is found, which results in a visualization.
In *CallFlow* [NBJ+21], visualizations are produced using the statistical properties of the dataset (for the correlation view and the histograms). For the Sankey diagram, an optimization algorithm is used to minimize the clutter and overlapping edges.

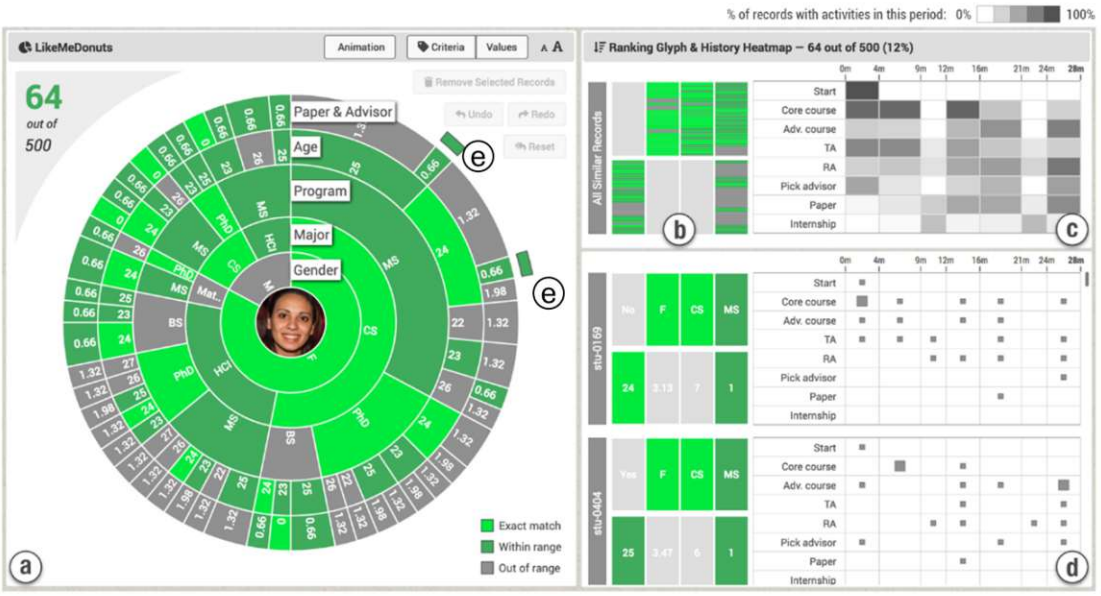

Figure 2.5: Advanced visualization components from *EventAction*. The *LikeMeDonut* (a) is a radial hierarchical treemap, where coloring represents records most similar to the currently selected one. Ranking Glyphs (b) provide an overview of records ranked by similarity. The heatmap in (c) shows the popularity of temporal events among the current similar records. Panel (d) displays more details on individual similar records. The most similar records are visually emphasized in (e). Adapted from [DPS+19].

To produce the threads it visualizes, *EventThread* [GXZ+18] first aligns the sequences before folding them in different stages. Once that is done, key features of the transformed data are captured and used to construct a tensor. The tensor is then decomposed, and threads are extracted from it.
Sequen-C [MSM+22] uses aggregate trees to produce and navigate through multilevel overviews of the dataset. To build the aggregate tree, a bottom-up aggregation approach is used. To represent the data clusters, they use what they called *'Align-Score-Simplify'*. Afterward, in order to find the optimal overview, they use the average silhouette width metric. From this, local maxima are found and constitute the ordered list of cluster size presented to the user.
Calliope [SXS+21] is the last contribution that is Rule-based. To generate the views, they first explore the dataset and look for data *facts*. The facts are then used to form a tree, which is then explored. A story is the path from root to leaf. Therefore, the exploration of the tree is done with an optimization method to produce the best story according to some metrics and scores. The optimization results also in an optimization of the layout of the views, as the order impacts the value of the story.

ML-based **-** Then there are the contributions whose system computes the Guidance

displayed using ML approaches. The sequence-to-sequence approach in *Data2Vis* [DD18] is based on an encoder-decoder architecture. A first encoder Recurrent Neural Network (RNN) takes a sequence as an input (e.g., a JSON dataset) and outputs a sequence of states. This sequence is then taken by another RNN (the decoder), which then outputs the visualization specification (e.g., a Vega-Lite [SMWH17] specification).
SeqCausal [JGC+21] uses Hawkes processes, which are ML-based, to model Granger causalities in the Event Sequences. A feedback mechanism is also implemented so that users can adapt the causal model as they want. However, the system also imposes a constraint so that whenever the user edits it, it still remains as optimized and valid as it can.
DashBot [DCCS+21] uses the statistical properties of the dataset, including entropy, variance, and coverage, to produce the most relevant panel. Visualizations already in the dashboard are taken into account in the selection. The user can either accept or reject the proposed panel. If accepted, the panel is added to the dashboard. If rejected with explanations, the specifications given by the user are used to create a new panel. When rejected without explanation, a reinforcement learning model is used to produce a new panel. Currently, both ϵ-*greedy* and *Softmax* algorithms are supported. The former being more efficient with a low number of panels, the latter being best with many panels.

Hybrid approaches **-** Some contributions presented both aspects of ML and RB. *MultiVision* [WWZ+22] uses deep learning models to produce the recommendation. The recommendation is not only given at a chart level, but also at a MV-level. The models are trained upon visualization corpus and provenance data (from logs). However, the deep learning models only output the chart types. For the visual encoding, heuristics are used.
In *EventAction* [DPS+19], the similarity between sequences is measured using a distance score. This scoring is based on weighted Euclidean distance. On the other hand, the sequence recommendation is based on Markov Decision Processes (MDPs). They chose to use MDPs instead of Recurrent Neural Networks or other deep learning approaches because model training requires much smaller datasets with MDPs.
The recommendation system of *Dowsing* [ZWZ+24] is divided into two parts. Firstly, we have the human-computer mixed agency, which infers the user's tasks and preferences. These inferred tasks can also be manually checked and defined by the user in the agency controller (B), see Figure 2.4. This is done with ML, and statistically derived attributes. For instance, Chi-square tests are used to determine if users have significant preferences. Then, answer set programming (which was derived from Draco [MWN+19]) is used to compute which visualization is best. The constraints here are on the data, the inferred tasks of the user (31 constraints are set for that in *Dowsing*), and finally there are soft constraints for the preferences as well.
Spotlight [HRM+23] uses heuristics, statistical properties, and both supervised and unsupervised training to first automatically discover insights in data. Then, using multiple learning-based methods, they rank those insights. From the rankings, visualizations are then produced and displayed.

VQL **-** Finally, we also have VQLs that are hard to group with others. In Draco [MWN+19], users submit and compile queries that contain the dataset they are using and constraints they wish to add on top of default ones. *Draco* then uses Answer Set Programming to produce a set of possible solutions, which are ranked on a cost score. Finally, the answers can then be translated into Vega-Lite [SMWH17] specifications. *Draco* uses expert-defined rules as constraints, but the weight of those rules is learned, making it a hybrid approach. In *Voyager 2* [WQM+17], the user specifies requirements in the control panel. From this is produced a *CompassQL* specification query, which produces the recommended visualizations.

Feedback

Behavioral Feedback **-** Overall, systems that use BF like *MultiVision* [WWZ+22] or *Dowsing* [ZWZ+24] use it to adapt their recommendation model to the user. In *Dowsing*, DF is also enabled, as the user can manually edit the current task being executed in panel (B) Figure 2.4. However, the model will then adapt to the user's input and behavior directly after. Moreover, the system infers by default the user's tasks. Thus making *Dowsing*'s Feedback more Behavioral than Direct. In practice, the main difference lies in how the model is used in the VizRec generation process.

Direct Feedback **-** Other systems use DF to iterate on their recommendation. *DashBot* [DCCS+21] proposes three different feedback mechanisms: yes, no with explanation, and no without explanation. The feedback is given by the user in panel (C) in Figure 2.3. It is interesting to note that ML is used in the case without explanations, and rules & heuristics are used in the case with explanations. Likewise, in *SeqCausal* [JGC+21], feedback is directly incorporated in the causal building process as it updates linked views and also because the ML model gets updated to accommodate this change.

Connection with Guidance Degree **-** We can observe that only directing and prescribing Guidance systems incorporate feedback. While it is hard to draw conclusions from that for BF, for DF it makes sense as asking the user for feedback can be disruptive, and therefore it seems hard to incorporate it in orienting Guidance.

Utility function

Because *Directing* and *Prescribing* Guidance need a way to compute which target or path is the best one, it is logical for them to all have a utility function. For instance, utility functions are used to compute the optimization of layouts, e.g., *EventThread* or *CallFlow*[GXZ+18, NBJ+21] or the optimization of recommendations, e.g., *Draco*. It should be noted that while the *'Engaging Dashboards'* [AdRORR19] is defined as prescribing, the fact that the recommendation output is produced through a direct match in a decision tree makes it an outlier. And while it recommends visualizations, it doesn't have any adaptation capacity whatsoever.

Layout

First, it should be noted that *Draco* [MWN+19], as a VQL, doesn't have an interface by itself. Implementations that use *Draco* might have different layouts and interfaces. For instance, Schmidt et al. [SPM24] implemented an interactive visualization of *Draco*'s recommendation system. In this system, users are able to input *Draco* queries, get the recommended view from the query, and have insights on how the recommendation was processed. *Dowsing* [ZWZ+24] also uses Draco in the background to compute VizRec but has a significantly different UI and goals compared to the system from Schmidt et al. In any case, intrinsically, *Draco* remains interface-less and has no layout.

Manual Layout **-** Manual editing of the layout is enabled by some systems. It is the case for Dowsing [ZWZ+24]. While there is still a hidden grid, its granularity is so small that no layouts are visibly enforced. Moreover, it also allows the user to change the size of visualizations manually; see Figure 2.4.

Layout Optimization **-** Other systems are optimizing layouts. *CallFlow* [NBJ+21], optimizes its Sankey diagram, (a) in Figure 2.2, to reduce clutter and edge overlaps. Likewise, *EventThreads* [GXZ+18] also optimizes its timeline chart to limit clutter.

Fixed Layout **-** All the other contributions impose the layout on the user. For example, in *EventAction* [DPS+19], the views cannot be edited; not that they are static, but the size, position, and layout overall cannot be changed. It is also the case in *'Engaging Dashboard'* [AdRORR19], *VISID* [YNKJ+25], *Calliope* [SXS+21], *Data2Vis* [DD18] and also in *Sequen-C* [MSM+22]. In *Voyager 2* [WQM+17], users can switch focus views and related views but cannot manually arrange them. In *DashBot* [DCCS+21], the layout is fixed, with the dashboard on one side and the panel recommendation on the other. The layout of the dashboard is a grid which visualizations are added one after the other; see Figure 2.3. *MultiVision* [WWZ+22] also has an imposed grid layout. However, the placement of the visualizations inside this grid is up to the user. *SpotLight* [HRM+23], orders its recommendations, but they are still displayed in a fixed grid layout that cannot be edited.

User Interface

Some systems provide minimal UI and interactions. Firstly, as mentioned before, *Draco* [MWN+19] doesn't intrinsically have an interface. Then systems like *'Engaging Dashboards'* [AdRORR19], *Data2Vis* [DD18], or *Sequen-C* [MSM+22] don't provide substantial UI.

Other systems have added functionalities that are useful, like tooltips for more information (*CallFlow* [NBJ+21], (d) in Figure 2.2), saving capabilities (*Voyager 2*, *Calliope*, *SpotLight*, *SeqCausal*, *'Interactive Similarity Builder'*), linking and brushing (*EventAction*, *EventThread*, *VISID*, *MultiVision*, *Dowsing*), or even a direct feedback interface (*DashBot*, (C) in Figure 2.3).

2.4.3 Visualization Dimension

Through our classification, we found out that multiple views are omnipresent in both VizRec and ESVA systems. Moreover, both Guidance Degree and Prominence do not seem to be limited to a specific type of visualizations. Likewise for both suggestion method and UI. The most present type of visualizations are chart-based graphs. Most recommendation systems are only limited to using those and in quantity (with MV) to express data.

Systems that handle ES ([GXZ+18, JGC+21, NBJ+21, XSZX22, MSM+22, YNKJ+25]) provide a good coverage of the different types of visualizations. All of them use Chart-based views, such as histograms, in *EventAction* for instance. Hierarchy-based views are also pretty common. For instance, *EventThread* uses a treemap to present an overview of the threads. Hierarchy charts are also present in *EventAction* with the *LikeMeDonuts*; see (a) in Figure 2.5 which can be described as a radial treemap. Matrix-based views are also used to provide overviews. In *SeqCausal*, users can use a matrix chart to compare the causal relations present in the current causality model. In *EventAction*, a heatmap is used to display the popularity of temporal events; see (c) in Figure 2.5.

Timeline-based graphs are used in *EventThread* to analyze the stage progression of events and to discover patterns; see Figure 2.1. In both *VISID* and *Sequen-C*, timelines are used to compare sequences between each other. Last, only *CallFlow* used Sankey-based graphs; see (a) Figure 2.2. This is done to represent a flow-based metaphor. It represents the resource being distributed from node to node in a process.

2.4.4 Data Dimension

Input Data Type

We have an even distribution between contributions that handle ES and contributions that handle other types of data. Interestingly, in the *other* category, most of the contributions only handle tabular data or spreadsheets ([WQM+17, AdRORR19, MWN+19, SXS+21, DCCS+21, WWZ+22, ZWZ+24]).
Data2Vis [DD18], utilizes JSON inputs. However, it isn't clear whether it could consist of tabular data, hierarchical data, or even ES. This could prove interesting if ES were supported, and it could be the aim of a future work. *SpotLight* [HRM+23], is the only system in the classification that could handle Time Series as they can visualize them and draw insight from them (TS outliers and correlation). Regardless, the systems that handled other types of data visualized them only with chart-based graphs and multiple views.

Systems that handle ES ([GXZ+18, JGC+21, NBJ+21, XSZX22, MSM+22, YNKJ+25]) all handle MES. In the classification, there are no contributions that only work with UES. It should be noted that all systems that are only supporting data other than MES, UES and TS all have Expanded Guidance.

Analysis Task supported

Since this classification was based on ESVA, and the tasks in this dimension are tailored especially to this data type. It is then normal that only contributions that handle ES are shown to support tasks listed in Zinat et al. [ZSCL25].

Anomaly Detection **-** Among the contributions, only *CallFlow* [NBJ+21] supports anomaly detection. In *CallFlow*, the overview histogram and its shadow lines ((b) in Figure 2.2) can visualize certain types of anomalies and can be used in that regard.

Stage Progression **-** SP is supported by *EventThread* [GXZ+18] and *CallFlow* [NBJ+21]. In *EventThread*, the thread view clearly symbolizes the progression of stages; see Figure 2.1. In fact, the nodes are aligned vertically by stage. Likewise, in the Sankey diagram of *CallFlow*, all supernodes represent a stage in the overall process; see Figure 2.2.

Pattern Exploration **-** PE is a well-supported task by the contributions in the classification. In *EvenThread* [GXZ+18], users can use the context-hierarchical-based views to inspect and compare patterns more efficiently. In *'Interactive Similarity Builder'* [XSZX22], users' main goal is to build efficient similarity metrics to evaluate their heuristics; multiple views are present to help them observe if the similarity metric matches sequences with similar patterns or not. Therefore, users have to explore the patterns themselves. Finally, *Sequen-C* [MSM+22] empowers users in their exploration of clusters and therefore of patterns, as clusters regroup sequences that share a set of Event types.

Prediction & Recommendation **-** In the classification, only *EventAction* enables PR. This is mainly done by showing records of similar ES through different views, e.g., *LikeMeDonut* ((a) in Figure 2.5) or similarity histograms or distributions.

Cohort Comparison **-** CC is the most supported task in this classification. In *SeqCausal* [JGC+21], cohort comparison is done in the context of causal comparison. In this case, we compare the effect of cohorts in the causal model being built. *EventAction* [DPSS16, DPS+19] enables comparison of sequences. Indeed, for a given sequence, the most similar sequences are listed in a view to enable comparison with the given sequence; see Figure 2.5. In *Sequen-C* [MSM+22], cluster comparison is supported through multiple views and by the fact that the systems allow for both multilevel vertical and horizontal overviews. In *'Interactive Similarity Builder'* [AdRORR19], users can and have to compare sequences to evaluate the similarity metric they are building.

Correlation & Causality **-** Last, we have CC. This is only supported by *SeqCausal* [JGC+21]. The system allows users to build in a mixed-initiative way a causal model optimized for the dataset. This is done through a multiple-view interface and iteratively with ML Guidance.

CHAPTER 3

VizREvent: Supporting Event Sequence Visual Analysis with Visualization Recommendation

3.1 Motivation

In chapter 2, through the classification and comparison with VizRec systems, we highlighted some possible areas of improvement for ESVA applications. First, all ESVA systems we examined have layouts that cannot be manually edited. While some do have some sort of layout optimization (e.g., *CallFlow* [NBJ+21], *EventThreads*[GXZ+18]), most have fixed layouts. This obviously limits the capabilities of the user to customize their interface, to use their expertise to build their own dashboards. Second, we have seen that while Guidance is present in all selected contributions, ESVA systems don't use VizRec as an approach to provide assistance.

These two shortcomings are the main motivation behind the creation of *VizREvent*. With this application, we aim to integrate both flexible layout management and VizRec into ESVA.

All the code can be found in our GitHub repository [Bor25].

3.2 User Interface & Interactions

In this section, we will describe the User Interface (UI), which you can find a picture of in Figure 3.1, and the interactions it enables. First, let us note that we had taken some inspiration from the *Dowsing* system [ZWZ+24]. Out of all the contributions we analyzed in chapter 2, we found that *Dowsing* was the most interesting one, with how it truly lets the user arrange their charts to create their dashboards. Moreover, in

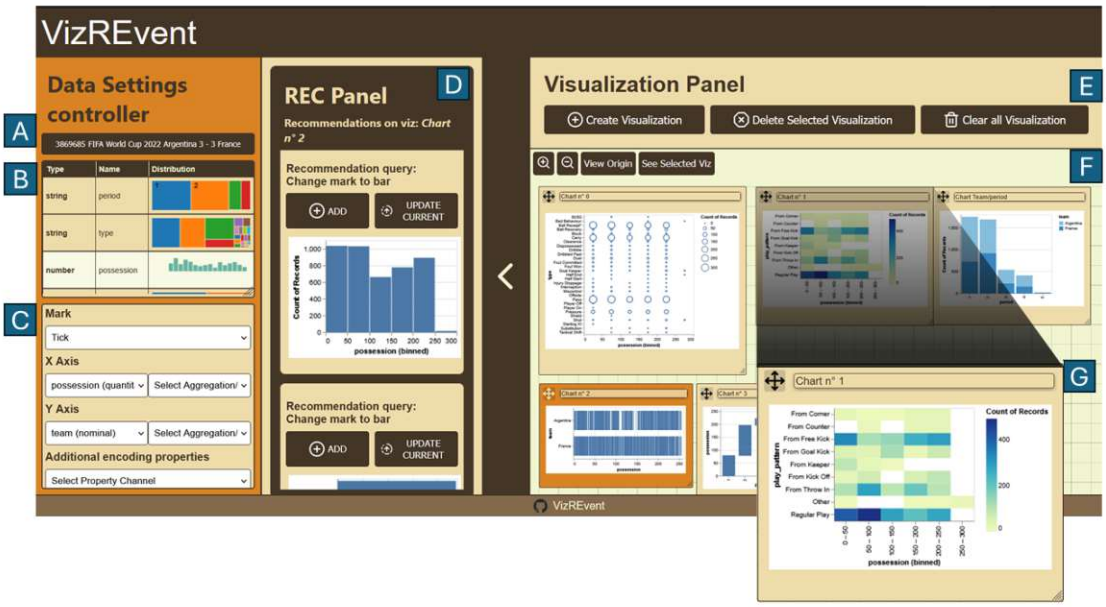

Figure 3.1: Interface of *VizREvent*. Extracted football datasets from [Sta21] can be selected for analysis in **(A)**. All available data fields are listed in **(B)**, with their names, data types, and distributions displayed. Visual encoding properties of the currently focused chart are shown in **(C)**. **(D)** is a collapsible panel that provides chart recommendations. Visualization instances **(G)** can be created or deleted using the Visualization Panel banner in **(E)**. All created Visualization instances **(G)** are displayed on the Canvas in **(F)**.

comparison to others, they also allow hiding the recommendations whenever they want, making it more *controllable*, which is something we should strive to achieve to make the Guidance more effective [CAA+20]. Besides that, we should also note here that the UI and interactions are obviously influenced by what is going on behind the scenes, how the recommendations are computed, the libraries that are used to create the visualizations, and also the datasets. For instance, we are using Draco 2 [YGZM23] to compute the VizRec and Vega-Lite [SMWH17] as specifications and for rendering. Therefore, we are constrained to their specifications and properties. This will be explained in more detail in section 3.3, as we focus here more on what the user will see, experience, and interact with.

VizREvent's UI is divided into three main part:

- The **Data Settings (DS) controller**, which contains **(A)**, **(B)**, and **(C)** in Figure 3.1. As its name implies, its role is to display information about the current state of the different data items (the dataset selected, its data fields, the visual encodings of the Visualization instance in focus), but also to modify them or select them.

Figure 3.2: Dataset Selection pop-up in *VizREvent*

- The **Recommendation Panel (REC)**, **(D)** in Figure 3.1, displays the recommendation of several charts (up to 10 charts at a time) for the user. It is mostly from here that Guidance is provided.
- The **Visualization Panel/View Panel (VP)**, which contains **(E)**, **(F)**, and **(G)** in Figure 3.1. It is here that all Visualizations instance, manually created or through the REC Panel, are displayed and where most of the analysis work by the user will be done.

Dataset selection (A)

In the current version of *VizREvent*, at the time of the writing, datasets from 13 different football games from the *FIFA World Cup* 2022 are available for analysis (more information about the datasets in section 3.3). When clicking on the **Dataset selection (A)**, a pop-up appears prompting the user to select a new dataset (see Figure 3.2). It should be noted that this interaction of selecting a dataset is the first step to do when using the system. If not done, then none of the other interactions are available (because there would not be any dataset to do interactions on).

Data Fields Display (B)

The Data Fields from the currently selected dataset that the user can analyze are displayed in **(B)**. Here for every field is displayed: their name, data type (as specified in Draco 2's documentations [YGZM23]), and value distributions (histograms for quantitative fields, treemaps for categorical and ordinal fields). The distribution visualizations are great at giving users an overview of the data fields with just a glance. The inspiration comes from how *CallFlow* provides *Orienting* Guidance by adding these small histograms on top of each node in the Sankey-Flow diagram (see Figure 2.2). In our approach, we don't

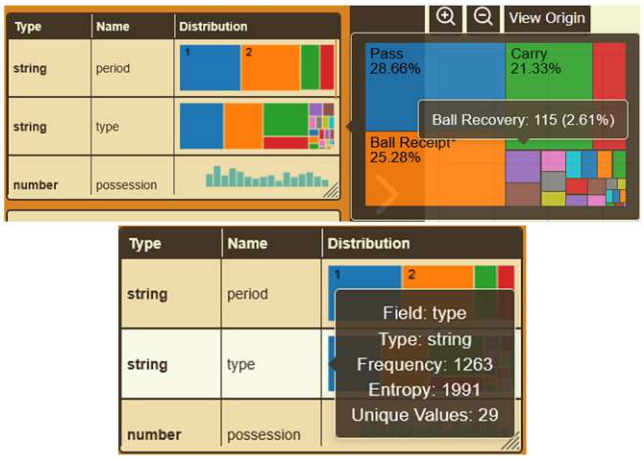

Figure 3.3: Tooltips in the Data Fields Display. The first tooltip displays a bigger version of the initial distribution visualization; in the treemap case, the names, frequency, and count of each value can be seen by hovering on the rectangle items. The second tooltip displays statistical properties of the hovered data field.

have Sankey-based visualizations (due to the limitations of current VizRec systems), but we still aim to provide a similar *Orienting* Guidance through also adding distribution visualizations. However, these charts are still quite small and don't necessarily explain the whole distribution (especially for treemaps, where the colors don't mean anything on their own without a legend or text to differentiate them). This is why tooltips (see Figure 3.3) were added to each data field item that magnifies the distribution visualizations when hovering on it (as well as allowing the user to see the values composing the treemap) and also gives statistical properties of the data fields. The statistical properties shown are *frequency*, *entropy*, and the count of *unique values*. Han and Schulz [HS23] highlight that such methods (visual data overviews/distributions and tooltips) can be considered as Guidance informing devices. And it is their role (here in particular) to *orient* the users. Lastly, this section also allows fields to be marked as interesting. Fields will be marked when they are already being used in the current selected Visualization instance in **(F)**. However, the user can also manually mark fields as being attractive if they so desire. This will result in them being highlighted when selecting a field in the Encoding Parameters Selection **(C)**. It also will impact the recommendation process, as some chart variants that include the selected fields will be generated (see section 3.3 for more detail). Thus allowing the user to make small adjustments to the recommendation output.

Visual Encoding Parameters Selection (C)

In *VizREvent*, we use Vega-Lite [SMWH17] to render and display the visualizations (except for the tooltips in **(B)**, where D3 [BOH11] is used). Therefore, the manual selection, as well as the display of the visual encoding of the currently selected Visualization instance, are following the structure and possible values of Vega-Lite. Moreover, we want users to

be able to manually create visualizations that could be recommended. This means that encoding values are filtered out if no recommendation will ever use them (e.g., Marks like *circle* or *geoshape* that are available in Vega-Lite but that are not handled by Draco 2 are filtered out of *VizREvent*).

If no Visualization instance is selected, then a placeholder text message is displayed, prompting the user to select an instance to get its encoding properties. Indeed, the properties shown in this block **(C)** are always the current visual encoding properties of the selected Visualization instance. It means that they will update when selecting a new instance, but also that the user can manually update the Visualization instance by selecting or changing the encoding values. As per the Vega-Lite specification [SMWH17], users can manually change the Mark (or primitive shape used to visualize the data, e.g., bars, lines, points), the data basis of the X and Y axes (by selecting a data field and/or performing binning or aggregations), and lastly by adding additional encoding properties like *color, shape, opacity*, or *size*.

Recommendation Panel (D)

As stated earlier, the main source of Guidance in *VizREvent* is provided here. The Recommendation Panel is by default closed on startup, but can be opened at any time by the user, thus allowing the user to turn on and off the Guidance (one of the effectiveness heuristics criterion in [CCEA+24]). The list of VizRec provided here are sorted according to their *cost* value in the Draco 2's recommendation pipeline. As the recommendation are ordered on a *cost* value, we provide *Directing* Guidance here. For performance and memory management reasons, we limit the number of recommendation to a maximum of ten.

The panel has a subtitle that indicates the query upon which the recommendations were computed. When no Visualization instance is currently selected, this subtitle will have the text *"General Recommendation"*, thus informing the user of the state and prompt of the recommendations. The VizRec provided will then be *generic* Draco 2 recommendation outputs, without any forced additional facts or constraints. These *generic* charts give quite an overview of the dataset and a good starting point for the user. However, they lack details and depth and are not sufficient enough for a complete analysis. Here, we aimed at providing something similar to what *Voyager 2* [WQM+17] does. When no encodings are chosen, *Voyager 2* also outputs overview charts to ease the user into the analysis and give them a starting point. With this, we aimed to do something similar.

When a Visualization instance is selected, then the subtitle changes to *"Recommendations on viz:"* followed by the name of the selected instance. In Figure 3.1, we can see that the subtitle indicates that the recommendations are influenced by "*Chart n°2*" which is currently selected (bottom left of the Canvas (F) in an orange coloring). The recommended chart items are also described in their title by the Creation Query with which this *Variant* (of the selected Visualization instance) was created. With this, we aim at giving the user more insight into how a specific recommended chart was created.

A similar experience happens when data fields are selected in **(B)**; the recommendation prompt is then extended with "*with selected fields:*" followed by the list of selected fields. Likewise, some of the recommended items may have been created from the specific selected field, and this is reflected in their title prompt as well. Note that the selection of a field can be done in both cases where a Visualization instance is selected or not.

Lastly, when the user has deemed one chart to be interesting for further analysis or modification, they can choose to add it to the Canvas **(F)**. In which case, a new Visualization instance **(G)** will be created. If an instance is already selected, then the user has the possibility to replace the chart in that Visualization instance by the recommended chart (button "*UPDATE CURRENT*" in Figure 3.1).

Visualization Panel Control Banner (E)

Adding a recommended item to the Canvas **(F)** is not the only option to create Visualization instance. Here, the user can manage the Canvas by creating new (empty) Visualization instances. They can also clean the Canvas by deleting the selected instance or by deleting all of them.

The Canvas (F)

The Canvas is where most of the analysis process will take place. Indeed, it is a very large (in comparison to the size of the Visualizations) square grid where all Visualizations instances are displayed and where users can build their own analysis dashboards. The user can move around the Canvas by panning; they can also zoom and de-zoom using the mouse wheel or the buttons. Additional buttons are provided to go to the Canvas origin (coordinates (0,0), top-left of the Canvas) and to go to the currently selected Visualization (if there is one). The *origin* is where all newly created Visualization instances are located. Visualizations can stack on top of each other, but their positions snap to the grid. This is done so to give a lot of creative liberty to the user but to still make it simpler to align items.

Visualization Instance (G)

The Visualization Instances are the blocks holding the charts created by the user. They can be selected by clicking on their center or the chart. They can be moved around (by dragging the top-left handle) in the Canvas or resized to the user's liking. Furthermore, they can also be renamed to names that make sense to the user; the customized name will still be displayed in the REC Panel **(D)** subtitle recommendation prompt. As mentioned earlier, a new instance can be created either through a recommended item in **(D)** or through the Visualization Panel Control Banner **(E)**. The chart inside one (selected) Visualization instance can be updated either manually through the Visual Encoding Parameters Selection **(C)** or through directly uploading a recommended item from **(D)**.

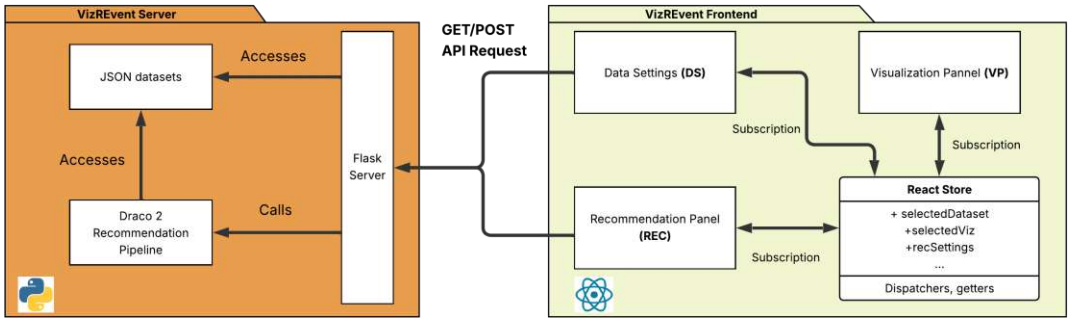

Figure 3.4: *VizREvent*'s global architecture. *VizREvent* follows a Frontend/Backend separation. The Backend or server, which is implemented in *Python* [Pyt24], handles the data and VizRec processing. The Frontend handles the UI and is what users will interact with.

3.3 Implementation & Architecture

3.3.1 Overview

VizREvent follows a Backend-Frontend separation of concerns. On the one hand, the server or Backend handles the database, the computations of the recommendations, and the data processing needed to do that. On the other hand, we have the Frontend which handles the UI and all user interactions. This separation was made for two main reasons. First, it helps with development and maintenance, as the parts are relatively independent from one another. Second, the datasets are quite heavy, and thus their processing is quite resource-intensive. It is better to do all that processing in a way that the user doesn't perceive it, so as to not disturb their analysis process, hence the server part.

The global architecture of *VizREvent* can be found in Figure 3.4. We will now give a bird's-eye view of the main parts of the App.

Draco 2

Before diving into the description of the Backend and the Frontend *VizREvent*, we will highlight how the VizRec are computed in the App. Here, we are using *Draco 2* [YGZM23]. As the names suggest, it is a newer and much improved version of *Draco* [MWN+19], which we described and analyzed in the chapter 2. The main improvements brought by the newer version mainly lie in the API implementations, as well as the dependencies used. Other improvements can be found in the documentation, development tooling, and debugging. However, both still work the same in terms of how they recommend visualizations. We will now mention *Draco 2* as just *Draco* for simplicity.

There are two main reasons why we chose to use Draco for the VizRec part of our tool. First, it is an established tool in the literature with extensive documentation

and great interoperability, and it can be easily reused in other projects. In contrast to most contributions, whose VizRec implementations were not necessarily intended to be reused in other projects. Second, newer approaches and recent research trends might lean toward using Machine Learning or other related technologies. However, in our case, since there are no VizRec systems especially targeted at ESVA, it would have meant that we would have needed to create our own system, train it, and then evaluate it. All of which falls way outside the scope of this thesis. Therefore, we chose to go with Draco in the implementation of VizREvent.

To produce VizRec, Draco uses its own visualization specification (inspired by Vega-Lite [SMWH17]). Through this, charts and datasets can be converted into *Facts*, which, with Answer Set Programming (ASP) and by using the Constraint Solver *CLingo*, can be used to produce recommendations. This is done by first inputting a partial specification of a chart, e.g., the data facts alongside the specifications of a simple two-dimensional chart (without any marks specified). Then Draco will complete the specification to build a full chart and, by using predetermined constraints to filter out bad charts (with hard constraints) and to sort the remaining chart by *cost* (with soft constraints). New constraints can be manually added by developers. Moreover, Draco models can use SVM (Support Vector Machine) to learn weights for the soft constraints. It should be noted that we did not create a specific Draco model for VizREvent using these methods. Indeed, our goal is to see whether current VizRec approaches (without extensive modification) are valid to provide Guidance in ESVA.

The Backend

The server was implemented in Python [Pyt24], mainly because Draco's main implementation is in Python. For the RESTful API and the route handling, we used Flask [Ron24] since it is easy to implement for smaller applications, and we are not expecting heavy traffic for the duration of the user study.

The main purposes of the server are:

- To serve as the database, and process the datasets to make it ready for displaying it in the UI.
- To produce additional information such as the list of data fields and their properties.
- To produce the VizRec according to the current selected items in the UI (or general recommendations if no Visualization instance or data field is selected).

For general recommendations, we only feed the most basic information and specifications that Draco needs as an initial incomplete specification.
However, to have recommendations that are adapted from the selected chart or data fields, we need to programmatically generate variations of incomplete Draco specifications in order to get variations of our current selected items. In the UI, we use Vega-Lite

specifications to display the charts, because Draco specifications can be converted to Vega-Lite ones quite easily. However, the other way around is much more difficult. Instead, we chose to parse the Vega-Lite specifications for the valuable parameters (e.g., data fields, marks, encodings, aggregates) and then build new incomplete Draco specifications from these. Here, we aimed at doing something similar to what [WQM+17] does, that is, displaying VizRec that are only slightly different from the one in focus. For instance, we build partial specifications while adding facts that eventually swap the mark to another one or that swap the axes.

Through this process, we generate several variations of partial specifications that we then complete with Draco. The completed specifications are then sorted on their cost, and only the ten cheapest charts are kept and sent sequentially to the Frontend to be displayed.

The Frontend

As mentioned earlier, the Frontend's role is constrained only to UI displays and user interactions. While there is some small data processing, it remains insignificant in comparison to the server, or it is only for rendering or interaction purposes. As we already described the interface and interactions in section 3.2, we will only focus here on the architecture of its implementation.

The Frontend and UI were developed with React [Met24]. This framework was chosen because it is very popular and has a lot of resources and open-source components, and it has a Vega-Lite library, which is really useful to display the Draco outputs since they can easily be converted to that specification format.

The Frontend's code follows the UI layout of the App (in Figure 3.1), and that can be seen in its architecture (in Figure 3.4). Indeed, all three main components of the app (**DS, VP and REC**) don't interact directly with one another. Instead, the global state management of the app is done through a *Store*. Here, all these components are subscribed to the Store (or at least to parts of it), meaning that whenever a Store property is updated, all components that are subscribed will get updated (and re-rendered) accordingly. These properties are updated when components that are subscribed *dispatch actions* to the Store to change the properties. This organization allows for multiple components to share common state attributes without being children/parents of one another, creating a separation of concerns. For instance, in the scenario described in Figure 3.5, in **(1)**, clicking the *ADD* button in REC dispatches this chart's specification to the Store on a pre-specified property. And because this property gets updated, VP is updated accordingly by creating a new Visualization instance with the specification from the Store.

Likewise, there is a separation of concerns in regard to the server's API. There is no need for VP to interact directly with the server, as all its user interactions (defined in section 3.2) don't need any processing from the server. Similarly to how all data-related API requests are only called by DS which then dispatches the important value to the

Store, REC is the only component that can send recommendation or Draco requests to the server. Also, this is done only when the panel is expanded, to optimize performance.

3.3.2 The Datasets

In this thesis, and through *VizREvent*, we attempted to integrate existing VizRec systems to ESVA. Knowing that, and to focus our research a bit more, we chose to focus on sports Events analysis. Indeed, there are many domains in which ESVA is used (as mentioned in chapter 1), leaving us with the freedom to pick which application domain to use. We decided to focus on sports Events, and more particularly football, for several reasons. First, and compared to other ESVA application domains, we felt that, with football being a mainstream sport and entertainment source, it would not require extensive domain knowledge for test participants in order to use the *VizREvent* App. Second, there is quite a large availability of open-source datasets related to sports analysis and sports Events, with often broader usage permissions. In comparison, if we had taken datasets from healthcare, the possible study participant pool would have been quite smaller, as it would have then required either participants with knowledge of this application domain or extensive descriptions of the data, which would then extend the duration of the study (and thus maybe refrain potential participants). It could also complexify the learning curve of *VizREvent*. Moreover, the availability and access to use are probably more restrictive due to the personal nature of the data.

The data that we analyze through *VizREvent* are JSON lists of football Events from a subset of the games from the 2022 *FIFA World Cup*. These datasets, as well as other football games and competitions datasets, are produced and published by *StatsBomb* in the following repository [Sta21].

Out of all 64 games from this tournament, we only kept 13. As most datasets represent about 3 MB of data, we didn't want the server to be unreasonably heavy while still allowing the freedom to choose different games to analyze. Besides those 13 football games Events datasets, we also used the related *Competition* dataset, which contains metadata about all the games from the 2022 *FIFA World Cup*. This specific dataset is only used in the dataset selection interaction; see the button's label of **(A)** in Figure 3.1 and all the data displayed in Figure 3.2.

The football games Events dataset each contains about 3000 Events, with the smallest having 2911 Events and the largest having 4707 Events. All of these Events don't exactly have all the same attributes. They all share some common metadata attributes; see Table 3.1. However, most possible attributes are only attached to certain Event types. For instance, Events with the type *pass* have the *recipient* attribute. But this *recipient* attribute is not used in any other Event type. Overall, there are more than a hundred possible attributes, but only about 10 can be found in every Event (metadata attributes). These datasets are therefore medium-sized in both the total number of Events and the number of attributes per Event [vdLdFvdEV23]. More extensive details can be found in the documentation of the Events datasets [Sta21].

Table 3.1: List of the data fields (and their descriptions) that can be analyzed with *VizREvent*. These data fields, and the descriptions written here, are extracted from football Event datasets from *StatsBomb* [Sta21].

Field	**Data Type (as per Draco 2 specification [YGZM23])**	**Description**	**Value Examples**
type	string	The name of the Event type.	*Shot*, *Half Start*, *Offside*, etc.
possession	number	Indicates the current unique possession in the game. A single possession denotes a period of play in which the ball is in play and controlled by one team.	1, 2, 3, etc.
possession_team	string	The name of the team that started this possession. This will appear even for opposition events like attempted tackles.	*Austria*, *Netherlands*
play_pattern	string	Name of the play pattern relevant to this event.	*Regular Play*, *From Corner*, *From Free Kick*, etc.
team	string	Name of the team this event relates to.	*Austria*, *Netherlands*
period	string (to preserve ordinality)	The part of the match in which the event took place.	1 = first half, 2 = second half, etc.

This heterogeneity poses a big issue, especially with the limitations of Draco. In order to compute the recommendations, Draco first needs to process *data facts* from a tabularized version of the dataset. Most importantly, this input table must not have empty values, which it will be full of in our case by flattening our JSON datasets. While replacing all empty values by default values (such as 0 for integers or empty strings) technically

fixes compilation errors, it produces very skewed recommendations and visualizations. This is mainly because most attributes are only present at most hundreds of times in the dataset, which makes them visually negligible compared to the number of default (or empty) values.

As we couldn't find a workaround for this issue, we decided to only keep the Events' metadata attributes, as they are present in every Events. It should be noted that we didn't include all available metadata attributes, as no real analysis can be done on some of them, or they cannot be adapted to our context (e.g., every Events has an *id* and a *timestamp* attribute; the former is useless to analyze, and the second uses a format not compatible with Draco). The descriptions of the attributes we kept can be found in Table 3.1.

3.4 User Scenario Example

In this section, we will present a typical scenario that users may face; see Figure 3.5. Here the user is trying to solve the following question (which is what we actually asked test participants to do; see chapter 4):

Which Event type is the most common, and which team did it the most?.

First, the user has to select a Dataset, for this example, let's choose the following: *3869685 | FIFA World Cup 2022 Final | Argentina 3 - 3 France | 2022-12-18* (see Figure 3.2).

In *VizREvent*, it is much faster to create new Visualization instances through the Recommendation Panel rather than manually from scratch. This is even more true when the user doesn't have an explicitly clear idea of what chart they want to see.

In **(1)** in Figure 3.5, the user has opened the Recommendation Panel right after the dataset finished loading. After scrolling through the recommendations, the user finds one that actually answers one of the questions they were looking for (the most common Event type). Therefore, they decide to add it to the Canvas. Alternatively, and for such simple questions, the user could have used the distribution Treemaps in the Data Field Display, **(B)** in Figure 3.1.

After adding the chart to the Canvas, the user notices that it is not correctly sized. Therefore, the user resizes it so that all values of the Y-axis are perceivable and distinguishable **(2)**.

However, this chart is not able to provide an answer for the other question that the user has. Therefore, the user looks once again at the recommendation (with the current selected chart) to see if the answer is provided there. Unfortunately, none of the recommendations provided satisfy the user. Thus, they decide to refine the recommendations by selecting the *team* data field (the recommendation prompt highlights the selected fields). Now, an interesting visualization has been suggested. And not wanting to replace the first chart created, the user decides to add this one to the Canvas **(3)**.

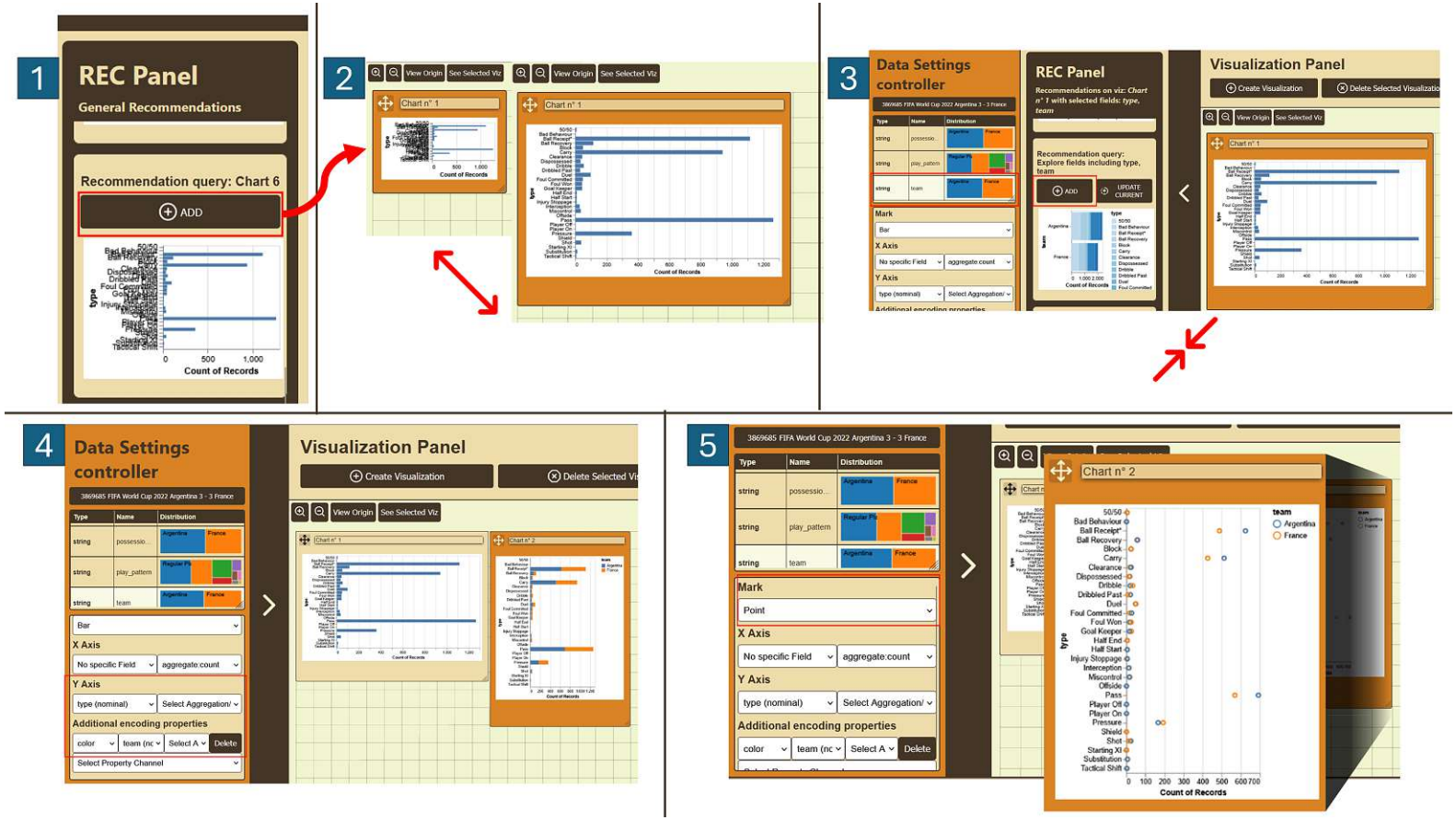

Figure 3.5: Typical User Scenario of *VizREvent*. Here the user first starts their analysis by opening up and checking the recommendation **(1)**. After they select one, they notice that it is not correctly sized; therefore, they resize it **(2)**. Afterward, the user looks at the updated recommendations but is not satisfied by them, so they select a data field in addition to the current Visualization instance. They add a new chart to the Canvas afterward **(3)**. The new chart is close to what the user is looking for, so they decide to manually change it. They also close the Recommendation Panel to not be disturbed by it **(4)**. Lastly, the user changes the mark to better compare the value count of each *team* for the same *Event type* **(5)**.

The newly added chart doesn't clearly provide the answer but provides a potential promising path to the answer, which the user decides to explore. After once again looking at the recommendations, the user decides to manually change the chart, as it is close to what could be interesting. So as to not be disturbed anymore, the user closes the Recommendation Panel. Then, they swap the fields on the Y-axis and on the color property to obtain a much more interesting chart **(4)**.

The answer here can be seen, but the bar's length difference of both teams' value on the *pass* Event type is too small to be distinguished with this mark. The user decides then to experiment with different marks. Eventually they end up on the *point* mark, which actually provides a clear answer to both questions the user was asking themselves **(5)**.

Through this user scenario, we have described how the different interactions come into play with each other but also with the user in a realistic situation. We also highlighted how the recommendation can prove to be useful when first starting the analysis or when switching up tasks. A much more experienced user or analyst may not have needed the recommendations as much (by looking directly at the distribution in **(1)** or by directly modifying the chart in **(3)** to find the answer, as the final chart is really close to it), but more amateurs or inexperienced users may not have the knowledge or the reflexes to do such manual transformations. They would, therefore, rely more on the recommendations.

CHAPTER 4

Evaluation

4.1 Overview

In this chapter, we will focus on evaluating the approach proposed in VizREvent. This assessment will be done by following the methodology and study protocol established by Ceneda et al. [Cen23, CCEA+24]. This paper describes a way to evaluate the effectiveness of Guidance approaches. We are following this method because we want to estimate the efficiency of our Guidance approach.

This study methodology is actually a two-part survey. Both are similar in the fact that participants of both studies evaluate the same system along eight *Effectiveness* criteria: *Flexible, Adaptative, Visible, Controllable, Explainable, Expressive, Timely,* and *Relevant*. However, they differ in both the recruitment of participants (especially on their expertise and skills), as well as the nature of the questions asked (see [Cen23] for the full list of questions).

One part focuses on Visual Analytics Experts (VAE), where participants need to have a serious background in VA, in designing visualization systems, and, quite importantly, having knowledge of what Guidance is and the terminology used in that field (as Ceneda et al. [CCEA+24] pointed out). Here, the design approach of the system is evaluated by having the participants rate the violation or application of design principles. VAE have to rate a total of 18 heuristics in the form of design principles on a Likert scale, from *clearly violated* to *clearly applied*.

On the other hand, the other part required Visual Analytics Users (VAU) as participants. Here, the only requirement is to be able to use a VA system. In contrast with the VAE part of the study, this part aims at evaluating the effectiveness of the system and especially of its guidance approach. VAU participants have to rate 27 heuristics in the form of statements, of which they have to indicate on a Likert scale whether they agree with it or not, from *strongly disagree* to *strongly agree*.

Effectiveness Criteria	Heuristics for Visual Analytics Users (VAUs)	Heuristics for Visual Analytics Experts (VAEs)
Flexible *the guidance degree is adequate and adapts to the situation*	**UH1** The guidance I received matched my need for support to resolve the problems I encountered.	**EH1** Make the guidance degree dynamic.
Adaptive *the guidance system considers user's preferences, habits, and the current task requirements for generating suggestions*	**UH2** When my task and goals changed, so did the guidance I received. **UH3** I felt the system was considering my input (e.g., previously selected suggestions/interactions) and/or personal preferences when producing the guidance.	**EH2** Use different sources of (user) input, e.g., selections and past interactions, to fine-tune the guidance.
Visible *the guidance and its status are easy to identify and well integrated with the rest of the visual environment*	**UH4** I could easily identify the guidance and distinguish it from the rest of the visual analytics environment. **UH5** The status and parameters of the guidance (e.g., if it is switched on, or if it is working, its parameters) were always visible on the screen.	**EH3** Integrate the guidance in the interactive visual environment. **EH4** Employ a meaningful visual schema to represent the status of the guidance and its parameters (show internal status).
Controllable *the guidance can be steered, allow the user to evaluate alternatives, switch on/off, revert the guidance*	**UH6** I could explicitly control the guidance by requesting or disabling it at any time. **UH7** The system allowed me to assess alternative suggestions and compare them. **UH8** I could directly provide feedback on the guidance I received. **UH9** I could fine-tune the guidance settings/parameters (e.g., change amount of suggestions, formulate queries).	**EH5** Allow requesting guidance on demand. **EH6** Allow switching the guidance off. **EH7** Facilitate assessing alternative suggestions (minimal memory load, minimal actions). **EH8** Allow direct feedback to the guidance suggestions (e.g., rating them). **EH9** Facilitate the specification/modification of the guidance parameters (e.g., specify queries).
Explainable *the guidance and the way it was generated can be easily understood*	**UH10** I understood what results the system was guiding me to. **UH11** The guidance suggestions were easy to understand. **UH12** I understood the reason why the suggestions were provided to me. **UH13** I could prompt the system for more details regarding the guidance. **UH14** I could trust the guidance.	**EH10** Display sufficient information to facilitate understanding of guidance. **EH11** Provide details-on-demand for explanations.
Expressive *the visual encoding of guidance is appropriate*	**UH15** The guidance suggestions used a language I could understand. **UH16** The guidance encoding is unambiguous (not open to more than one interpretation).	**EH12** Use the user's language to provide guidance. **EH13** Use an intuitive and consistent visual encoding for guidance. **EH14** Avoid misleading representations for the guidance.
Timely *the guidance is provided on time when needed*	**UH17** The guidance was provided at the right time when I needed it. **UH18** The guidance did not disturb my workflow.	**EH15** Deliver guidance in real-time, without delays for the user.
Relevant *the guidance helps prevent or recover from errors, guides towards a goal, helps discover the unexpected, provides multiple options and perspectives*	**UH19** The guidance helped me get out of a stalled analysis. **UH20** The guidance helped me solve my tasks accurately. **UH21** The guidance saved me time. **UH22** The guidance facilitated my reasoning. **UH23** The guidance helped me answer questions about the data. **UH24** The guidance was adequate for the task I was performing. **UH25** Through guidance I made serendipitous discoveries. **UH26** The guidance helped me generate new hypotheses. **UH27** I felt confident about the results obtained with guidance support.	**EH16** The system prompts guidance to recover from error states and stalled analyses. **EH17** The system provides guidance to warn of potential inconsistencies or errors. **EH18** Help explore all alternatives when several actions are possible depending on the context.

Table 4.1: The sets of heuristics used to evaluate the effectiveness of guidance. The columns shows the quality criteria (left), heuristics for VAUs (center), and VAEs (right). Adapted from [CCEA+24].

	Timeliness	Visibility	Expressiveness	Adaptability	Controllability	Relevance	Flexibility	Explainability
E1	7,0	7,0	7,0	7,0	5,2	6,0	6,0	5,0
E2	7,0	5,0	5,7	6,0	4,8	5,0	5,0	4,5
E3	7,0	5,0	4,7	4,0	6,0	5,0	5,0	3,5
E4	6,0	6,0	6,0	5,0	5,6	4,7	2,0	4,0
E5	6,0	5,0	4,7	5,0	4,2	4,3	5,0	3,5
E6	6,0	5,5	5,3	5,0	3,6	3,0	2,0	2,0
AVG	6,5	5,6	5,6	5,3	4,9	4,7	4,2	3,8

Figure 4.1: Results of the Expert part of the study. The results for each criterion are grouped, and the columns are reverse-ordered by average rating of the design dimension. The rows are also reverse-ordered by the average rating of the participant. The cells are bluer the closer their rating is to 7 ("Clearly Applied" heuristic). They are redder the closer their rating is to 1 ("Clearly Violated" heuristic). The white threshold, set up at 5, highlights a sufficient Guidance threshold.

The description of the effectiveness criteria and of both sets of heuristics can be found in Table 4.1

While the evaluation of both sets of heuristics is quantitative in nature (by the usage of Likert scales), Ceneda et al. [CCEA+24] suggest that the analysis of the results is more qualitative. Indeed, the heuristics are focused on qualitative criteria and could actually be phrased as questions to be asked in interviews. Moreover, for the VAE part of the study, the participants actually have to describe the observed violation or applications of the heuristic design principles. The analysis of the results will thus be qualitative.

4.2 Study Design

First, it should be noted that both studies followed the same design; only the questions asked were different. The studies took the shape of online surveys where, after first getting the consent of the participants, a 3-minute video tutorial introducing *VizREvent* and how it can be used was shown (a textual example tutorial with steps and instructions was also made available to the participants).

After finishing the tutorial, participants are given the link to access *VizREvent*, as well as two small tasks. The tasks are here to drive the analysis of the participants, to give them something to do so that they do not feel as overwhelmed as if we just asked them to interact with the system (like, free exploration). The quality of the answer is not judged, and this is made known to the user. The tasks took the form of questions such as the one we used to describe a typical user scenario in section 3.4. We also highlight that there is no time limit to this part, though we suggest they spend at least 5 minutes interacting with the system. Afterward, participants are asked to rate their relative heuristics.

4.3 Visual Analytics Expert Study

For this part of the study, we contacted 6 VAE by sending them the link to our survey, and all of them participated in the study. With this number of participants, we are above the recommended minimum number suggested by Ceneda et al., which is 3 [CCEA+24]. All of them are working in a Visualization research department and can therefore be considered as experts. The rating they gave can be found in Figure 4.1. Like in Ceneda et al., we grouped together the ratings obtained by design criteria for visual convenience. In the online questionnaire, the heuristics were not rated numerically, only with statements like "Applied" or "Somewhat Violated". The answers were then converted to numerical values, with the latter value examples being equal to 6 and 3, respectively. Overall, the system got a global average rating of 5.1, calculated across all responses provided by expert participants in the study. This score is slightly above the threshold (5) of sufficient Guidance set up by Ceneda et al. [CCEA+24]

Qualitative analysis

Flexibility - The staticity of the Guidance degree was observed by some VAE. (E6) denoted the lack of Prescribing Guidance, but they were unsure as to the presence of Orienting Guidance. (E4) pointed out that the Guidance degree was static. Meanwhile, (E2), (E3), and (E5) indicated how the Guidance output was changed according to the user inputs.

Adaptability - Here, the VAE positively observed how the VizRec was updating on the user interactions (by selecting a Visualization instance or a data field). They also remarked that the adaptability was only considering short-term interaction history. (E2) remarked that "current interactions and past ones (up to an extent)" were taken into account. While (E6) noted the history of interactions as being quite short or nonexistent.

Visibility - The guidance provided in VizREvent was perceived to be easy to distinguish from the rest of the app. However, the VAE pointed out that other internal statuses of the Guidance were missing from the UI, or that it wasn't clear if it was displayed or not (EH4).

Controllability - While the participants positively assessed the on-demand and the possibility to switch off the Guidance (by collapsing the panel). Some also negatively judged the recommendations on the lack of possibility to give direct feedback and to change the settings. For instance, (E6) found that the most interesting charts were often the ones recommended at the end, but they could not tell the system to show more of those. According to Draco, these were the ones with the highest *cost*. This doesn't mean that they are necessarily of bad quality; it could also mean that they are of a higher dimensionality and thus inherently violating more constraints.

Explainability - This criterion was the most badly rated, with an average of 3.8, which puts it in a negative average (as 4 is the middle "Neither Violated or Applied" value). Several experts could not understand how and why the recommendations were given.

	Timeliness	Expressiveness	Visibility	Relevance	Controllability	Adaptability	Flexibility	Explainability
U1	6,00	5,50	5,50	5,78	6,00	6,00	6,00	6,20
U2	6,00	6,00	6,00	5,89	6,00	5,50	7,00	5,40
U3	6,00	6,50	5,50	5,44	5,75	5,50	6,00	5,60
U4	7,00	6,50	6,50	5,44	5,00	5,50	5,00	5,20
U5	7,00	5,50	6,00	5,50	4,50	6,00	4,00	6,00
U6	6,00	5,50	5,00	6,13	5,33	6,50	6,00	4,00
U7	4,50	6,00	7,00	5,88	4,50	2,50	6,00	5,00
U8	6,50	5,50	5,00	4,89	5,00	5,50	5,00	4,00
U9	4,50	5,00	5,50	4,78	5,50	4,50	3,00	4,80
U10	5,50	6,50	5,00	5,44	4,75	4,00	5,00	3,00
U11	5,00	6,00	4,00	4,56	4,75	5,50	5,00	3,80
U12	4,50	5,50	6,00	4,00	4,25	3,50	3,00	4,80
U13	6,00	6,00	5,50	3,78	3,75	4,00	3,00	4,00
U14	5,00	3,50	6,00	3,44	5,25	5,50	3,00	4,00
U15	5,50	5,68	5,50	4,56	3,75	3,00	3,00	2,80
AVG	5,70	5,68	5,62	5,02	4,93	4,87	4,67	4,57

Figure 4.2: Results of the User Part of the Study. The results for each criterion are grouped, and the columns are reverse-ordered by average rating of the design dimension. The rows are also reverse-ordered by the average rating of the participant. The cells are bluer the closer their rating is to 7 ("Strongly Agree" heuristic). They are redder the closer their rating is to 1 ("Strongly Disagree" heuristic). The white threshold, set up at 5, highlights a sufficient Guidance threshold.

The fact that no details-on-demand could be provided was also highlighted by multiple experts.

Expressiveness - The experts globally rated this criterion positively. The layout of the app was judged to be quite simple to understand (E5). However, (E2) mentioned that the suggestions prompt could be ambiguous, as several items can have the same one.

Timeliness - This criterion was the highest rated. All VAE agreed that the system was fast in providing the recommendations and that, performance-wise, there were no issues.

Relevance - A lack of help to recover from error states was pointed out by the participants. (E1) mentioned that the system didn't seem to be able to warn of inconsistencies or errors. (E5) suggested adding other interactions, such as undo/redo, to provide help in recovering from errors. Lastly, (E6) pointed out that not all alternatives seemed to be shown to the user and that there weren't more than 10 recommendations at the time (which is true and explained in section 3.2).

4.4 Visual Analytics User Study

For this second part of the study, it is not necessary for participants to have extensive knowledge (or any knowledge of the theoretical concepts) about VA, Guidance, or information visualization. Or at least this is what is described by Ceneda et al. in their protocol [CCEA+24, Cen23]. The only requirement is that the study participants are able to use a VA system. While this is quite vague, we decided to only contact engineers or engineering students to participate in our study. Indeed, in this line of work, it is

probable the participants have either already used VA tools in their line of work/study, or they have enough skills to adapt quickly to these tools.

In the end we managed to have 15 participants, which is the minimum amount suggested by this methodology. Similarly to the study for the expert (section 4.3), the heuristics were not rated numerically, only with statements like "Agree" or "Somewhat Disagree". The answers were then converted to number with the latter value examples being equal to 6 and 3, respectively. The ratings obtained from the survey can be found in Figure 4.2. Once again, we grouped together the ratings obtained by design criteria. We also only encode the value with colors for visual convenience and to reduce cluttering.

Overall, the system got a global average of 5.04, calculated across all responses provided by participants in this part of the study. This is close to the average expert rating (see section 4.3) and only slightly above the threshold (5) of sufficient Guidance set up by Ceneda et al. [CCEA+24]. It should also be noted that we did not force the participants to answer all questions, and thus we had a few empty values.

Qualitative analysis

Flexibility - The vast majority of participants agreed that the Guidance received matched their need for support. However, nearly a third were either unsure or not really convinced about this (UH1, see Table 4.1).

Adaptability - While most users considered that the system was considering their inputs (UH3), an important share of them did not fully believe that the guidance was adapting to their changing tasks and goals (UH2).

Visibility - However, regarding the visibility of VizREvent, the VAU had positive feedback for both how easy it was to distinguish the Guidance from the rest of the app (UH4) and the visibility of the status and parameters of the Guidance (UH5).

Controllability - The controllability of the app was more divisive. On the one hand, the participants positively assessed the fact that Guidance could be switched off (UH6) and the presentations of alternative suggestions (UH7). On the other hand, most noted the lack of possibility to give direct feedback (UH8), and there was an even split on opinions about the presence of fine-tuning (UH9).

Explainability - Most VAU understood both the suggestions provided by the Guidance (UH11) and also to what direction they were being led to with them (UH10). Ultimately, the majority of them felt they could trust the Guidance (UH14). However, an important minority were unsure about that. Moreover, the main part of the participants reported that they could not prompt the system for more details on the guidance (UH13), and while most understood why the suggestions were proposed (UH12), two participants strongly disagreed with that, which should be noted.

Expressiveness - All participants (except one) acclaimed the expressiveness of the Guidance; there were only agreements (of various degrees) with both of the two heuristics

(UH15, UH16). It means that, to the participants, the encoding was not ambiguous, and that the language used to provide the Guidance was easy to understand.

Timeliness - Likewise, no users were disturbed in their workflow by the Guidance (UH18). Moreover, all except three felt that they received the Guidance at the right time (UH19).

Relevance - Lastly, the VAU globally felt positive about the Relevance of the Guidance. For instance, nearly all noted that it helped them save time (UH21), it facilitated their reasoning (UH22), and it gave them answers about the data (UH23). The other remaining heuristics from this effectiveness criterion were also globally positively rated.

CHAPTER 5

Discussion

5.1 On the Synergy between the Research Fields

In this section, we will answer the **(RQ1.1)** research question and sub-research questions defined in subsection 2.2.1.

First, from the classification of recent work, we have established that across both VizRec and ESVA fields, all degrees and prominence of Guidance are used **(RQ1.1.1)**.

Second, we have established that Guidance is used throughout several types of visualizations. Contributions spanned over the visualization types of matrix-based, timeline-based, hierarchy-based, Sankey-based and chart-based **(RQ1.1.2)**.

Then, all ES high-level tasks (or objectives) established in [ZSCL25] are supported by Guidance. The most supported tasks are, in order, cohort comparison, pattern exploration, stage progression, and then anomaly detection and correlation & causality **(RQ1.1.3)**.

Lastly, in ES, Guidance differs from VizRec by the degree to which it is applied, but mainly by its prominence. While VizRec systems feature prominently in expanded views, Guidance in ES can also be found embedded or implicit. Moreover, all VizRec contributions are somewhat providing Guidance at a higher degree (Orienting:1, Directing: 6, Prescribing:4) compared to ES contributions (Orienting: 3, Directing:5, Prescribing:1) **(RQ1.1.4)**, see section 2.3.

5.2 On VizREvent and its Evaluation

VizREvent was globally positively received by both VAE and VAU. This can be observed through the global average ratings of both parts of the study (which are both above 5). Moreover, this approach proves that integrating VizRec into ESVA can be done and

could be useful for analysts and users.
However, VizREvent also presents some weaknesses, as shown by the global average ratings of both studies being just above the sufficient Guidance threshold. As highlighted by both groups, the main weaknesses of VizREvent lie in its *Explainibility*, *Flexibility*, and *Controllability*. Regarding the first, both the experts and users noted the lack of details on demand. Moreover, not enough information (notably about the internal state of the Guidance, as some experts pointed out) is provided to make it clear why the suggestions are proposed and toward where the Guidance is leading the user. Likewise, for *Controllability*, the fine-tuning of the recommendation should be made clearer (especially how the interactions in the tuning process impact the VizRec outputs). Moreover, the participants of both parts correctly noted the non-presence of a feedback mechanism to adjust the recommendation. Lastly, and due to the nature of VizRec but also to recommendations overall, the main Guidance provided is Directing and doesn't change to Prescribing or Orienting. This was also correctly noted by some experts, as the recommendation only takes into account the selected Visualization instance and not a global context or previous user interactions.

Nevertheless, through VizREvent we also observed interesting points. VizREvent provides *expanded* Guidance (through the REC panel Figure 3.1). In both parts of the study, the *Visibility* of the Guidance was highly rated, demonstrating a quality of this Guidance Prominence. Moreover, the visual encodings of the recommendations (or *Expressiveness*) were also praised, which highlights Draco's ability to choose the right encodings . Lastly, throughout the development phase of VizREvent, we faced several performance issues in computing and displaying the recommendations. An important part of the work was to improve the efficiency of the pipeline, and this was rewarded by the rating approval of both sides (as *Timeliness* is the highest-rated criterion in both parts of the study).

Answering the research questions

With the VAU part of the study and with the positive ratings from the *Relevance* criteria, we have seen that the Guidance approach of VizREvent was successful in helping users in their analysis. In particular, participants agreed in majority that the Guidance helped them get out of a stalled analysis (UH19), solve their tasks (UH20), was adequate for the task they were doing (UH24), and made them discover new things and hypotheses about the data (UH25, UH26). This actually answers the remaining research questions **(RQ1.2)**, **(RQ1.3)**, and **(RQ.1.4)** because through this Orienting but mostly Directing Guidance approach, by using VizRec and in the context of sports Event analysis, as described in chapter 3 and evaluated in chapter 4, we have been able to positively improve the users' experience.

5.3 Limitations and Future Research

Rigor of literature search **-** In the writing of this thesis, it should be noted that the literature research first started by analyzing previous related surveys, their references,

and their citations. The keyword search with exclusion criteria was conducted later, after the research already had progressed significantly. As a result, pre-established research methodologies, like scoping reviews or systematic surveys [MPS+18], were not rigorously or strictly followed. However, with this thesis, we still aimed at providing an overview of contributions from the fields of Guidance, VizRec and ESVA. While we have highlighted some innovations from those fields, we do not claim to have comprehensively covered all existing literature. Future research could build upon this work by applying a systematic review framework to produce a more exhaustive coverage.

On the Classification - The classification created and used in this thesis was inspired by contributions from both research fields, as discussed in section 2.3. While this allows us to draw on the strengths of classifications and typologies from established surveys [GGJ+22, YM24, SdRBV24], we also inherit their weaknesses. Furthermore, while the combination of multiple design spaces to create this classification is valuable at establishing a bridge between those fields, it could be argued that it is ill-suited as it might lack the specificity of the context. Future work could address these limitations and create a typology tailored more explicitly to this domain.

Limitations of the study - The main limitation of the study here would be its online nature. Indeed, even though we provided participants with an exhaustive tutorial (with both a video and a text version), we cannot be fully sure whether the participants really watched it or if they understood all available interactions before starting to interact with the system and then rating the heuristics. Moreover, we cannot help them if they have questions or misunderstandings. This may have happened at least once, since one of the experts complained about not being able to turn the guidance off when it was shown in the tutorial that the panel can be collapsed (effectively shutting off the recommendations).

Limitations and Improvement of VizREvent - By using Draco, we also inherit its weaknesses. First, as mentioned in subsection 3.3.2, we had to remove most Event attributes from the datasets in order to use Draco. This obviously limits the extent of the analysis. Second, ESVA uses different types of visualization [YM24]. But here, due to Draco's limitation, only chart-based visualizations are handled. Thus limiting the capabilities of the analysis and possibly of the Guidance provided.

Then, we decided to use Draco as is, without providing additional constraints or modifying their weight. As mentioned earlier, the reasoning behind this choice was to judge the performance of existing tools to handle ES. And while we doubt it would have made significant changes, optimizing constraints and weight for ESVA could have provided better Guidance and a better user experience.

Draco doesn't inherently propose a way to explore the design space. Rather, they describe in their documentation that this can be done programmatically by generating incomplete specifications. We did exactly that, and while it led to our current version of VizREvent. Changing the variant generation process can alter significantly the recommendation

outputs. Other approaches to that generation process would have led to different, possibly interesting, outcomes.

The study done also provides us with axes for future improvement of VizREvent. First, the Explainability could be improved by making the recommendation process clearer. This could be done by adding details on demand where internal states of the recommendation could be displayed to the user (including the Draco cost of the chart, a more extended variant prompt, or maybe explanations about the variant generation as well). Second, an expert suggested adding functionalities, like undo/redo to help users recover from an error state. Such functionalities were considered but were not able to be implemented because of a lack of resources. Then, to improve the Relevance criterion, error states or inconsistencies could be better prevented, both in the recommendation outputs and also in the manual edition of the charts. For instance, we could also provide Guidance there by suggesting which encoding to choose in (C, Figure 3.1). Lastly, instead of just providing VizRec on the selected Visualization instance and eventually on selected data fields, the global state of the Canvas or the user history could also be taken into account as well. This will generate more alternatives and also provide Guidance that is tailored to a broader context or a deeper timeline (instead of just the present). This could lead to improved *Adaptability* and *Flexibility* (which received subpar ratings).

Extending Recommendation to other Visualization types - VizRec systems (including *VizREvent* through its use of Draco) only handle basic chart-based views. It could be interesting that different visualization types, like hierarchical-based or Sankey-based diagrams, could be recommended to users. Future research could determine the feasibility of such a recommendation system as well as evaluate its impacts on users.

Creating a new VizRec system tailored for ESVA - The most impactful Draco's limitation here is the data format they accept, being tabular data. And while we can still output some recommendations with this, it still severely limits the extent of the Guidance we can provide (see subsection 3.3.2). Future Research could focus on creating or adapting existing VizRec approaches to make them better suited to handle ES data and its heterogeneity so that recommendations can be applied to all data attributes.

CHAPTER 6

Conclusion

This thesis presented an attempt to build synergy between the research fields of Guidance, Visualization Recommendation, and Event Sequence Visual Analysis. This was first done through introducing a new categorization built upon existing taxonomies and then describing recent works with it. The classification of recent works in this thesis highlights the presence of Guidance concepts and mechanisms in Event Sequence visual analysis and therefore the potential to connect both fields even more.

Then, the thesis introduced *VizREvent*, a Visualization Recommendation for Event Sequence app created for this thesis. *VizREvent* is an attempt at integrating a well-known Visualization Recommendation approach (*Draco*) into the context of Event Sequence Visual Analysis. This app and its Guidance approach were evaluated using a two-part user study methodology adapted from a previous work. Results from this evaluation highlighted the effectiveness of this approach but also several paths for future improvements.

Overview of Generative AI Tools Used

AI tools such as ChatGPT [Ope24] or Le Chat[Mis24] were used in the creation process of *VizREvent* for development purposes. Regarding the writing of this thesis, these tools were exclusively used for the formatting of the tables (Table 2.1,Table 3.1 and Table 4.1) present in this thesis.

As mentioned in section 2.2, the AI tool *ResearchRabbit* [SY24] was used in the literature research process. However, it was only used to visualize citation and reference relationships between papers and not to summarize scientific articles.

List of Figures

2.1 Thread Visualization in *EventThread*. 19
2.2 Screenshot of *CallFlow*'s interface. 21
2.3 *Dashbot*'s interface. 22
2.4 Interface of *Dowsing*. 23
2.5 Advanced visualization components from *EventAction*. 24

3.1 Interface of *VizREvent*. 32
3.2 Dataset Selection pop-up in *VizREvent* 33
3.3 Tooltips in the Data Fields Display. 34
3.4 *VizREvent*'s global architecture. 37
3.5 Typical User Scenario of *VizREvent* 43

4.1 Results of the Expert part of the study. 47
4.2 Results of the User part of the study. 49

List of Tables

2.1 Classification of the selected contributions. 18

3.1 Data fields list 41

4.1 The sets of heuristics used to evaluate the effectiveness of guidance. . . . 46

Acronyms

ASP Answer Set Programming. 38

DS Data Settings. 32, 39

ES Event Sequence. 1–3, 5, 7–9, 14–17, 19, 22, 28, 29, 53, 55, 56

ESVA Event Sequence Visual Analysis. 2–4, 7–9, 14, 15, 17, 28, 29, 31, 38, 40, 53, 55, 56

JSON JavaScript Object Notation. 16, 25, 40, 41

REC Recommendation Panel. 33, 39, 40, 54

UI User Interface. 4, 7, 13, 27, 32, 37–39, 48

VA Visual Analytics. 2, 6, 15, 17, 19, 45, 49, 50

VAE Visual Analytics Experts. 45, 47–49, 53

VAU Visual Analytics Users. 45, 50, 51, 53, 54

VizRec Visualization Recommendation. 2–4, 6, 7, 16, 21–23, 26–28, 31, 32, 34, 35, 37–40, 48, 53–56

VP Visualization Panel/View Panel. 33, 39

Bibliography

[AdRORR19] Ünal Aksu, Adela del Río-Ortega, Manuel Resinas, and Hajo A. Reijers. An Approach for the Automated Generation of Engaging Dashboards. In Hervé Panetto, Christophe Debruyne, Martin Hepp, Dave Lewis, Claudio Agostino Ardagna, and Robert Meersman, editors, *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, pages 363–384, Cham, 2019. Springer International Publishing. DOI: https://doi.org/10.1007/978-3-030-33246-4_24.

[BM13] Matthew Brehmer and Tamara Munzner. A Multi-Level Typology of Abstract Visualization Tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–2385, December 2013. DOI: https://doi.org/10.1109/TVCG.2013.124.

[BOH11] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D^3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, December 2011. DOI: https://doi.org/10.1109/TVCG.2011.185.

[Bor25] Thibault Borde. Vizrevent github repository. https://github.com/Tthide/VizREvent, 2025. Accessed: 2025-06-04.

[CAA+20] Davide Ceneda, Natalia Andrienko, Gennady Andrienko, Theresia Gschwandtner, Silvia Miksch, Nikolaus Piccolotto, Tobias Schreck, Marc Streit, Josef Suschnigg, and Christian Tominski. Guide Me in Analysis: A Framework for Guidance Designers. *Computer Graphics Forum*, 39(6):269–288, 2020. DOI: https://doi.org/10.1111/cgf.14017.

[CAS+18] Christopher Collins, Natalia Andrienko, Tobias Schreck, Jing Yang, Jaegul Choo, Ulrich Engelke, Amit Jena, and Tim Dwyer. Guidance in the human–machine analytics process. *Visual Informatics*, 2(3):166–180, September 2018. DOI: https://doi.org/10.1016/j.visinf.2018.09.003.

[CCEA+24] Davide Ceneda, Christopher Collins, Mennatallah El-Assady, Silvia Miksch, Christian Tominski, and Alessio Arleo. A Heuristic Approach for Dual Expert/End-User Evaluation of Guidance in Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):997–1007, January 2024. DOI: https://doi.org/10.1109/TVCG.2023.3327152.

[CCI+15] Kristin Cook, Nick Cramer, David Israel, Michael Wolverton, Joe Bruce, Russ Burtner, and Alex Endert. Mixed-initiative visual analytics using task-driven recommendations. In *2015 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 9–16, October 2015. DOI: https://doi.org/10.1109/VAST.2015.7347625.

[CCW+18] Siming Chen, Shuai Chen, Zhenhuang Wang, Jie Liang, Yadong Wu, and Xiaoru Yuan. D-Map+: Interactive Visual Analysis and Exploration of Ego-centric and Event-centric Information Diffusion Patterns in Social Media. *ACM Trans. Intell. Syst. Technol.*, 10(1):11:1–11:26, November 2018. DOI: https://doi.org/10.1145/3183347.

[Cen23] Davide Ceneda. Heuristics guidance effectiveness – supplementary material. https://gitlab.cvast.tuwien.ac.at/dceneda/heuristics-guidance-effectiveness, 2023. Accessed: 2025-06-05.

[CGM+17] Davide Ceneda, Theresia Gschwandtner, Thorsten May, Silvia Miksch, Hans-Jorg Schulz, Marc Streit, and Christian Tominski. Characterizing Guidance in Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):111–120, January 2017. DOI: https://doi.org/10.1109/TVCG.2016.2598468.

[CGM19] Davide Ceneda, Theresia Gschwandtner, and Silvia Miksch. A Review of Guidance Approaches in Visual Data Analysis: A Multifocal Perspective. *Computer Graphics Forum*, 38(3):861–879, June 2019. DOI: https://doi.org/10.1111/cgf.13730.

[DCCS+21] Sandrine Da Col, Radu Ciucanu, Marta Soare, Nassim Bouarour, and Sihem Amer-Yahia. DashBot: An ML-Guided Dashboard Generation System. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, pages 4696–4700, New York, NY, USA, October 2021. Association for Computing Machinery. DOI: https://doi.org/10.1145/3459637.3481968.

[DD18] Victor Dibia and Çağatay Demiralp. Data2Vis: Automatic Generation of Data Visualizations Using Sequence to Sequence Recurrent Neural Networks, November 2018. DOI: https://doi.org/10.48550/arXiv.1804.03126.

[DPS+19] Fan Du, Catherine Plaisant, Neil Spring, Kenyon Crowley, and Ben Shneiderman. EventAction: A Visual Analytics Approach to Explainable Recommendation for Event Sequences. *ACM Trans. Interact. Intell. Syst.*, 9(4):21:1–21:31, August 2019. DOI: https://doi.org/10.1145/3301402.

[DPSS16] Fan Du, Catherine Plaisant, Neil Spring, and Ben Shneiderman. EventAction: Visual analytics for temporal event sequence recommendation. In *2016 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 61–70, October 2016. DOI: https://doi.org/10.1109/VAST.2016.7883512.

[Enc25] Encyclopædia Britannica, Inc. Definition of the word guidance. https://www.britannica.com/dictionary/guidance, 2025. Accessed: 2025-06-30.

[GB97] Karen Golden-Biddle. Constructing Opportunities for Contribution: Structuring Intertextual Coherence and "Problematizing" in Organizational Studies. *The Academy of Management Journal*, 40, October 1997. DOI: https://doi.org/10.2307/256926.

[GGJ+22] Yi Guo, Shunan Guo, Zhuochen Jin, Smiti Kaul, David Gotz, and Nan Cao. Survey on Visual Analysis of Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):5091–5112, December 2022. DOI: https://doi.org/10.1109/TVCG.2021.3100413.

[GXZ+18] Shunan Guo, Ke Xu, Rongwen Zhao, David Gotz, Hongyuan Zha, and Nan Cao. EventThread: Visual Summarization and Stage Analysis of Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):56–65, January 2018. DOI: https://doi.org/10.1109/TVCG.2017.2745320.

[HRM+23] Camille Harris, Ryan Rossi, Sana Malik, Jane Hoffswell, Fan Du, Tak Yeon Lee, Eunyee Koh, and Handong Zhao. SpotLight: Visual Insight Recommendation. In *Companion Proceedings of the ACM Web Conference 2023*, WWW '23 Companion, pages 19–23, New York, NY, USA, April 2023. Association for Computing Machinery. DOI: https://doi.org/10.1145/3543873.3587302.

[HS23] Wenkai Han and Hans-Jörg Schulz. Providing visual analytics guidance through decision support. *Information Visualization*, 22(2):140–165, April 2023. DOI: https://doi.org/10.1177/14738716221147289.

[JGC+21] Zhuochen Jin, Shunan Guo, Nan Chen, Daniel Weiskopf, David Gotz, and Nan Cao. Visual Causality Analysis of Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1343–1352, February 2021. DOI: https://doi.org/10.1109/TVCG.2020.3030465.

[KO17] Pawandeep Kaur and Michael Owonibi. A Review on Visualization Recommendation Strategies. volume 4, pages 266–273. SCITEPRESS, February 2017. DOI: https://doi.org/10.5220/0006175002660273.

[LLMB19] Po-Ming Law, Zhicheng Liu, Sana Malik, and Rahul C. Basole. MAQUI: Interweaving Queries and Pattern Mining for Recursive Event Sequence Exploration. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):396–406, January 2019. DOI: https://doi.org/10.1109/TVCG.2018.2864886.

[Mer25] Merriam-Webster, Inc. Definition of the word guidance. https://www.merriam-webster.com/dictionary/guidance, 2025. Accessed: 2025-06-30.

[Met24] Meta Platforms, Inc. React – a javascript library for building user interfaces. https://reactjs.org/, 2024. Accessed: 2025-06-12.

[Mis24] Mistral AI. Mistral chat. https://chat.mistral.ai/chat, 2024. Accessed: 2025-06-05.

[MPS+18] Zachary Munn, Micah D. J. Peters, Cindy Stern, Catalin Tufanaru, Alexa McArthur, and Edoardo Aromataris. Systematic review or scoping review? Guidance for authors when choosing between a systematic or scoping review approach. *BMC Medical Research Methodology*, 18(1):143, November 2018. DOI: https://doi.org/10.1186/s12874-018-0611-x.

[MSM+22] Jessica Magallanes, Tony Stone, Paul D Morris, Suzanne Mason, Steven Wood, and Maria-Cruz Villa-Uriol. Sequen-C: A Multilevel Overview of Temporal Event Sequences. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):901–911, January 2022. DOI: https://doi.org/10.1109/TVCG.2021.3114868.

[MWN+19] Dominik Moritz, Chenglong Wang, Greg L. Nelson, Halden Lin, Adam M. Smith, Bill Howe, and Jeffrey Heer. Formalizing Visualization Design Knowledge as Constraints: Actionable and Extensible Models in Draco. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):438–448, January 2019. DOI: https://doi.org/10.1109/TVCG.2018.2865240.

[NBJ+21] Huu Tan Nguyen, Abhinav Bhatele, Nikhil Jain, Suraj P. Kesavan, Harsh Bhatia, Todd Gamblin, Kwan-Liu Ma, and Peer-Timo Bremer. Visualizing Hierarchical Performance Profiles of Parallel Codes Using CallFlow. *IEEE Transactions on Visualization and Computer Graphics*, 27(4):2455–2468, April 2021. DOI: https://doi.org/10.1109/TVCG.2019.2953746.

[Ope24] OpenAI. Chatgpt (june 2024 version). https://chat.openai.com, 2024. Accessed: 2025-06-05.

[PMCEA+22] I. Pérez-Messina, D. Ceneda, M. El-Assady, S. Miksch, and F. Sperrle. A Typology of Guidance Tasks in Mixed-Initiative Visual Analytics Environments. *Computer Graphics Forum*, 41(3):465–476, 2022. DOI: https://doi.org/10.1111/cgf.14555.

[Pyt24] Python Software Foundation. Python language reference, version 3.x. https://www.python.org/, 2024. Accessed: 2025-06-12.

[QRD+22] Xin Qian, Ryan A. Rossi, Fan Du, Sungchul Kim, Eunyee Koh, Sana Malik, Tak Yeon Lee, and Nesreen K. Ahmed. Personalized Visualization Recommendation. *ACM Trans. Web*, 16(3):11:1–11:47, September 2022. DOI: https://doi.org/10.1145/3538703.

[Ron24] Armin Ronacher. Flask documentation. https://flask.palletsprojects.com/, 2024. Accessed: 2025-06-12.

[SdRBV24] Praveen Soni, Cyril de Runz, Fatma Bouali, and Gilles Venturini. A survey on automatic dashboard recommendation systems. *Visual Informatics*, 8(1):67–79, March 2024. DOI: https://doi.org/10.1016/j.visinf.2024.01.002.

[SHC+20] Dong Sun, Renfei Huang, Yuanzhe Chen, Yong Wang, Jia Zeng, Mingxuan Yuan, Ting-Chuen Pong, and Huamin Qu. PlanningVis: A Visual Analytics Approach to Production Planning in Smart Factories. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):579–589, January 2020. DOI: https://doi.org/10.1109/TVCG.2019.2934275.

[SJB+20] Fabian Sperrle, Astrik Jeitler, Jürgen Bernard, Daniel A. Keim, and Mennatallah El-Assady. *Learning and Teaching in Co-Adaptive Guidance for Mixed-Initiative Visual Analytics*. The Eurographics Association, 2020. DOI: https://doi.org/10.2312/eurova.20201088.

[SMWH17] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):341–350, January 2017. DOI: https://doi.org/10.1109/TVCG.2016.2599030.

[SPM24] Johanna Schmidt, Bernhard Pointner, and Silvia Miksch. Visual Analytics for Understanding Draco's Knowledge Base. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):392–402, January 2024. DOI: https://doi.org/10.1109/TVCG.2023.3326912.

[Sta21] StatsBomb. Statsbomb open data. https://github.com/statsbomb/open-data, 2021. Accessed: 2025-06-05.

[SXS+21] Danqing Shi, Xinyue Xu, Fuling Sun, Yang Shi, and Nan Cao. Calliope: Automatic Visual Data Story Generation from a Spreadsheet. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):453–463, February 2021. DOI: https://doi.org/10.1109/TVCG.2020.3030403.

[SY24] Oleksandr Shtykalo and Iuliia Yamnenko. ChatGPT and Other AI Tools for Academic Research and Education. In Andriy Luntovskyy, Mikhailo Klymash, Igor Melnyk, Mykola Beshley, and Alexander Schill, editors, *Digital Ecosystems: Interconnecting Advanced Networks with AI Applications*, pages 605–630, Cham, 2024. Springer Nature Switzerland. DOI: https://doi.org/10.1007/978-3-031-61221-3_29.

[vdLdFvdEV23] Sanne van der Linden, Evie de Fouw, Stef van den Elzen, and Anna Vilanova. A survey of visualization techniques for comparing event sequences. *Computers & Graphics*, 115:522–542, October 2023. DOI: https://doi.org/10.1016/j.cag.2023.05.016.

[vdLWvG+23] S. van der Linden, B.m. Wulterkens, M.m. van Gilst, S. Overeem, C. van Pul, A. Vilanova, and S. van den Elzen. FlexEvent: going beyond Case-Centric Exploration and Analysis of Multivariate Event Sequences. *Computer Graphics Forum*, 42(3):161–172, 2023. DOI: https://doi.org/10.1111/cgf.14820.

[WQM+17] Kanit Wongsuphasawat, Zening Qu, Dominik Moritz, Riley Chang, Felix Ouk, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Voyager 2: Augmenting Visual Analysis with Partial View Specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 2648–2659, Denver Colorado USA, May 2017. ACM. DOI: https://doi.org/10.1145/3025453.3025768.

[WWZ+22] Aoyu Wu, Yun Wang, Mengyu Zhou, Xinyi He, Haidong Zhang, Huamin Qu, and Dongmei Zhang. MultiVision: Designing Analytical Dashboards with Deep Learning Based Recommendation. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):162–172, January 2022. DOI: https://doi.org/10.1109/TVCG.2021.3114826.

[XSZX22] Shaobin Xu, Minghui Sun, Zhengtai Zhang, and Hao Xue. Exploring Multivariate Event Sequences with an Interactive Similarity Builder. 2022. DOI: https://doi.org/10.1111/cgf.14539.

[XWY+20] Ke Xu, Yun Wang, Leni Yang, Yifang Wang, Bo Qiao, Si Qin, Yong Xu, Haidong Zhang, and Huamin Qu. CloudDet: Interactive Visual Analysis of Anomalous Performances in Cloud Computing Systems. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1107–1117, January 2020. DOI: https://doi.org/10.1109/TVCG.2019.2934613.

[XYF+17] Shuai Xiao, Junchi Yan, Mehrdad Farajtabar, Le Song, Xiaokang Yang, and Hongyuan Zha. Joint Modeling of Event Sequence and Time Series with Attentional Twin Recurrent Neural Networks, March 2017. DOI: https://doi.org/10.48550/arXiv.1703.08524.

[YGZM23] Junran Yang, Péter Ferenc Gyarmati, Zehua Zeng, and Dominik Moritz. Draco 2: An Extensible Platform to Model Visualization Design. In *2023 IEEE Visualization and Visual Analytics (VIS)*, pages 166–170, October 2023. DOI: https://doi.org/10.1109/VIS54172.2023.00042.

[YM24] Anton Yeshchenko and Jan Mendling. A survey of approaches for event sequence analysis and visualization. *Information Systems*, 120:102283, February 2024. DOI: https://doi.org/10.1016/j.is.2023.102283.

[YNKJ+25] Peilin Yu, Aida Nordman, Marta Koc-Januchta, Konrad Schönborn, Lonni Besançon, and Katerina Vrotsou. Revealing Interaction Dynamics: Multi-Level Visual Exploration of User Strategies with an Interactive Digital Environment. *IEEE Transactions on Visualization and Computer Graphics*, 31(1):831–841, January 2025. DOI: https://doi.org/10.1109/TVCG.2024.3456187.

[ZMD+22] Zehua Zeng, Phoebe Moh, Fan Du, Jane Hoffswell, Tak Yeon Lee, Sana Malik, Eunyee Koh, and Leilani Battle. An Evaluation-Focused Framework for Visualization Recommendation Algorithms. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):346–356, January 2022. DOI: https://doi.org/10.1109/TVCG.2021.3114814.

[ZSCL25] Kazi Tasnim Zinat, Saimadhav Naga Sakhamuri, Aaron Sun Chen, and Zhicheng Liu. A Multi-Level Task Framework for Event Sequence Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 31(1):842–852, January 2025. DOI: https://doi.org/10.1109/TVCG.2024.3456510.

[ZSJ+20] Sujia Zhu, Guodao Sun, Qi Jiang, Meng Zha, and Ronghua Liang. A survey on automatic infographics and visualization recommendations. *Visual Informatics*, 4(3):24–40, September 2020. DOI: https://doi.org/10.1016/j.visinf.2020.07.002.

[ZWZ+24] Jiamin Zhu, Meixuan Wu, Yi Zhou, Nan Cao, Haotian Zhu, and Min Zhu. Dowsing: a task-driven approach for multiple-view visualizations dynamic recommendation. *Journal of Visualization*, 27(4):695–712, August 2024. DOI: https://doi.org/10.1007/s12650-024-00989-9.

[ZYG+23] Kazi Tasnim Zinat, Jinhua Yang, Arjun Gandhi, Nistha Mitra, and Zhicheng Liu. A Comparative Evaluation of Visual Summarization Techniques for Event Sequences, June 2023. DOI: https://doi.org/10.48550/arXiv.2306.02489.