Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

STRive: An association rule-based system for the exploration of spatiotemporal categorical data

Mauro Diaz[a,*], Luis Sante[a], Joel Perca[a], João Victor da Silva[b], Nivan Ferreira[b], Jorge Poco[a]

[a] *Fundação Getulio Vargas. Praia de Botafogo, 190 - Botafogo, Rio de Janeiro - RJ, 22250-145, Brazil*
[b] *Universidade Federal de Pernambuco. Av. Jornalista Aníbal Fernandes, s/n – Cidade Universitária, Recife, Pernambuco, Brazil*

ARTICLE INFO

Article history:
Received August 13, 2025

Keywords: Spatiotemporal visualization, Association rule mining, Pattern extraction, Visual Analytics system.

ABSTRACT

Effectively analyzing spatiotemporal data plays a central role in understanding real-world phenomena and informing decision-making. Capturing the interaction between spatial and temporal dimensions also helps explain the underlying structure of the data. However, most datasets do not reveal attribute relationships, requiring additional algorithms to extract meaningful patterns. Existing visualization tools often focus either on attribute relationships or spatiotemporal analysis, but rarely support both simultaneously. In this paper, we present *STRive* (SpatioTemporal Rule Interactive Visual Explorer), a visual analytics system that enables users to uncover and explore spatial and temporal patterns in data. At the core of *STRive* lies Association Rule Mining (ARM), which we apply to spatiotemporal datasets to generate interpretable and actionable insights. We combine ARM with multiple interactive mechanisms to analyze the extracted relationships. Association rules serve as interpretable guidance mechanisms for visual analytics by highlighting the meaningful aspects of the data that users should investigate. Our methodology includes three key steps: rule generation, rule clustering, and interactive visualization. *STRive* offers two modes of analysis. The first operates at the rule cluster level and includes four coordinated views, each showing a different facet of a cluster, including its temporal and spatial behavior. The second mode mirrors the first but focuses on individual rules within a selected cluster. We evaluate the effectiveness of *STRive* through two case studies involving real-world datasets — fatal vehicle accidents and urban crime. Results demonstrate the system's ability to support the discovery and analysis of interpretable patterns in complex spatiotemporal contexts.

1. Introduction

Spatiotemporal data mining continues to pose significant challenges in real-world scenarios because relationships between attributes often lack evident structure [1, 2, 3]. Due to their inherent complexity, visualization methods aimed at exploring spatiotemporal phenomena increasingly benefit from user-guided mechanisms, which have gained prominence in visual analytics environments [4]. A notable guidance mechanism is *Association Rule Mining (ARM)*, which is widely employed to reveal meaningful relationships within datasets [5]. ARM systematically extracts associations, clearly highlighting co-occurrences and conditional relationships among variables.

Association rules provide effective suggestions or initial exploration points within visual analytics workflows, enabling users to focus analyses on specific data aspects. Each rule explicitly defines relationships through *"if-then"* statements, clar-

*Corresponding author:
e-mail: dany.espino@fgv.br (Mauro Diaz)
[1] Footnote 1.

ifying the interactions between variables. Instead of initiating analyses from scratch or employing manual queries, analysts leverage these rules as preliminary hypotheses, accelerating their discovery processes. Furthermore, association rules offer greater interpretability than opaque models, such as neural networks or ensemble methods, allowing analysts to comprehend underlying patterns without requiring intricate explanations.

Although ARM demonstrates extensive applicability across various domains, its integration within spatiotemporal visual analytics faces considerable challenges. Generating large-scale association rules introduces computational complexities, and effectively visualizing these rules constitutes a significant hurdle in spatiotemporal contexts.

In this work, *we present a novel methodology to address these challenges and uncover interpretable patterns in spatiotemporal datasets.* The approach applies association-rule mining to automatically detect significant patterns, which are then systematically organized into meaningful clusters. We implemented the methodology in a visual analytics system, *STRive*, which provides multiple coordinated views that facilitate the interactive exploration of these patterns using well-established visualizations. The system enables a structured drill-down analysis, beginning with general trends and progressing toward detailed insights. *STRive* effectively directs analysts' attention to critical areas of interest, enabling more focused and efficient exploration. *STRive* Interactive visualizations enable users to thoroughly investigate rule compositions, temporal dynamics, pattern comparisons, and spatiotemporal occurrences. By combining powerful computational methods with intuitive visual analytics, STRive significantly enhances analysts' ability to interpret and gain actionable insights from spatiotemporal data. Our contributions are as follows:

- A methodology using established algorithms for identifying and clustering temporal patterns.
- *STRive*, an interactive visual analytics tool that significantly improves exploration and interpretation of spatiotemporal datasets.
- Two comprehensive case studies using real-world datasets that demonstrate significant findings, accompanied by clear and insightful explanations.

2. Related Works

Analyzing spatiotemporal data patterns is necessary for public safety, transportation, epidemiology, and environmental monitoring [1]. Visual analytics systems supporting this analysis typically use coordinated interactive views. For comprehensive reviews of these visualization techniques, see Liu et al. [6] and Mota et al. [7]. Unlike purely user-driven exploration, our method automatically identifies interpretable patterns first and then uses interactive visualizations for detailed analysis. This section reviews guidance in visual analytics, association rule visualization, and hypergraph representations.

2.1. Guidance in Visual Analytics

Guidance in visual analytics is "a computer-assisted process that actively resolves a knowledge gap encountered by users during an interactive visual analytics session" [8]. According to Collins et al. [9], guidance supports analysts in identifying and managing data patterns. They propose approaches for implementing guidance, such as facilitating pattern extraction — visually through effective representations or computationally via suitable algorithms. They also emphasize guidance in pattern evaluation, considering properties like frequency, intensity, and rate of change. Several studies have introduced strategies for selecting [4, 10, 11] and evaluating [9] guidance techniques, underscoring their practical importance.

Numerous visual analytics systems have incorporated guidance mechanisms. Doraiswamy et al. [12] propose a method that uses topological features to identify events in scalar time-varying data automatically. Their visual interface guides users toward noteworthy events and supports similarity searches. Valdivia et al. [13] represent data as graphs and apply graph wavelet transforms to detect regions with significant variation, identifying abrupt changes as important events. Liu et al. [14] developed a workflow for pattern discovery and comparison, utilizing tensor decomposition to recommend optimal data slices and highlight patterns automatically.

Although these approaches successfully utilize algorithms to guide exploration, the complexity of the underlying methods may require users to exert additional effort to interpret the extracted patterns.

2.2. Association Rules in Visual Analytics

Association rules summarize interpretable relationships within data and serve as effective guidance mechanisms. Varu et al. [15] present a framework using an item-to-rule matrix visualization. Items form rows, rules form columns, and glyphs indicate antecedent-consequent relationships. Metrics are visually represented through histograms aligned with rows and columns. Menin et al. [16] utilize linked visualizations, including scatterplots for rule metrics, chord diagrams for item relationships, and association graphs for exploratory item analysis. Wang et al. [17] employ association rules to analyze spatiotemporal climate data, discretizing continuous attributes for effective analysis. Their approach integrates spatial context, enriching the analysis beyond attribute relationships alone.

Despite advances, existing visualizations for association rules often lack sufficient contextual visualization, leading to information loss. Fister et al. [18] provide a detailed critique of these limitations. In contrast, our system STRive addresses this limitation by embedding association rules within a rich spatiotemporal framework. It enables users to analyze the relationships between attributes and *when* and *where* these patterns emerge.

2.3. Hypergraph Representations

Association rules can be modeled using hypergraphs. In this representation, items are represented as nodes, and rules are defined as hyperedges that connect multiple nodes. Visualizing such structures is challenging. Valdivia et al. [19] propose a matrix-like visualization to display dynamic hypergraphs, with nodes represented as rows and hyperedges as vertical lines connecting relevant nodes across time. Oliver et al. [20] develop polygon-based hypergraph visualizations using iterative simplification to optimize layouts. Additional node-link, timeline-based, and matrix-based visualization approaches are reviewed in Fischer et al. [21].

Our work draws particularly from Valdivia et al. [19], adopting their approach of representing ordered sets of hyperedges (rules) over time. We extend this method by encoding additional contextual data to support comprehensive analysis.

3. Background

We now briefly present key concepts in Association Rule Mining (ARM). For a more detailed discussion on this topic, see Zhang and Zhang [22] and Nath et al. [23]. Agrawal et al. [5] first introduced ARM as a data mining method to uncover frequent associations between variables in a dataset. Let $\mathcal{D}$ denote the dataset, and capital letters (*e.g.*, X, Y) represent subsets of $\mathcal{D}$. Following Kisilevich et al. [24], we focus on spatiotemporal (ST) event datasets. Each entry in the dataset is a point in space-time, defined as $p = (location_p, t_p, attrib_p)$, where $attrib_p$ is a vector of additional attributes. To apply ARM, we assume all components, including location and time, are categorical. When necessary, we discretize geographic coordinates into administrative regions (*e.g.*, countries or states) and time into intervals such as months or years.

ARM expresses relationships as association rules in the form $r := X \Rightarrow Y$, where X is the antecedent ($Ante(r)$) and Y is the consequent ($Cons(r)$). A rule states that if X appears, then Y is likely to follow. For instance, in a sales dataset, the rule $\{Bread, Coffee\} \Rightarrow \{Milk\}$ suggests that customers who buy bread and coffee often also buy milk. Although originally applied to market basket analysis [5], researchers now use ARM in other domains such as traffic accident analysis [25], electronic health records [26], and microblogging text analysis [27].

The ARM process begins by identifying *frequent itemsets*, *i.e.*, values that commonly appear together. Algorithms like Apriori [28] and FP-Growth [29] perform this task. From these *itemsets*, the algorithm generates association rules by splitting the itemsets into antecedent and consequent parts. To evaluate the quality of the generated rules, ARM uses several metrics: For example:

- **Support** [5] measures how often an itemset appears in the dataset: $sup(X) = |X|/|\mathcal{D}|$. For a rule, $sup(X \Rightarrow Y) = sup(X \cap Y)$.
- **Confidence** [5] estimates the probability of Y appearing when X is present: $conf(X \Rightarrow Y) = |X \cap Y|/|X|$.
- **Lift** [30] measures the strength of the association by comparing the joint probability of X and Y to their expected probability if independent: $lift(X \Rightarrow Y) = sup(X \cup Y)/(sup(X) \times sup(Y))$.

These metrics help identify meaningful rules; however, setting appropriate threshold values can be challenging. ARM algorithms often generate a large number of rules [31], which can overwhelm the analysis process.

Low thresholds may capture rare but important patterns, especially in critical datasets. For example, in a dataset of 1,000 crime incidents, a minimum support of 0.1 requires at least 100 matching incidents to retain a rule. Lowering the threshold increases coverage but also inflates the number of rules. The best threshold values and choice of metrics vary by application. In this context, interactive visualization plays a key role. It allows analysts to explore different thresholds and evaluate rules dynamically.

Our approach builds on these foundations to support the analysis of spatiotemporal data with ARM. We utilize association rules to automatically extract relevant patterns. These patterns serve as guidelines that facilitate the visual exploration process. Unlike black-box models, association rules provide transparent and interpretable insights that are directly tied to the dataset's attributes.

4. System Overview

This section outlines the design methodology of our system. We describe the system requirements, analytical tasks, and overall workflow implemented in *STRive*.

4.1. System Requirements

To guide users toward meaningful and explainable patterns, we define key design requirements for *STRive*. These requirements are based on a review of interactive visual analytics for spatiotemporal pattern discovery and our team's prior research experience.

- **Analyze relationships among attributes.** Understanding how attributes co-occur can uncover hidden patterns. Due to the numerous possible attribute combinations, users benefit from automated techniques that identify frequent associations. ARM supports this task by identifying recurring combinations in the dataset [32, 31].
- **Identify temporal trends and spatial patterns.** Detecting how patterns evolve over time and space is essential for understanding trends and anomalies. Interactive visualizations help users observe when and where association rules appear [18].
- **Manage a large number of association rules.** ARM often produces many rules, making interpretation difficult. Clustering and summarization techniques simplify exploration and help users compare patterns more easily [33].

The first two requirements align with long-established goals in spatiotemporal visualization research, whereas the third addresses specific limitations inherent to applying ARM in this context.

4.2. Analytical Tasks

Based on the system requirements, we define the main analytical tasks supported by *STRive*. These tasks help users discover patterns, observe temporal and spatial concentrations, detect shifts over time, and explore results interactively.

- **T1. Attribute, date, and algorithm configuration.** The system should offer controls to select attributes of interest, define a time range, set thresholds for rule measures (*e.g.*, support, lift), and configure clustering granularity.

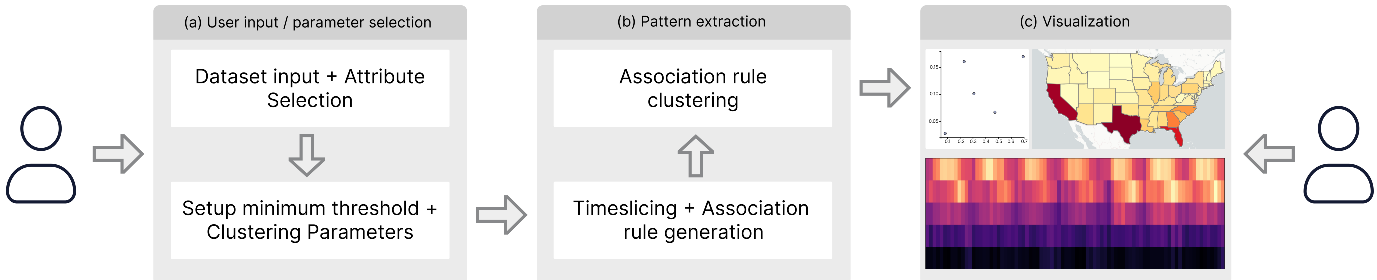

Fig. 1. ***STRive*** **workflow. (a) The user provides a dataset of interest, selects relevant attributes, sets up minimum thresholds for rule generation and defines clustering parameters. (b) The system generates rules and clusters based on the user's configuration and computes their metrics. (c) Finally, users explore the generated patterns through six visual components, organized into two views.**

- **T2. Rule and cluster composition comparison.** The user should be able to compare common associations at the cluster level and explore attribute differences at the rule level in a unified view.
- **T3. Temporal behavior analysis.** The system should display the frequency of each rule and cluster over time slices. This helps detect recurring trends and unusual time-dependent behaviors.
- **T4. Spatial distribution analysis.** The user should be able to examine the spatial distribution of rule and cluster occurrences to understand where specific patterns emerge.
- **T5. Rule and cluster ordering.** The system should organize rules and clusters based on the similarity of their temporal patterns.
- **T6. Cluster metric display.** The system should present key cluster metrics — such as rule count, average support, and average confidence — in one layout to simplify performance comparison.

4.3. Workflow

Figure 1 illustrates the system workflow. Users begin by uploading a dataset in CSV format, which includes temporal and location fields, as described in Section 3. The tool accepts uploading datasets with multiple categorical attributes, provided they contain both a `DATE` column (formatted as `YYYY-mm-dd`) and a `PLACE` column. Users must also upload a GeoJSON file to render the locations specified in the `PLACE` field. The place names in the GeoJSON file must exactly match the values in the `PLACE` column so that aggregated data displays correctly. After both files are loaded, users select the attributes to be used for generating association rules. Records with null or 'Unknown' values in selected columns are discarded to prevent meaningless associations. When attributes contain many distinct values, users should group similar values to reduce complexity. For instance, they can categorize vehicle models into broader types such as SUVs, sedans, or trucks. This improves rule generation by increasing support levels. Next, users define the analysis time range and set thresholds for support and lift. They also configure a resolution parameter for clustering, explained in Section 5.

The system then extracts and clusters association rules (Figure 1.b). It filters records by the selected time range and divides the data into time slices. For each slice, the system generates rules using the specified configuration. It removes duplicate rules that appear across slices and applies clustering to the combined rule set. Section 5 details the rule generation, deduplication, and clustering steps. Finally, users explore the patterns through *STRive*'s interface (Figure 1.c). The interface includes coordinated views for analyzing clusters and drilling down into individual rules. Section 6 describes the interface components and interactions.

5. Association Rule Mining and Clustering

A key feature of our system is the drill-down analysis capability. Users begin with an overview of rule clusters and then proceed to a detailed exploration of individual rules. This section describes the processes of generating and clustering association rules.

Our system utilizes the FP-Growth algorithm for ARM due to its computational efficiency compared to alternatives, such as Apriori. Users typically set thresholds, such as minimum support and lift, to discard irrelevant rules.

Our primary goal is to identify spatiotemporal patterns using ARM. To incorporate temporal information, we divide the dataset into multiple time slices, as detailed in Section 4.3, and generate rules separately for each period. Generating rules for the entire dataset may highlight global trends but can miss important time-specific patterns. By examining rules across multiple slices, we track their evolution over time.

However, two key challenges can complicate the visualization and interpretation of these temporal rules. First, small temporal shifts in data can generate *superfluous* rules — rules differing only by attribute placement between antecedent and consequent. For example, it is common to generate rules such as the following three:

$$\{A, B, C\} \Rightarrow \{D\}$$
$$\{A, B\} \Rightarrow \{C, D\}$$
$$\{A, B, D\} \Rightarrow \{C\}$$

Though semantically distinct, these rules share the same support. To reduce redundancy, we retain only the rule with the highest mean lift from each group of similar rules. Second, even after this reduction step, the number of generated rules

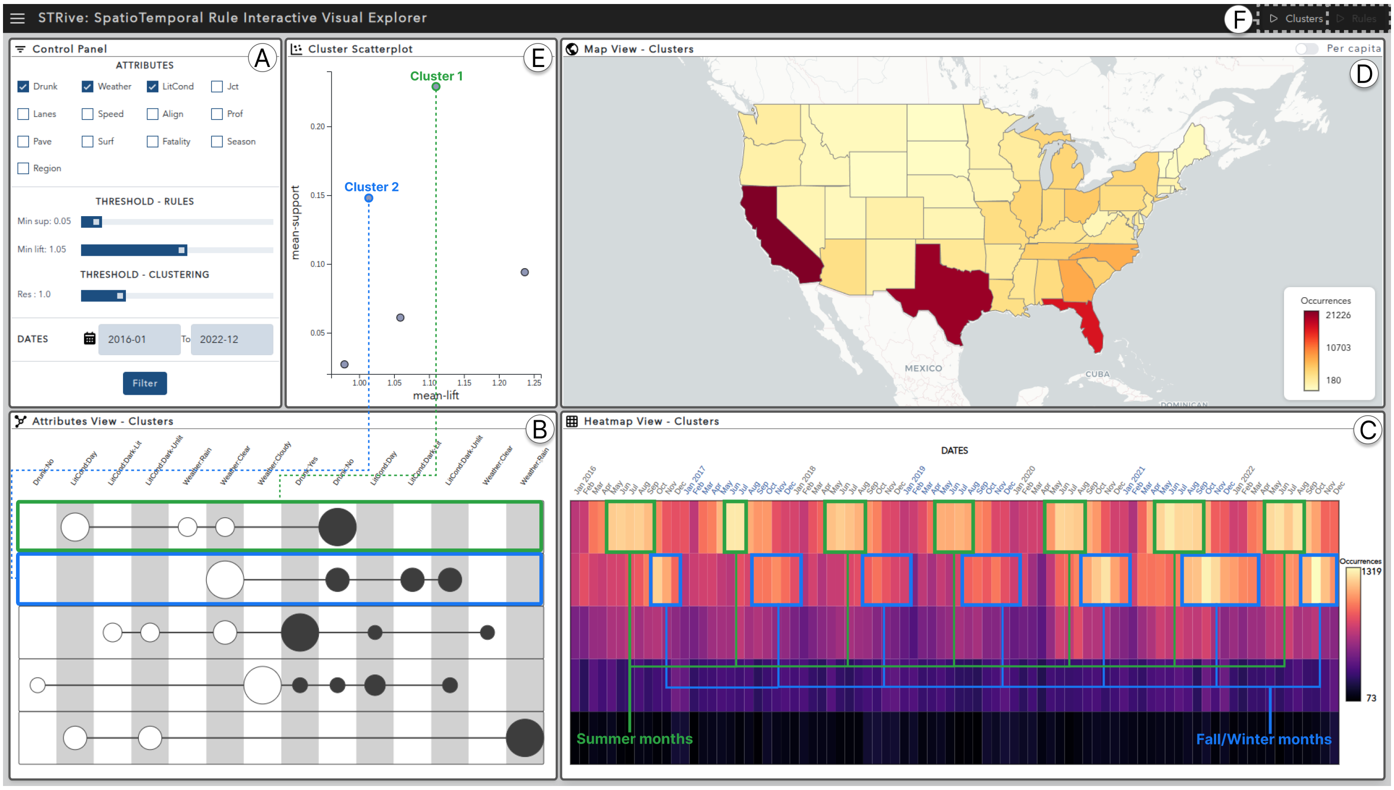

Fig. 2. ***STRive*** **interface with annotated components from the first case study. (A)** ***Control Panel*** **– allows users to select attributes and configure rule thresholds. (B)** ***Attributes View*** **– displays the composition of each cluster. (C)** ***Heatmap View*** **– reveals the temporal behavior of clusters across time slices. (D)** ***Map View*** **– shows the spatial distribution of rule clusters. (E)** ***Cluster Scatterplot*** **– provides a summary of cluster performance based on rule metrics. (F)** ***Mode Selector*** **– allows to change between** ***cluster-level*** **and** ***rule-level*** **modes.**

may remain large and difficult to visualize effectively. To handle this, we cluster rules into semantically coherent groups. We use the similarity metric from Fister et al. [34], defined as:

$$\mathrm{Sim}(r_1, r_2) = \frac{|\mathrm{Ante}(r_1) \cap \mathrm{Ante}(r_2)| + |\mathrm{Cons}(r_1) \cap \mathrm{Cons}(r_2)|}{|\mathrm{Ante}(r_1) \cup \mathrm{Ante}(r_2)| + |\mathrm{Cons}(r_1) \cup \mathrm{Cons}(r_2)|}$$

This metric evaluates rule similarity based on common attributes in antecedents and consequents. We build a complete similarity graph, where nodes represent rules and edge weights reflect the degree of similarity between rules. For clustering, we employ the Louvain algorithm [35], a greedy, agglomerative method effective in detecting communities without predefined cluster counts. Preliminary tests confirmed its superior cluster coherence and computational efficiency performance. The Louvain algorithm includes a resolution parameter that controls cluster granularity: lower resolutions yield fewer, larger clusters, while higher resolutions create more numerous, smaller clusters. Users can adjust this parameter based on their analysis needs. The resulting clusters serve as the basis for initial data exploration, providing meaningful groupings that simplify further analysis.

6. The *STRive* System

STRive is a novel visualization tool designed to support interactive visual analysis of spatiotemporal data via association rules. It comprises seven interactive components organized in two windows (the main interface is shown in Figure 2) designed to support the design tasks described in Section 4.2. Some components support two modes, *cluster-level* and *rule-level*, which can be toggled using the buttons located in the top-right corner of the interface (Figure 2.F). These modes adjust the views to accommodate different levels of detail. We describe each of these components in the rest of this section.

Control Panel. The *Control Panel* (Figure 2.A) provides options for configuring the rule generation and clustering processes **(T1)**. Users can import a spatiotemporal dataset and select the relevant attributes to be considered for rule generation. Additionally, the panel allows the user to specify starting and ending dates for rule generation. To explore diverse clustering configurations, the panel includes an input field for adjusting the resolution parameter of the clustering algorithm, enabling users to control the size of the resulting rule clusters.

Attributes View. After users configure the rule-generation and clustering parameters in the *Control Panel*, the resulting rule clusters appear in the *Attributes View* (Figure 2.B). The design adapts the ARMatrix visualization by Varu et al. [15] and extends the work of Valdivia et al. [19] with additional information. Each row represents one cluster **(T2, T5)**, and each column corresponds to an attribute–value pair found in the rules. Within a row, a circle marks the presence of an attribute; its radius encodes the frequency with which that attribute occurs across the cluster's rules, while its fill color distinguishes antecedent attributes (white) from consequent attributes (black).

In *rule-level* mode, the view switches to display individual rules rather than clusters: each row now represents a single rule, the column layout is unchanged, and all circles have equal size because a given attribute can appear only once within a rule. Users can interact with the *Attributes View* by selecting a cluster, which updates the *Map View* to display the occurrences satisfying the rules per location. Clicking on a specific cluster also makes the *Attributes View* and *Heatmap View* eligible for switching to *rule-level* mode, allowing users to inspect the individual rules within that cluster if they choose to activate it.

Heatmap View. The *Heatmap View* (Figure 2.C) depicts the temporal evolution of the records that satisfy each rule cluster **(T3)**. Rows represent clusters, columns represent time intervals, and cell colour encodes the record count, making temporal trends easy to spot. Clusters with similar temporal signatures are placed adjacent to one another, and the row order is shared with the *Attributes View*; scrolling in either view keeps the other in sync.

The view also supports a *rule-level* mode. When a cluster is selected, each row switches to an individual rule while the columns continue to denote time intervals. The cell color now indicates how often that rule is satisfied in each period. This finer granularity complements the *Attributes View* by directly linking rule structure to temporal behavior, while maintaining the same row order and linked scrolling for a cohesive exploration experience.

Map View. The *Map View* (Figure 2.D) depicts the spatial distribution of records captured by the generated clusters **(T4)**. After rule generation and clustering, the map initializes at the spatial resolution defined by the dataset's geographic attribute. Each location is shaded with a sequential color scale that encodes the total number of occurrences observed at that location. By default, the map aggregates occurrences from *all* clusters, providing an overview of the entire dataset. Selecting a cluster in the *Attributes View* filters the map to display only that cluster's occurrences, revealing its distinctive spatial footprint. The view is tightly linked to the *Heatmap View*: clicking a time label in the heatmap restricts the map to the chosen interval. If a cluster is selected, the filter becomes the intersection of the cluster and the period; otherwise, the map shows occurrences from every cluster within that period. In *rule-level* mode, the design and interactions remain unchanged, but the focus shifts to individual rules. Selecting a rule in the *Attributes View* updates the map to display its corresponding occurrences, and time filtering continues to operate via the heatmap, allowing users to track how each rule's spatial pattern evolves.

Cluster Scatterplot. To facilitate easy comparison of cluster performances, the *Cluster Scatterplot* (Figure 2.E) provides an overview of cluster-level metrics **(T6)**. Each dot in the scatterplot represents a cluster, and two metrics determine its position. By default, the X-axis shows the number of rules within a cluster, and the Y-axis displays the mean lift of the cluster. Customization of the scatterplot can be done by clicking on either axis label, which will display a dropdown menu listing additional metrics: mean number of occurrences, mean support, and mean confidence. Upon selecting a different metric, the scatterplot is redrawn to reflect the updated metrics. Additional interaction between views was implemented: selecting a cluster in the *Attributes View* highlights its corresponding point in the scatterplot, while selecting a point in the scatterplot highlights the respective cluster in the *Attributes View*. This enables the joint analysis of cluster composition and cluster metrics. This component is only available in cluster mode, as its purpose is to compare general properties across clusters.

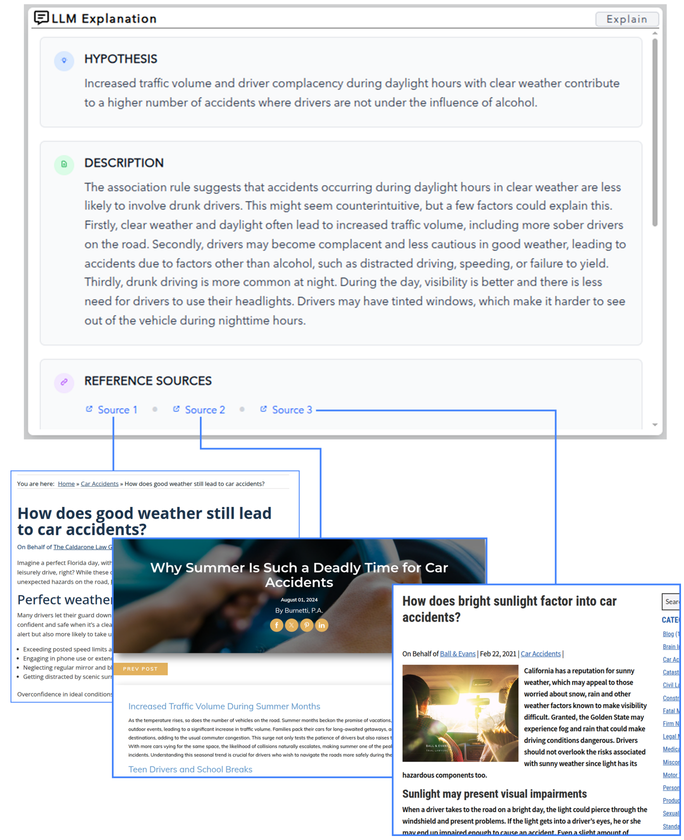

Fig. 3. *STRive* LLM explanation panel. The component uses the `gemini-2.0-flash` model to generate multiple hypotheses explaining the selected rule and searches the internet for supporting sources.

LLM Explanation Panel. The *LLM Explanation Panel* (Figure 3) employs a large-language model (`gemini-2.0-flash`) to generate contextual explanations for a selected rule. After users select a rule in the *Attributes View*, they can refine the scope by choosing specific regions on the map or particular time intervals. If no additional selection is made, the system defaults to the three locations with the highest rule frequency and the full date range. Pressing the *Explain* button sends this context as a prompt (see Appendix B.1) to the LLM, which returns hypotheses, pattern descriptions, and links to relevant external sources such as news articles. The resulting insights are displayed in the panel. This component is available only in *rule-level* mode, where such fine-grained explanations are most useful.

Implementation details. The system was implemented using Python, JavaScript, and HTML. The backend was built us-

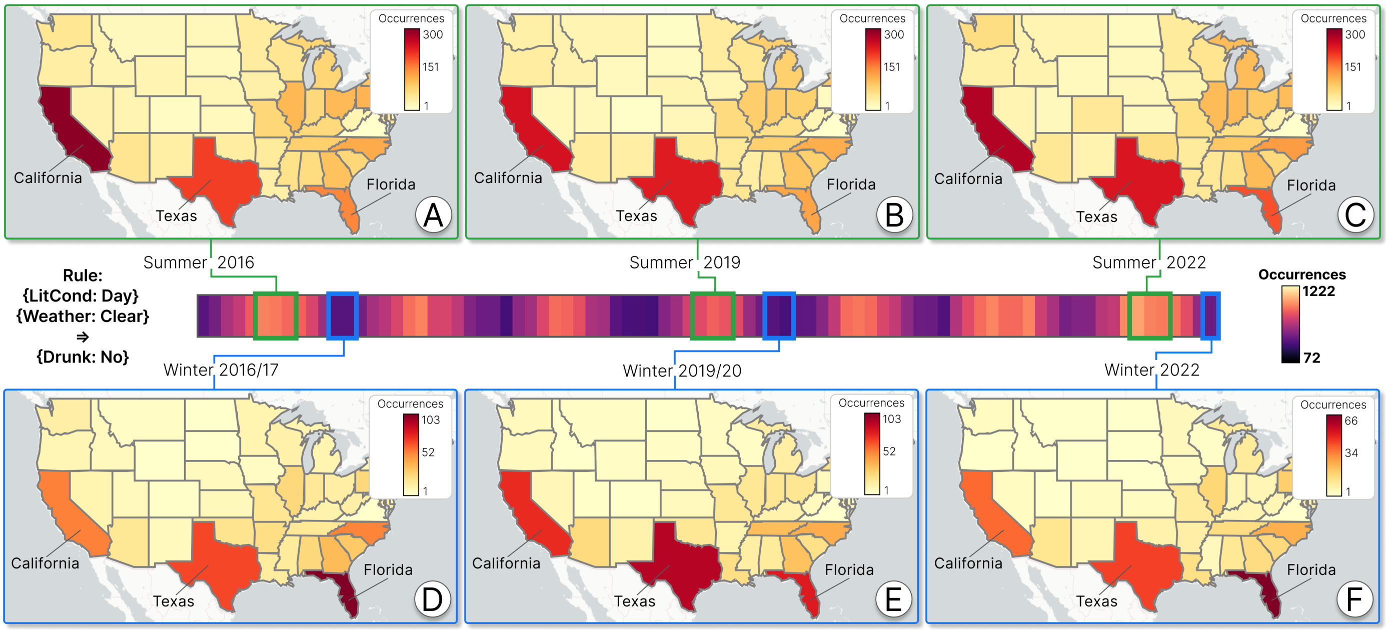

Fig. 4. Example rule from Cluster 1 involving the attributes `Drunk`, `Lit Condition`, and `Weather`. During the summer months, accidents characterized by daylight, clear weather, and non-drunk drivers tend to concentrate in California and Texas (A, B, and C). In contrast, during the winter months, such accidents increase notably in Florida (D, E, and F).

ing Flask [2], the mlxtend [3] library provided the rule generation algorithms, and for clustering, the Louvain algorithm was implemented using the networkx library. On the client side, the interface was developed with HTML and JavaScript. The visualizations were implemented using D3.js [4] and Leaflet [5]. The reorder.js library was used to reorder the rows in the *Attributes View* and *Heatmap View* components, based on the occurrences per time slice. In the present implementation, the algorithms for rule generation and clustering are fixed. Nevertheless, the preprocessing pipeline is modular, so replacing it is straightforward: users need only update the relevant module's function calls, provided the new algorithms emit output in the same format. All source code will be released on GitHub to ensure full reproducibility.

7. Case Studies

We present two case studies to demonstrate *STRive*'s capabilities for analyzing association rules.

7.1. Analyzing Seasonal Patterns in Vehicular Accidents

Our first case study explores factors related to fatal vehicular accidents. We use the Fatality Analysis Reporting System (FARS) dataset [36], which contains detailed records of fatal accidents in the United States. Given the dataset's extensive nature, we select only relevant attributes and rename them for clarity. The list of attributes and their descriptions is provided in Appendix A.

We analyze data from 2016 to 2022. Our goal is to understand how weather, public illumination, and driver intoxication (attributes: `Weather`, `Lit Condition`, `Drunk`) relate to fatal accidents. After removing null and 'Not Reported' values, we split the data into monthly segments, each with approximately 2,000 accidents. We set the minimum support at 0.05, minimum lift at 1.05, and cluster resolution at 1. The resulting clusters contain 2, 4, and 6 rules. Figure 2.B highlights the two clusters with more occurrences (Cluster 1 and Cluster 2). Cluster 1 involves clear weather (`Weather:Clear`), daylight (`LitCond:Day`), and non-drunk drivers (`Drunk:No`). Cluster 2 includes clear weather, dark conditions (`LitCond:Dark-Lit` and `LitCond:Dark-Unlit`), and non-drunk drivers (`Drunk:No`).

These associations help analyze the role of lighting conditions in fatal accidents. The *Cluster Scatterplot* (Figure 2.E) indicates both clusters have reliable associations (mean lift > 1). Cluster 1 has a higher average support than Cluster 2, meaning its described conditions occur more frequently. The *Heatmap View* (Figure 2.C) shows monthly occurrences. Cluster 1 peaks from May to August, while Cluster 2 increases from October to January. Although Cluster 1 has generally higher occurrence rates, both clusters show clear seasonal peaks. The remaining rows show temporal behaviors involving drunk drivers (`Drunk:Yes`). The significant difference between Clusters 1 and 2 is lighting conditions. This suggests that lighting has a greater influence on fatal accidents than driver intoxication — the remaining clusters involving similar conditions and drunk drivers have fewer occurrences. We further analyze individual rules within these clusters.

This indicates a spatial shift in the distribution of rules.

[2] https://flask.palletsprojects.com
[3] https://github.com/rasbt/mlxtend
[4] https://d3js.org
[5] https://leafletjs.com

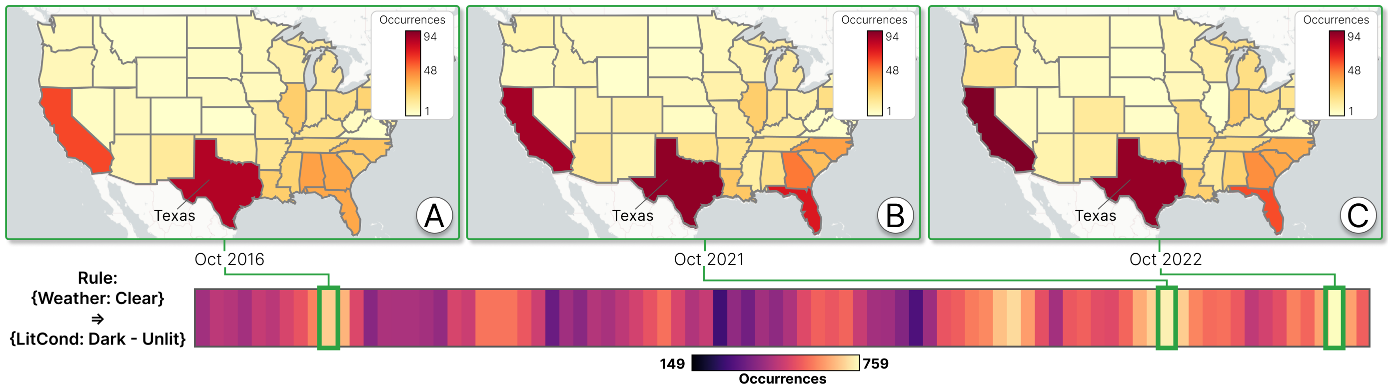

Fig. 5. Rule from Cluster 2 describing nighttime accidents under unlit conditions. The state of Texas consistently appears as one of the most affected regions, with notable peaks in incident counts during October (A, B, and C).

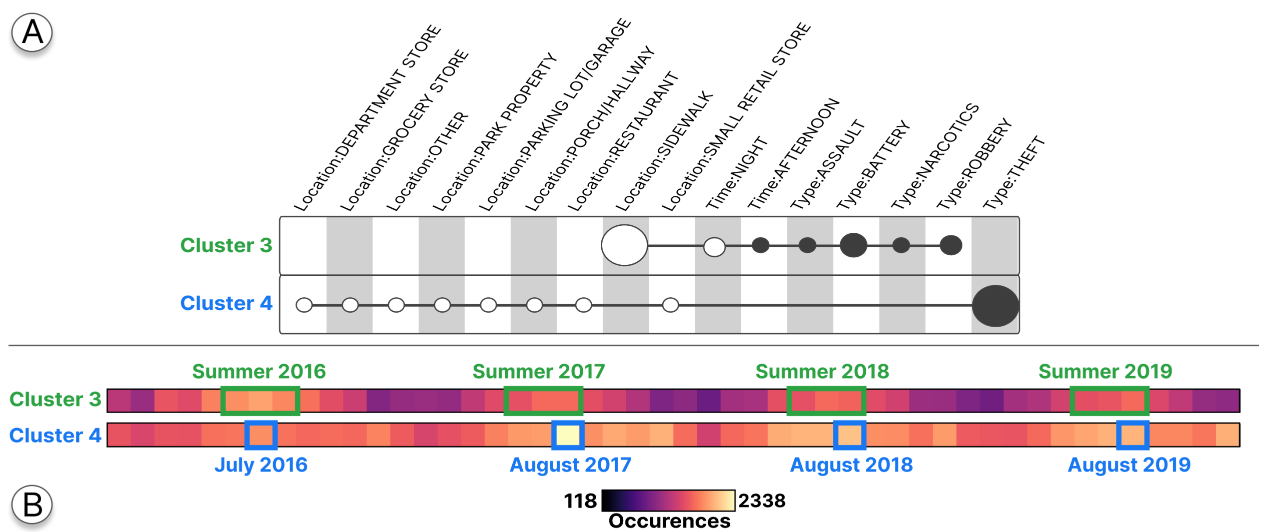

Fig. 6. Two clusters selected for analysis. (A) Cluster composition. (B) Cluster temporal behavior.

Selecting Cluster 1, we examine daytime accidents (`Lit Condition:Daylight`). We set the tool to *rule-level* mode by clicking on the *Rules* button (Figure 2, F). The components in the *rule-level* mode mirror their counterparts in the *cluster-level* mode, allowing us to visually identify the rule structure and inspect their temporal and spatial behavior. Figure 4 illustrates the temporal pattern of the rule {LitCond:Day, Weather:Clear} ⇒ {Drunk:No}, confirming a summer peak. Filtering data for June, July, and August, the *Map View* reveals that most accidents occur in California and Texas (Figure 4.A–C). Conversely, selecting December and January shifts the highest incidence to Florida (Figure 4.D–F). This shift aligns with reports indicating an increase in winter travel to Florida due to milder weather, which increases the flux of people traveling to the state, causing the number of vehicle accidents to rise [37, 38, 39]. Next, we explore nighttime accidents from Cluster 2. Figure 5 shows the rule {LitCond:Dark-Unlit, Weather:Clear} ⇒ {Drunk:No}. Accidents peak consistently in October. Spatial analysis reveals that Texas consistently has high accident rates this month (Figure 5.A–C). This observation aligns with official reports stating that October is particularly deadly for pedestrians in Texas due to reduced daylight [40].

7.2. Analyzing Crimes in Chicago

Our second case study explores factors related to crime incidents in Chicago. The data comes from the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system [41]. This dataset includes date, location, crime type, and arrest status. For our analysis, we select records from 2016 to 2019. We focus on examining the relationship between crime types and locations (attributes: `Type` and `Location`) and the timing of incidents (attribute: `Time`). After preprocessing to remove null and "Not Reported" entries, we segment data monthly, resulting in approximately 17,000 to 25,000 incidents per month. Given the large dataset, we set minimum *support* and *lift* thresholds at 0.01 and 1.5, respectively, to capture meaningful associations. We also set the clustering resolution parameter at 2.5 to achieve more refined clusters.

Figure 6.A displays the resulting clusters. Our analysis focuses on two clusters, specifically Cluster 3 and Cluster 4. Cluster 3 includes rules involving multiple crime types that

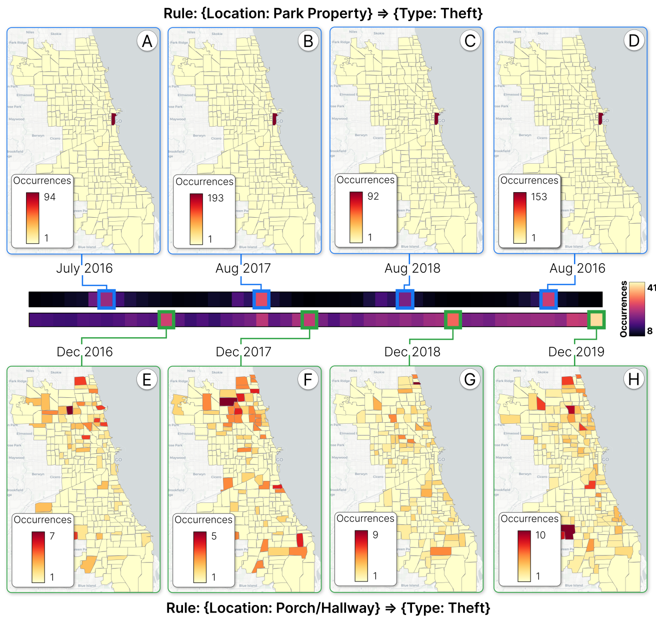

Fig. 7. Two example rules from Cluster 4. The top row (A, B, C, and D) shows the spatial distribution of thefts occurring in parks. The bottom row (E, F, G, and H) shows the spatial distribution of thefts on residential porches and in hallways.

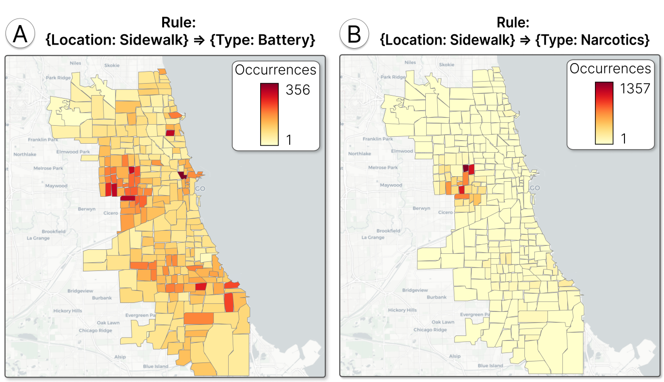

Fig. 8. Spatial distribution of different crime types. (A) Map showing the distribution of battery incidents. (B) Map showing the distribution of narcotics-related offenses.

occur primarily on sidewalks (`Location:SIDEWALK`), mostly during the afternoon (`Time:afternoon`) and nighttime hours (`Time:night`). Cluster 4 consists exclusively of theft incidents (`Type:THEFT`) across various locations (see attribute `Locations`. These clusters stand out due to their distinct seasonal trends (see Figure 6.B). Cluster 3 has the most incidents and exhibits clear seasonality, with peaks in the summer. Cluster 4 also demonstrates seasonality, though weaker, with notable spikes.

We first examine Cluster 4 to understand the seasonal spikes. The cluster contains rules describing thefts at locations such as parks, residential porches and hallways, and grocery stores (Figure 6.A). A detailed temporal analysis (Figure 7) reveals that thefts on porches and hallways peak during December and January, likely due to increased online shopping and unattended deliveries during the holidays. Conversely, thefts in parks spike notably in July and August, despite being relatively rare. Spatial analysis identifies the Loop district, specifically Beat 0114, as the primary location for these park thefts (Figure 7.A–D). Grant Park, within Beat 0114, hosts the Lollapalooza festival during peak theft periods, explaining the spikes in theft incidents during these months (July 28–31, 2016; August 3–6, 2017; August 2–5, 2018; August 1–4, 2019).

Next, we examine the rules in Cluster 3, with a focus on sidewalk-related incidents. The cluster prominently features battery incidents on sidewalks (see rule on Figure 8.A). Examining narcotics offenses on sidewalks reveals high incident concentrations in West Garfield Park, Humboldt Park, and Austin (see rule on Figure 8.B). These areas are known for gang activity and drug-related issues, aligning with reports indicating they account for approximately 30% of Chicago's emergency drug-related calls [42].

The different treatments of time and location in the two case studies stem from differences in the underlying datasets. In the crime dataset, location is already provided as a categorical attribute (e.g., "sidewalk"), and time can be discretized into broad periods such as morning, afternoon, and evening; both attributes can therefore be included in rule generation. By contrast, the vehicular-accident dataset offers richer contextual variables — such as speed limit, road type, and light condition — while its geographic information is limited to continuous coordinates, which are not suitable for direct ARM. Consequently, the vehicular-accident case study concentrates on those categorical road-context attributes instead. In both studies, spatial and temporal information is employed mainly in the analysis phase to contextualize the resulting rules.

8. Expert Feedback

To further evaluate our system, we conducted semi-structured interviews with domain experts, focusing on its usability and usefulness. This section describes the participants' backgrounds, the interview methodology, and the main findings.

8.1. Participants' Backgrounds

We invited two computer science experts, each with over 20 years of experience in data analysis. One specializes in data visualization, and the other in data mining; both have extensive hands-on experience with ARM in their research. Their combined expertise, therefore, spans the principal areas on which our system is built.

8.2. Interview Process

We conducted separate one-hour semi-structured interviews with each expert. We began by introducing the class of datasets targeted by STRive and outlining the main steps of the methodology—rule generation, rule clustering, and metric computation. Next, we demonstrated the interface in detail, explaining how each component supports insight discovery. We then presented our case studies and provided a link to a live demo populated with the same data, allowing the experts to explore the tool hands-on. Finally, each expert completed a questionnaire that captured their views on the methodology, the system's usefulness and usability, and potential areas for improvement.

8.3. Results

Below, we summarize the main feedback obtained during the interviews.

Methodology. When asked, *"What is your opinion of the proposed methodology for mining spatiotemporal patterns?"*, both experts described the approach as compelling and pointed to the accompanying use cases as persuasive examples. One expert observed that the current framework could not capture more complex temporal structures and recommended extending it to support sequential rule mining [43]. The other expert noted that analysts rarely begin exploration with the entire rule set produced by an ARM algorithm; instead, they typically start with a specific hypothesis — for example, requiring a given attribute to appear in the consequent — and then investigate the factors that lead to that outcome. Although STRive does not yet support this workflow, it could be implemented by filtering the generated rules for user-specified attribute values. We plan to incorporate this capability in the next version of the system.

Usability. In response to the questions *"Was the system easy to use?"* and *"Do you believe people will learn how to use this system quickly?"*, both experts reported that the interface is straightforward and the visualizations are intuitive. They cautioned, however, that their considerable experience with association-rule mining may influence this impression — and, in one case, specifically with ARM visualization. When asked whether setting the initial parameters or selecting attributes for rule generation was difficult, both replied that the process was simple. One expert noted that the evaluation datasets contained relatively few attributes, making the checkbox selection effective; with larger datasets, the same approach might become a usability bottleneck.

Usefulness. In response to *"Do you consider STRive a useful tool, and why?"*, both experts affirmed its utility. One highlighted that STRive is especially compelling when patterns emerge in specific regions during specific time frames, where it could support decision-making — for instance, helping policymakers address pedestrian incidents in October in Texas. He remarked: "*The main benefit I see is STRive's capacity to abstract data into simple, interpretable patterns that are otherwise very difficult to extract. In scenarios where such abstraction is acceptable, the tool can indeed be very useful.*"

The second expert echoed this view and emphasized the value of visualizing incidents that satisfy the rules across regions and over time. He expressed interest in being able to switch the displayed metrics (e.g., showing *lift* instead of the number of occurrences) in both the *Heatmap* and *Map views*, as well as in supporting multiple temporal resolutions (e.g., weekly) in the *Heatmap*.

Both experts felt that all components are important for the analyses shown in the case studies; none was judged more critical than the others. They specifically praised the LLM explanation panel, noting that it substantially enhances the interpretability of the discovered patterns.

Limitations. Asked about potential shortcomings, the first expert observed that datasets with hundreds of attributes—common in spatiotemporal analysis—could overwhelm the interface and that rules containing many attributes may impose a high cognitive load. He stressed, however, that these challenges are intrinsic to association-rule mining rather than unique to STRive. The second expert reiterated scalability concerns and pointed out the potential loss of interpretability when clustering rules. He acknowledged, though, that clustering is necessary to manage the large rule sets and to enable focused, drill-down exploration.

Overall, the expert evaluation confirmed STRive's ease of use and practical value while highlighting its limitations and suggesting directions for future improvement.

9. Discussion and Future Work

As shown in Section 7, *STRive* enables users to effectively identify and analyze spatiotemporal data patterns. This section discusses the current workflow, limitations, and future directions for improving *STRive*.

Use of clustering algorithms: We currently use the Louvain algorithm to create coherent rule clusters. After evaluating multiple methods, Louvain demonstrated superior results in terms of cluster quality, parameter flexibility, and computational efficiency. However, Louvain is non-deterministic, meaning different runs may produce varying clusters. As the clustering step plays a major role in our workflow, this variability can hinder reproducibility and complicate analyses. Future implementations could include alternative clustering methods with user-defined parameters. Additionally, extending clustering to incorporate spatial and temporal rule distributions could provide more comprehensive insights.

Interactive rule modification: Currently, *STRive* supports rule comparison based on spatial and temporal attributes. Allowing users to add or remove attributes within rules dynamically would provide more direct rule manipulation. This feature could guide initial exploration and reveal additional patterns derived from user-modified rules.

Computational Scalability: Our current implementation can handle moderately sized datasets with diverse spatiotemporal attributes; however, it is not specifically optimized for "big-data" workloads. Computational bottlenecks primarily arise in the rule-extraction and clustering phases. These bottlenecks can be mitigated in several ways, as both tasks are well-studied. For instance, more sophisticated algorithms for rule mining and clustering could significantly improve STRive's scalability [44, 45, 46].

Visual Scalability: The present implementation of the *Heatmap* and *Attributes views* does not scale well as the number of time slices or data attributes grows. This limitation can be alleviated by incorporating filtering — for example, focusing on specific periods or attribute subsets — and by adopting focus–plus–context techniques [47]. We plan to integrate these strategies in the next version of the system.

Visual Encoding Enhancements: Further work on visual encodings may help improve spatial interpretation by using glyphs

Conversation 9.1:

User: There was an increase in thefts in parks on the following dates: July 2016, August 2017, August 2018, and August 2019 in the Loop district in Chicago. Do you know any possible causes?

ChatGPT: The increases in thefts in Chicago's Loop district parks during July 2016, August 2017, August 2018, and August 2019 can be attributed to several factors:
1. Large Summer Events: Major festivals, such as Lollapalooza, held annually in late July or early August in Grant Park, draw substantial crowds...
2. Seasonal Crime Patterns: Warmer summer months often correlate with increased crime rates...
3. Resource Allocation Challenges: Large events require significant police presence, which can strain resources and potentially leave other areas more vulnerable...
4. Public Transit Crime: There have been instances of increased crime on Chicago's public transit during summer months...
These factors combined likely contributed to the observed increases in park thefts during the specified periods.

User: Can you provide the exact dates and locations of the Lollapalooza festival?

ChatGPT: Lollapalooza is an annual music festival held in Chicago's Grant Park. Here are the exact dates for the years you inquired about:

- 2016: July 28–31.
- 2017: August 3–6.
- 2018: August 2–5.
- 2019: August 1–4.

Each year, the festival spans four days, attracting large crowds to Grant Park in Chicago, Illinois.

to show the distribution of attributes within each region on the map. This could make it easier to spot local variations that are not visible through aggregate values alone.

Pattern explanation support: While our system identifies significant patterns, fully understanding their underlying causes often requires external context. Leveraging large language models (LLMs), such as ChatGPT-4o, proved beneficial for obtaining explanations in our first case study. For example, we asked about theft increases in Chicago parks during specific dates. ChatGPT-4o effectively identified factors such as major events and seasonal crime patterns contributing to this phenomenon. A partial interaction transcript appears in Conversation 9.1. While *STRive* already utilizes LLMs to generate contextual explanations and surface relevant external information, future enhancements could offer even more sophisticated guidance — for instance, helping analysts interpret visualizations and automatically highlighting potential patterns revealed by the system.

Rule causality: Although association rules can appear to express causal relationships, they in fact capture only frequent co-occurrences in the data. This limitation is inherent to ARM, so additional analysis—such as the interactive exploration offered by our system—is essential for identifying the rules that matter most. Users can refine the search by specifying attributes that must appear in the antecedent, the consequent, or both, thereby steering the mining process toward patterns that align with domain knowledge. Moreover, rule directionality plays a minor role in our workflow because we use *lift* — a symmetric metric independent of rule direction — as the principal filtering criterion.

10. Conclusion

This paper introduces *STRive*, a visual analytics system that utilizes Association Rule Mining (ARM) as a guidance mechanism for exploring spatiotemporal data. *STRive* leverages the FP-Growth algorithm to extract association rules and groups these rules into coherent clusters using the Louvain algorithm. The system provides users with interactive linked visualizations by computing rule metrics over time and spatial distributions.

Our work demonstrates that ARM effectively reveals meaningful patterns, significantly aids users in navigating complex spatiotemporal datasets, and inherits the ease of interpretation of these patterns. This feature is very much required in modern systems. The case studies validate the practical usefulness of ARM-based guidance, illustrating its ability to uncover insightful and interpretable relationships in real-world scenarios.

In future work, we plan to conduct a user study to formally evaluate the usability and effectiveness of *STRive*. Additionally, we aim to integrate large language models (LLMs) to automatically generate explanations for the discovered patterns, further enhancing the system's interpretability and utility. Finally, our system focuses on spatiotemporal *event* data [48]. Other forms (such as trajectories) pose different analytical tasks and pattern types, which may call for sequential or trajectory-aware rule-mining techniques [43]. Extending *STRive* to support these data, therefore, represents a promising avenue for future research.

Acknowledgments

To be added in the camera-ready version.

References

[1] Hamdi, A, Shaban, K, Erradi, A, Mohamed, A, Rumi, SK, Salim, FD. Spatiotemporal data mining: a survey on challenges and open problems. Artificial Intelligence Review 2022;:1–48.

[2] Kumar, R, Bhanu, M, Mendes-Moreira, Ja, Chandra, J. Spatio-temporal predictive modeling techniques for different domains: a survey. ACM Comput Surv 2024;57(2).
[3] Zhang, Q, Wang, H, Long, C, Su, L, He, X, Chang, J, et al. A survey of generative techniques for spatial-temporal data mining. arXiv preprint arXiv:240509592 2024;.
[4] Pérez-Messina, I, Ceneda, D, Miksch, S. Enhancing visual analytics systems with guidance: A task-driven methodology. Computers & Graphics 2024;125:104121.
[5] Agrawal, R, Imieliński, T, Swami, A. Mining association rules between sets of items in large databases. SIGMOD '93; New York, NY, USA; 1993, p. 207–216.
[6] Liu, S, Maljovec, D, Wang, B, Bremer, PT, Pascucci, V. Visualizing high-dimensional data: Advances in the past decade. IEEE TVCG 2016;23(3):1249–1268.
[7] Mota, R, Ferreira, N, Silva, JD, Horga, M, Lage, M, Ceferino, L, et al. A comparison of spatiotemporal visualizations for 3d urban analytics. IEEE TVCG 2022;29(1):1277–1287.
[8] Ceneda, D, Gschwandtner, T, May, T, Miksch, S, Schulz, HJ, Streit, M, et al. Characterizing guidance in visual analytics. IEEE TVCG 2016;23(1):111–120.
[9] Collins, C, Andrienko, N, Schreck, T, Yang, J, Choo, J, Engelke, U, et al. Guidance in the human–machine analytics process. Visual Informatics 2018;2(3):166–180.
[10] Sperrle, F, Ceneda, D, El-Assady, M. Lotse: A practical framework for guidance in visual analytics. IEEE TVCG 2023;29(1):1124–1134.
[11] Han, W, Schulz, HJ. Providing visual analytics guidance through decision support. Information Visualization 2023;22(2):140–165.
[12] Doraiswamy, H, Ferreira, N, Damoulas, T, Freire, J, Silva, C. Using topological analysis to support event-guided exploration in urban data. IEEE TVCG 2014;20(12):2634–2643.
[13] Valdivia, P, Dias, F, Petronetto, F, Silva, C, Nonato, L. Wavelet-based visualization of time-varying data on graphs. In: Proc. of the 2015 IEEE Conf. on Visual Analytics Science and Technology. Chicago, USA; 2015, p. 1–8.
[14] Liu, D, Xu, P, Ren, L. TPFlow: Progressive partition and multidimensional pattern extraction for large-scale spatio-temporal data analysis. IEEE TVCG 2019;25(1):1–11.
[15] Varu, R, Christino, L, Paulovich, FV. Armatrix: An interactive item-to-rule matrix for association rules visual analytics. Electronics 2022;11(9).
[16] Menin, A, Cadorel, L, Tettamanzi, A, Giboin, A, Gandon, F, Winckler, M. Arviz: Interactive visualization of association rules for rdf data exploration. In: 2021 25th International Conference Information Visualisation (IV). 2021, p. 13–20.
[17] Wang, F, Li, W, Wang, S, Johnson, CR. Association rules-based multivariate analysis and visualization of spatiotemporal climate data. ISPRS International Journal of Geo-Information 2018;7(7).
[18] Fister, I, Fister, I, Fister, D, Podgorelec, V, Salcedo-Sanz, S. A comprehensive review of visualization methods for association rule mining: Taxonomy, challenges, open problems and future ideas. Expert Systems with Applications 2023;233.
[19] Valdivia, P, Buono, P, Plaisant, C, Dufournaud, N, Fekete, JD. Analyzing dynamic hypergraphs with parallel aggregated ordered hypergraph visualization. IEEE TVCG 2021;27(1):1–13.
[20] Oliver, P, Zhang, E, Zhang, Y. Scalable hypergraph visualization. IEEE TVCG 2024;30(1):595–605.
[21] Fischer, MT, Frings, A, Keim, DA, Seebacher, D. Towards a survey on static and dynamic hypergraph visualizations. In: 2021 IEEE Visualization Conference (VIS). 2021, p. 81–85.
[22] Zhang, C, Zhang, S. Association rule mining: models and algorithms. Springer; 2002.
[23] Nath, B, Bhattacharyya, DK, Ghosh, A. Incremental association rule mining: a survey. WIREs Data Mining and Knowledge Discovery 2013;3(3):157–169.
[24] Kisilevich, S, Mansmann, F, Nanni, M, Rinzivillo, S. Spatio-temporal clustering. Boston, MA: Springer US. ISBN 978-0-387-09823-4; 2010, p. 855–874.
[25] Momeni Kho, S, Pahlavani, P, Bigdeli, B. Classification and association rule mining of road collisions for analyzing the fatal severity, a case study. Journal of Transport & Health 2021;23:101278.
[26] Lakshmi, K, Vadivu, G. Extracting association rules from medical health records using multi-criteria decision analysis. Procedia Computer Science 2017;115:290–295.
[27] Diaz-Garcia, JA, Ruiz, MD, Martin-Bautista, MJ. Non-query-based pattern mining and sentiment analysis for massive microblogging online texts. IEEE Access 2020;8:78166–78182.
[28] Agrawal, R, Srikant, R. Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases. VLDB '94; 1994, p. 487–499.
[29] Han, J, Pei, J, Yin, Y. Mining frequent patterns without candidate generation. ACM sigmod record 2000;29(2):1–12.
[30] Brin, S, Motwani, R, Ullman, JD, Tsur, S. Dynamic itemset counting and implication rules for market basket data. SIGMOD '97; New York, NY, USA: Association for Computing Machinery; 1997, p. 255–264.
[31] Selvi, CK, Tamilarasi, A. An automated association rule mining technique with cumulative support thresholds. Int J Open Problems in Compt Math 2009;2(3):12.
[32] Altaf, W, Shahbaz, M, Guergachi, A. Applications of association rule mining in health informatics: a survey. Artificial Intelligence Review 2017;47:313–340.
[33] Hahsler, M, Karpienko, R. Visualizing association rules in hierarchical groups. Journal of Business Economics 2017;87:317–335.
[34] Fister Jr, I, Fister, I. Association rules over time. In: Frontiers in nature-inspired industrial optimization. Springer; 2021, p. 1–16.
[35] Blondel, VD, Guillaume, JL, Lambiotte, R, Lefebvre, E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008;2008(10):P10008.
[36] National Highway Traffic Safety Administration, . Fatality analysis reporting system (fars). https://www.nhtsa.gov/research-data/fatality-analysis-reporting-system-fars; n.d. Accessed: 2025-03-20.
[37] White, L, Murphey, . Are There More Car Accidents During Winter Visitor Season? https://www.lwmpersonalinjurylawyers.com/blog/are-there-more-car-accidents-during-winter-visitor-season/; n.d. Accessed: 2025-03-20.
[38] Logistics, S. Do car accidents increase during 'snowbird' migration in florida? https://www.sakaemlogistics.com/do-car-accidents-increase-during-snowbird-migration-in-florida/; 2024. Accessed: 2025-03-20.
[39] Perez, M. Reasons south florida car accidents involving snowbirds occur. https://www.mallardperez.com/library/why-sarasota-car-crashes-spike-when-snowbirds-flock-to-florida.cfm; n.d. Accessed: 2025-03-20.
[40] Texas Department of Transportation, . October is deadly month for texas pedestrians. https://www.txdot.gov/about/newsroom/statewide/2023/october-is-deadly-month-for-texas-pedestrians.html; n.d. Accessed: 2025-03-20.
[41] Chicago Police Department, . Crimes - 2001 to present. https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2/about_data; 2025. Accessed: 2025-03-20.
[42] Institue, IPH. Responding to opioid overdoses and treating opioid use disorder. https://www.chicago.gov/content/dam/city/depts/cdph/CDPH/Healthy%20Chicago/LA_Full_Report_IPHI.pdf; 2023. Accessed: 2025-03-20.
[43] Huang, Y, Zhang, L, Zhang, P. A framework for mining sequential patterns from spatio-temporal event data sets. IEEE Transactions on Knowledge and data engineering 2008;20(4):433–448.
[44] Zhao, Z, Jian, Z, Gaba, GS, Alroobaea, R, Masud, M, Rubaiee, S. An improved association rule mining algorithm for large data. Journal of Intelligent Systems 2021;30(1):750–762.
[45] Wang, HB, Gao, YJ. Research on parallelization of apriori algorithm in association rule mining. Procedia Computer Science 2021;183:641–647.
[46] Do, DH, Phan, THD. An improvement on the louvain algorithm using random walks. arXiv preprint arXiv:240308313 2024;.
[47] Cuenca, E, Sallaberry, A, Wang, FY, Poncelet, P. Multistream: A multiresolution streamgraph approach to explore hierarchical time series. IEEE TVCG 2018;24(12):3160–3173.
[48] Kisilevich, S, Mansmann, F, Nanni, M, Rinzivillo, S. Spatio-temporal clustering. Springer; 2010.

Appendix A. Fatality Analysis Reporting System (FARS) dataset

The FARS dataset includes a wide range of attributes related to different aspects of fatal vehicle accidents. In our analysis, we retain only a small subset of these attributes. Table A.1 presents the selected attributes and their descriptions.

Attribute	Description
Id	Unique identifier for the accident.
Date	Date of the accident.
State	State where the accident happened.
Drunk	Indicates whether one of the involved drivers was drunk.
Lanes	Number of lanes of the road where the accident occurred.
Speed	Maximum speed limit of the road where the accident occurred.
Surface	Status of the road surface at the time of the accident (e.g. Dry or Wet).
Lit Condition	Status of the public illumination at the time of the accident (e.g. Dark - Lighted, Dark - Not Lighted or Daylight).
Pavement	Type of pavement of the road where the accident occurred (e.g. Asphalt).
Level	Inclination of the road where the accident occurred (e.g. Level or Uphill).
Align	Alignment of the road where the accident occurred (e.g. Straight or Curve).
Junction	Accident location in relation to its proximity to junction or interchange areas (e.g. Intersection or Driveway Access).
Weather	Atmospheric conditions at the time of the accident (e.g. Clear or Rain).
Season	Season of the year when the accident occurred (e.g. Summer or Winter).
Region	Region where the accident occurred (e.g. Midwest or South).

Table A.1. Attributes found in the FARS dataset.

It is important to note that attributes such as Season and Region are not originally present in the data; they were derived from the Date and State attributes and added by us.

Appendix B. Prompt Used for LLM Explanations

Appendix B.1 outlines the prompt used to generate explanations. Note that the text in **bold** highlights elements specific to the dataset in use. This prompt differs from the one in the Discussion section, as it is automatically generated from raw data, whereas the other was manually written after identifying the main pattern.

Prompt Appendix B.1

Here we have an association rule, describing a pattern found in a **vehicular accidents dataset**. Each element in the antecedent or consequent is an attribute–value pair.

Antecedent: {**LitCond:Daylight**, **Weather:Clear**}
Consequent: {**Drunk:No**}

Location:Florida
2016-01: 27
2016-02: 44

⋮

Location:Texas
2016-01: 61
2016-02: 52

⋮

Tasks:
1. Identify trends both in time and space.
2. Formulate a couple specific hypotheses explaining the identified behavior.
3. Search the internet for information sources to validate your hypothesis.
3. Use the Google Search tool to find specific news articles, reports, and studies
4. Provide actual working URLs, not placeholder URLs.
If no information was found, just return the hypothesis and description.
Output the findings as a JSON list of dictionaries with the following format (strictly valid JSON only):
{
”hypothesis”: ””,
”description”: ””,
”sources”: []
}
Output each source as a JSON dictionary with title and URL:
{
”title”: ””,
”url”: ””
}