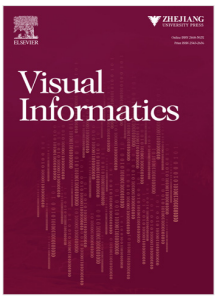
Journal Pre-proof



DFCollection: Scalable and hierarchical visual exploration of data facts

Yikai Li, Xiaolin Wen, Qiaomu Shen, Yong Wang

PII: S2468-502X(26)00072-0
DOI: https://doi.org/10.1016/j.visinf.2026.100376
Reference: VISINF 100376

To appear in: *Visual Informatics*

Received date : 4 January 2026
Revised date : 5 June 2026
Accepted date : 28 October 2026

Please cite this article as: Y. Li, X. Wen, Q. Shen et al., DFCollection: Scalable and hierarchical visual exploration of data facts. *Visual Informatics* (2026), doi: https://doi.org/10.1016/j.visinf.2026.100376.

This is a PDF of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability. This version will undergo additional copyediting, typesetting and review before it is published in its final form. As such, this version is no longer the Accepted Manuscript, but it is not yet the definitive Version of Record; we are providing this early version to give early visibility of the article. Please note that Elsevier's sharing policy for the Published Journal Article applies to this version, see: https://www.elsevier.com/about/policies-and-standards/sharing#4-published-journal-article. Please also note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2026 The Author(s). Published by Elsevier B.V. on behalf of Zhejiang University and Zhejiang University Press.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Journal Pre-proof

DFCollection: Scalable and Hierarchical Visual Exploration of Data Facts

Yikai Li[a], Xiaolin Wen[a], Qiaomu Shen[b], Yong Wang[a,*]

[a]Nanyang Technological University, 50 Nanyang Avenue, Singapore, 639798, Singapore
[b]Beijing Institute of Technology, No. 6 Jinfeng Road, Tangjiawan, Zhuhai, 519088, Guangdong, China

Abstract

Exploratory data analysis (EDA) aims to uncover meaningful data facts that support understanding and decision-making. However, existing automated EDA systems still face challenges in scalability, hierarchical exploration, and similarity computation across heterogeneous subspaces. To address these issues, we introduce the concept of fact collections, which organize data facts according to shared subspace dimensions for scalable and hierarchical exploration. We develop DFCollection, a scalable and hierarchical visual analytics system for data exploration. The system comprises a Data Preparation module for fact extraction and similarity computation and a Data Exploration module with three coordinated views: the Collection View for an overview of collections, the Collection-Fact View for hierarchical exploration, and the Fact View for similarity visualization. The Fact Gallery additionally supports recording and reviewing interesting findings. We evaluate DFCollection through two usage scenarios and a user study with 13 participants. The results demonstrate that DFCollection effectively improves the scalability, interpretability, and usability of data fact exploration.

Keywords: Exploratory Data Analysis, Visual Analytics, Data Fact, Fact Collection

1. Introduction

Exploratory Data Analysis (EDA) focuses on uncovering interesting facts and gaining knowledge from data (Tukey, 1977; Jebb et al., 2017; Jankun-Kelly et al., 2007). Data facts obtained through EDA play a crucial role in facilitating further analysis and supporting decision-making. In recent years, many automatic and interactive EDA tools have been developed, some of which advance beyond identifying individual data facts to exploring the relationships among them by incorporating similarity metrics and analyzing overall distributions (Chen et al., 2024; Wu et al., 2025; Jiang et al., 2025).

Despite the rapid development of automated EDA tools, we identify three major challenges that remain. Firstly, scalability issues persist. Existing tools, such as Calliope-Net (Chen et al., 2024), perform effectively only when handling a limited number of data facts, to avoid visual clutter and preserve the readability of individual glyphs. Other tools, like FactExplorer (Jiang et al., 2025), attempt to incorporate more data facts but consequently produce visually overwhelming results. This indicates that, although datasets may contain numerous data facts, current tools can only effectively present a limited subset. Secondly, supporting exploration over large numbers of data facts remains challenging. Prior studies suggest that hierarchical organizations can facilitate navigation in complex information spaces (Hornof, 2004; Müller et al., 2017). However, most existing tools present data facts in flat similarity-based layouts, making it difficult to explore relationships at different levels of granularity. Our literature review indicates that hierarchical organization of data facts remains largely underexplored. Thirdly, comparing relationships among large numbers of data facts also remains challenging. Since each data fact is associated with a subspace defined by filters, similarity computation depends not only on fact properties but also on the structures of their corresponding subspaces. Existing approaches (Dennig et al., 2024) mainly rely on simple overlap-based similarity metrics, which may become difficult to interpret when facts originate from subspaces with different dimensionalities.

To address these challenges, we present **DFCollection**, a novel analytics system for scalable and hierarchical visual exploration of **D**ata **F**acts through grouping them into **Collection**s, which can accommodate a substantially greater number of data facts while maintaining scalability compared to existing systems. Specifically, we introduce the concept of fact collection, which groups data facts sharing the same subspace dimensionality into a single collection, thereby improving scalability, enabling hierarchical exploration, and ensuring meaningful subspace similarity computation. Unlike traditional group-by or Online Analytical Processing (OLAP) operations, fact collections organize automatically extracted analytical facts by shared subspace semantics. DFCollection groups facts with the same filter attribute set, enabling consistent similarity computation and hierarchical exploration, transforming large, isolated facts into structured units that support scalable navigation, overview-to-detail exploration, and interpretable comparison.

The framework of DFCollection comprises two primary modules: Data Preparation and Data Exploration. In the Data Preparation module, DFCollection employs a customized au-

*Corresponding author.
Email addresses: liyi0084@e.ntu.edu.sg (Yikai Li), xiaolin004@e.ntu.edu.sg (Xiaolin Wen), shenqm@bit.edu.cn (Qiaomu Shen), yong-wang@ntu.edu.sg (Yong Wang)

Journal Pre-proof

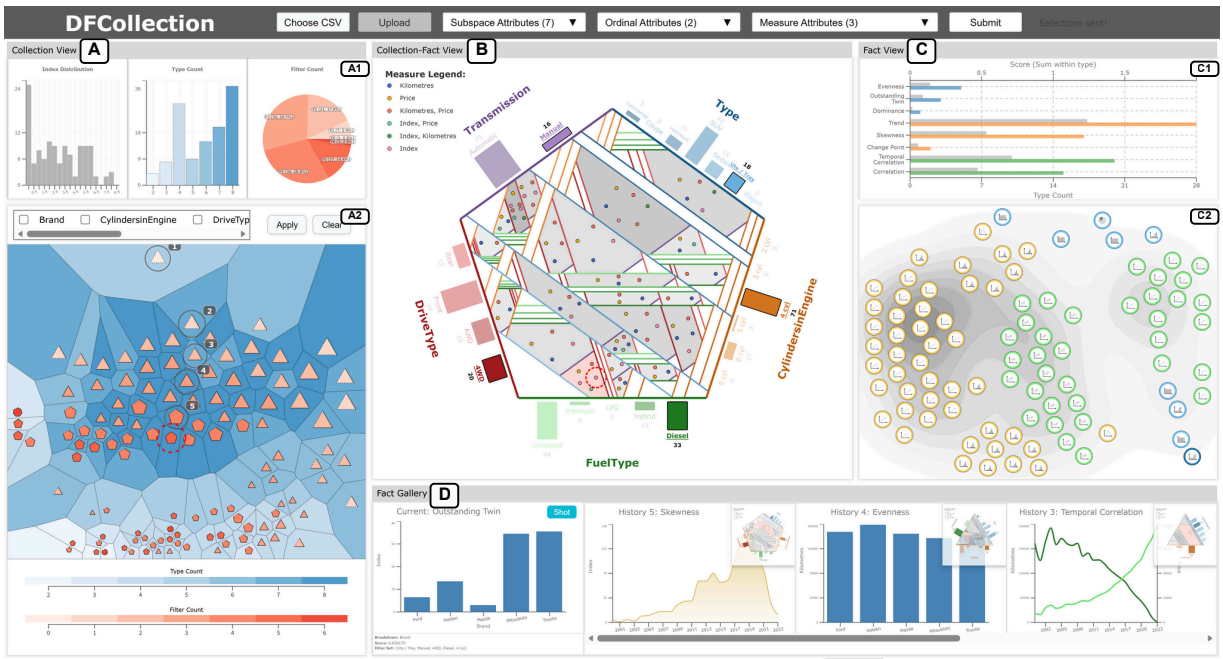


Figure 1: The interface of DFCollection. (A) The **Collection View** provides an overview of all collections, including statistical distributions of index, type count, and filter count (A1), and visualizes the overall spatial layout of fact collections (A2). (B) The **Collection–Fact View** reveals the hierarchical relationship between a collection and its contained facts. (C) The **Fact View** presents a summary of the facts within a selected collection, showing both the score distribution and type counts (C1), and visualizing their spatial layout based on similarity with score-density cues (C2). (D) The **Fact Gallery** displays details of the currently selected fact and records those marked as interesting during exploration, supporting the construction of personalized data stories.

tomated pipeline that extracts data facts from the dataset, organizes them into collections, and computes similarities both among individual facts and between collections. The Data Exploration module offers three coordinated visualization views to facilitate interactive exploration. The Collection View adopts a Voronoi-style layout to depict the spatial distribution of fact collections and assist users in selecting regions of interest. The Collection–Fact View reveals multidimensional relationships and hierarchical distribution of data facts within the selected collection, allowing users to explore data facts at multiple levels with different attribute focuses. The Fact View provides a detailed similarity-based layout of facts within the chosen collection and supports score-density identification. Additionally, DFCollection includes the Fact Gallery, enabling users to inspect individual fact details, document discovered patterns, and construct personalized data stories based on their exploration process. We present two usage scenarios and in-depth user interviews to demonstrate the effectiveness and usability of our system. The key contributions of this paper are as follows:

- We introduce the concept of *fact collection*, which groups data facts according to their subspaces to enable scalable and hierarchical exploration.
- We develop an interactive system with a complete framework comprising two modules, and publicly release both the system implementation and demo video at `https://github.com/lyk6666/DFCollection`.
- We adapt and extend existing visualization techniques to the context of data facts, enabling the illustration of their distributions, hierarchies, and relationships.
- We evaluate the effectiveness of our system through two usage scenarios and user interviews with 13 participants.

2. Related Work

In this section, we review the prior studies on automatic fact-driven tools, hierarchical and multidimensional visualization, and similarity metrics for categorical data.

2.1. *Fact-driven Exploration Tools*

Over the past decade, research on automatic fact-driven tools has evolved from analyzing raw data to structuring knowledge as data facts that capture meaningful patterns. Early systems such as Foresight (Demiralp et al., 2017) introduced fact classes and interactive ranking pipelines, while Top-K Insights (Tang et al., 2017) formalized a computational framework for identifying top-k facts. QuickInsights (Ding et al., 2019) proposed a unified fact formulation later integrated into Microsoft Power BI, and MetaInsight (Ma et al., 2021) expanded this by classifying facts into specific types with an efficient mining method.

Recent studies have increasingly explored intelligent visualization recommendation, automated visual insight mining, and fact-driven analytical workflows. Lian et al. (2025) provide a recent survey of visual insight mining, summarizing how visualization connects data and insights through tasks such as visualization generation, visualization recommendation, chart question answering, and insight generation. Building on this broader research landscape, LIDA (Dibia, 2023) uses LLMs to generate grammar-agnostic visualizations and infographics, while

Journal Pre-proof

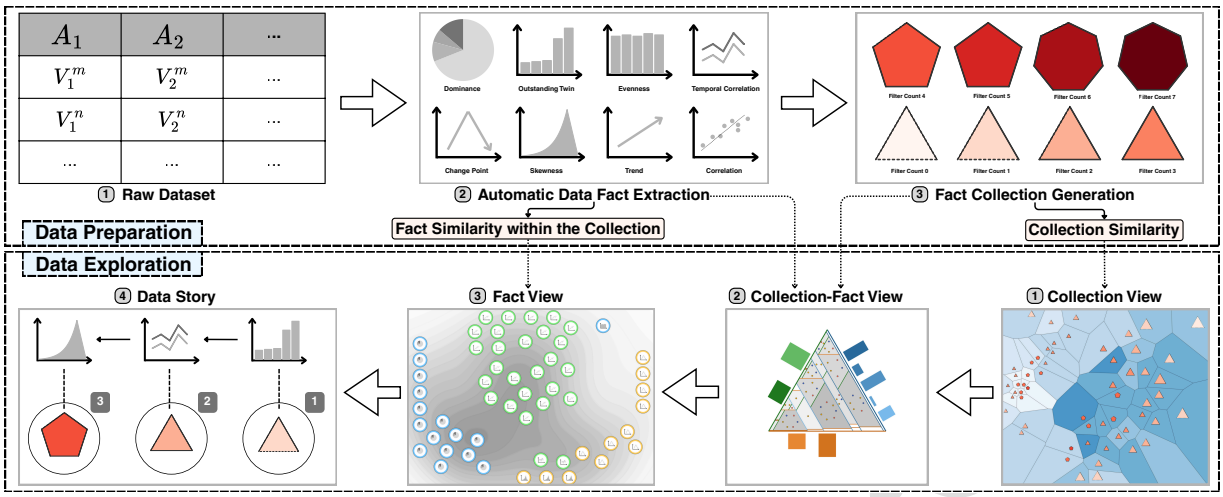


Figure 2: The DFCollection framework comprises two main modules: Data Preparation and Data Exploration. Data Preparation automatically extracts data facts from the dataset, groups them into collections, and computes similarities among them. Data Exploration enables users to navigate through the Collection View, Collection–Fact View, and Fact View, and to examine details via the Fact Gallery, allowing them to construct a data story based on their exploration history.

LLM4Vis (Wang et al., 2023) focuses on explainable visualization recommendation through natural-language interaction. DashChat (Shen et al., 2026) supports dashboard authoring with LLM-powered agents, DracoGPT (Wang et al., 2025) extracts visualization design preferences from LLMs, and Prompt4Vis (Li et al., 2024c) improves tabular data visualization through example mining and schema filtering. Explanova (Huang, 2026) further explores automatic insight discovery from $N \times M$ tables by combining explainable AI techniques with an LLM workflow. InsightPilot (Ma et al., 2023) integrates an LLM and fact engine to generate multi-step explorations, while InsightLens (Weng et al., 2025) employs an LLM-agent framework for automated fact recording and organization. InReAcTable (Aodeng et al., 2025) enhances story construction by linking insights structurally and semantically, and Calliope (Shi et al., 2021) uses a logic-driven Monte Carlo tree search to assess fact importance. InkSight (Lin et al., 2024) enables sketch-based subset selection for tailored fact recommendations, InsighTable (Li et al., 2025) applies deep reinforcement learning for hierarchical fact discovery, and InsightCube (Zhang et al., 2025) introduces a cube structure for efficient fact retrieval. Datashot (Wang et al., 2020) compiles salient facts into presentation-ready data shots. Similar to CoInsight (Li et al., 2024a), our system also allows users to record patterns and construct personalized data stories. Compared with current intelligent recommendation and LLM-powered systems, DFCollection addresses a complementary problem: the scalable organization of large heterogeneous fact spaces. While existing approaches mainly focus on chart generation, visualization recommendation, or conversational assistance, DFCollection organizes extracted data facts into semantically consistent analytical units for structured visual exploration. In addition, DFCollection offers advantages in determinism, transparency, and user control through interpretable hierarchical fact organization.

Recent work also extends beyond individual facts to explore their relationships and distributions. Calliope-Net (Chen et al., 2024) models facts as graph nodes with annotation routing, and CoInsight (Li et al., 2024a) visualizes fact relations through a nested graph. InsightMap (Wu et al., 2025) introduces similarity-based mapping, and FactExplorer (Jiang et al., 2025) integrates logical and visual representations for comprehensive overviews. Unlike these systems, which primarily focus on visualizing relationships or similarity distributions among facts, DFCollection emphasizes the scalable and hierarchical organization of related facts. By grouping facts according to shared filter attribute sets, our system supports interpretable similarity computation, transparent hierarchical structures, and user-controlled navigation across fact collections.

2.2. *Hierarchical and Multidimensional Visualization*

We have investigated a range of hierarchical and multidimensional visualization techniques that facilitate hierarchical exploration. Traditional techniques such as Treemap (Shneiderman, 1992) represent nested structures with packed rectangles whose areas encode quantitative values, while Sunburst charts (Stasko and Zhang, 2000) extend this idea into a radial form to emphasize hierarchical depth. Star plots (Chambers et al., 2018) arrange attributes along radial axes for quick comparative profiling, and Circle packing (Wang et al., 2006) illustrates grouping relationships through nested circles. Similarly, Chernoff faces (Chernoff, 1973) map multiple variables to facial features, providing an expressive yet abstract summary of complex data. Area-proportional set diagrams (Chow and Ruskey, 2003) visualize overlaps by matching geometric area to data magnitude, whereas Parallel Coordinates (Inselberg, 1985) align attributes along parallel axes to reveal inter-variable correlations.

Recent tools like Hi-D Map (Reza and Watson, 2019) partition data along ordered dimensions to enable structured navigation, and Categorical Data Map (Dennig et al., 2024) position items based on categorical similarity to highlight related patterns. Data Context Maps (Cheng and Mueller, 2016) unify items and attributes in a shared projection for integrated

Journal Pre-proof

reasoning. Parallel Hierarchies (Vosough et al., 2018) visualize multiple hierarchies and their cross-links. In addition, Star Coordinates (Kandogan, 2001) project high-dimensional attributes onto radial axes for interactive overview and exploration. Among these techniques, Hi-D Map (Reza and Watson, 2019) aligns closely with our system design, as we build on its visualization approach to develop the Collection–Fact View for depicting the hierarchical distribution of facts within a selected collection. We also adopt a Voronoi-style visualization for collections inspired by Categorical Data Map (Dennig et al., 2024).

2.3. Similarity Metrics for Categorical Data

Since the subspaces of data facts can be represented as combinations of filters across categorical attributes (Ding et al., 2019; Tang et al., 2017; Shi et al., 2021), we draw on prior studies of categorical data similarity and visualization to inform our subspace similarity computation. Similarity metrics for categorical data have been systematically examined in prior evaluations (Sulc and Rezanková, 2019; Boriah et al., 2008). Common approaches include Goodall's index (Goodall, 1966), which emphasizes matches on rare categories. Lin's information-theoretic measure (Lin, 1998) balances shared and total descriptive information. Eskin's metric (Eskin et al., 2002) adjusts for attribute arity to mitigate chance agreement. Inverse occurrence frequency (Jones, 2004) down-weights common values. In DFCollection, we employ Lin1 (Boriah et al., 2008), an extended version of Lin's similarity measure, to evaluate the similarity between subspaces.

3. System Design

To enable scalable and hierarchical exploration, we derived the design requirements from existing studies and their limitations, and developed the overall framework of our system.

3.1. Design Requirements

Our goal is to enable users to efficiently conduct scalable and hierarchical visual exploration of datasets containing numerous data facts in a structured manner, while also allowing them to review and record those they find interesting. To accomplish this, we identify five key requirements based on a comprehensive literature review:

R1: Automatic Fact Extraction. Datasets typically comprise numerous subspaces, making it crucial to efficiently identify and extract meaningful data facts while minimizing redundant computations on uninformative subspaces or insignificant results (Ding et al., 2019; Tang et al., 2017). To achieve this, the fact-mining algorithm should align with the overall system design and integrate effective strategies to filter out results that fall below predefined thresholds.

R2: Observation of Overall Distribution Patterns. Beyond examining individual data facts, users should be able to explore the overall distributions of both data facts and their collections (Jiang et al., 2025; Wu et al., 2025). Achieving this requires the development of appropriate similarity metrics for comparing facts and collections, which can in turn guide layout design (Jiang et al., 2025; Chen et al., 2024). Such metrics enable users to reason not only about individual elements but also about the broader patterns and relationships among them.

R3: Scalable Visualization. Given that the system must present numerous data facts within a constrained display space, a scalable grouping and visualization strategy is essential. Directly projecting all data facts can quickly lead to visual clutter as their number increases. Therefore, it is important to design effective grouping mechanisms and visualization techniques that maintain readability and detail while minimizing clutter. Such strategies should also support smooth transitions from overview to detail, enabling users to navigate large fact spaces efficiently (Pienta et al., 2015; Qin et al., 2020).

R4: Hierarchical Exploration. To enable effective exploration, the system should adopt a hierarchical approach rather than flattening diverse facts into a view, thereby facilitating progressive understanding and revealing multidimensional relationships instead of only focusing on similarity-based connections (Hornof, 2004; Müller et al., 2017; Bikakis et al., 2016).

R5: Data Story Construction from User Exploration. As users navigate and identify interesting facts, the system should enable them to record their exploration history, including detailed fact information and interaction sequences. These recorded interactions can then be synthesized into a coherent data story, enabling the creation of personalized narratives that reflect each user's unique exploration process (Li et al., 2024a).

3.2. Framework

As shown in Figure 2, our system framework comprises two main modules: Data Preparation and Data Exploration.

The data preparation module processes datasets and transforms them into a format suitable for visualization. The process begins with automatic fact extraction, where pruning and fast detection techniques (**R1**) improve efficiency by eliminating redundant computations while preserving only meaningful and eligible facts. The extracted facts are subsequently organized into collections, forming the basis for scalable visualization and hierarchical exploration (**R3, R4**). Next, similarity metrics are computed both between collections and among facts within each collection (**R2**). Upon completion, the data is fully prepared for visualization and interactive exploration.

The data exploration module builds on this foundation and provides four interactive components. The Collection View presents the overall distribution of collections (**R2**). From this view, users can select a collection and examine its internal structure in the Collection–Fact View, which displays constituent facts and their hierarchical organization. To investigate relationships and patterns in greater depth, the Fact View highlights similarities among individual facts (**R2**). Throughout the process, users can review and record detailed patterns and noteworthy findings in the Fact Gallery (**R5**). These records can then be assembled into a coherent narrative that reflects the exploration sequence, highlights key insights found during exploration, and illustrates the user's developing understanding.

Journal Pre-proof

Table 1: Fact Details Overview

Category	Type	Description	Fast Detection	Significance
Point	Dominance	The top value dominates the scope and is remarkably higher than the remaining values	Top-1 value ≥ 50% share	Fits a power-law to the rest, models residuals, and tests if the top value is unusually large.
	Outstanding Twin	The top two values are remarkably higher than the remaining values	Top-1 value ≥ 33% share, Top-2 value ≥ 33% share	Same as Dominance, but for the sum of the top two values.
	Evenness	All values are close to each other	COV ≤ 0.15	Converts values to probabilities and computes normalized Shannon entropy.
Shape	Change Point	A time series has a significant change in mean value and slope between its preceding and successive regions	max(gaps) > Q3 + 1.5IQR	Slides a split point, comparing left/right windows; tests mean shift and slope change via regression, returning the stronger effect.
	Trend	A time series has an obvious upward or downward trend within a certain level of turbulence.	Mono ≥ 0.75, Turb ≤ 0.60	Fits a line by regression and tests whether the slope differs from zero.
	Skewness	Data are unevenly distributed around the mean and are skewed to the left or right	Normalized COM shift ≥ 0.15	Computes skewness and tests whether it significantly differs from zero.
Compound	Temporal Correlation	Two measures have remarkable correlation within the same subspace and ordinal breakdown.	$\|r\| > 0.85$	Computes Pearson correlation and tests its significance with a t-test.
	Correlation	Same as temporal correlation but for categorical measures.	$\|r\| > 0.85$	Same as Temporal, but for categorical pairs.

Notes: COV denotes the coefficient of variation of data. Gaps are the absolute differences between consecutive rolling means; Q3 is the third quartile and IQR the interquartile range, both calculated from the distribution of gaps. Mono refers to the fraction of same-direction steps among non-zero steps, while Turb indicates the regression error relative to the total trend amplitude. Normalized COM shift denotes the relative movement of the data's Center Of Mass, scaled by a reference range or spread. r represents the Pearson correlation coefficient between two measures.

4. Data Preparation

In this section, we present the formulation of *data facts* and *fact collections*, along with the corresponding similarity metrics that form the foundation for subsequent visualization and exploration. For clarity, we begin by introducing the parameters used in the following definitions. Let the subspace of a fact be defined over n attributes. The k-th attribute is denoted by a_k, its assigned value by v_k, and the set of all distinct values of a_k by V_k. The i-th distinct value in V_k is represented as v_k^i, and the set of all attributes is denoted by A. A wildcard symbol $*$ indicates that no filter is applied to the corresponding attribute. Let R denote the total number of rows in the dataset.

4.1. Data Fact

Data facts, which encapsulate interesting patterns within subspaces, form the fundamental building blocks of data exploration. Building on prior work (Ding et al., 2019; Tang et al., 2017), we adopt a simplified formulation of data facts for our DFCollection system and introduce tailored procedures to systematically process datasets and extract all valuable facts that satisfy the defined requirements. Each data fact is characterized by five core elements, which can be represented as a 5-tuple:

$$(subspace, breakdown, measure, type, score).$$

Subspace defines the scope of a data fact and is specified by a set of filters applied to the dataset's attributes. To construct a subspace, we consider all categorical attributes, including both nominal and ordinal types, and organize them into a fixed-length tuple of size n. In our system, wildcards are retained within the tuple, as the subspace degree determines its completeness during the algorithm. Formally, a subspace can be expressed as:

$$\{\{a_1 = v_1\}, \{a_2 = v_2\}, \ldots, \{a_n = v_n\}\}, \quad v_k \in V_k \cup \{*\}.$$

These values function as filters applied to the dataset, and their combination specifies the domain of the data fact.

Breakdown operates as a group-by function on a subspace. For each subspace, the breakdown attribute must be chosen from among the attributes without applied filters. This operation partitions the data within the subspace into smaller groups, preparing it for the subsequent aggregation step.

Measure refers to one or two numerical attributes used for aggregation. After uploading a file, the user selects aggregation functions for each numerical attribute (e.g., sum, average, minimum, maximum) through the interface. These functions are then applied to compute aggregated values for each group produced in the breakdown step.

Type of data facts is derived from prior research (Chen et al., 2024; Ding et al., 2019; Tang et al., 2017), from which representative types are selected and incorporated into our system. Facts are categorized as Point, Shape, or Compound (Ding et al., 2019). Point facts use categorical breakdowns, Shape facts use ordinal ones, and Compound facts combine two measures to show relationships. A detailed taxonomy of these types is presented in Table 1.

Score quantifies the interestingness of a data fact and is determined by two factors: impact and significance. For each data fact, impact is evaluated using an aggregation method that satisfies the anti-monotonic condition (Ng et al., 1998; Ding et al., 2019). In our system, we apply a count aggregation method for all measures, making the impact depend solely on the number of data entries within each subspace, thereby ensuring simplicity and efficiency. Thus, the impact of $fact_i$ is defined as $impact_i = \frac{\text{COUNT}(subspace_i)}{\text{COUNT}(\{*\})}$. The calculation of significance is designed based on prior work (Tang et al., 2017; Ding et al., 2019; Chen et al., 2024), and the specific methods for computing significance for each type are summarized in Table 1. By

Journal Pre-proof

Algorithm 1: Automatic Fact Extraction

Input: Dataset S
Output: Set of extracted facts F

$iThreshold \leftarrow 0.01$;
$sThreshold \leftarrow 0.80$;
$F \leftarrow \emptyset$;
Function AFE(S, *filterSet*, *mask*, k):
  $coverage \leftarrow \frac{1}{R}\sum_{i=1}^{R} mask[i]$;
  if $coverage < iThreshold$ **then**
    return;
  end
  if $k = n$ **then**
    $impact \leftarrow coverage$;
    $subspace \leftarrow filterSet$;
    $type \leftarrow$ fastDetection(*subspace*);
    if $type \neq$ NULL **then**
      $fact, significance \leftarrow$ SigCalculation(*type*);
      if $significance \geq sThreshold$ **then**
        $F \leftarrow F \cup \{fact\}$;
      end
    end
    return;
  end
  foreach $v_{k+1}^i \in V_{k+1} \cup \{*\}$ **do**
    $newMask \leftarrow mask \wedge [S : a_{k+1} = v_{k+1}^i]$;
    AFE($filterSet \cup \{a_{k+1} = v_{k+1}^i\}, newMask, k+1$);
  end
AFE($S, \emptyset, \mathbf{1}^R, 0$);

combining these two factors, the score of $fact_i$ is expressed as:

$$score_i = impact_i \times significance_i. \quad (1)$$

We present one example to further illustrate the concept of a data fact. Consider a dataset with three subspace attributes: *Country*, *Gender*, and *Year*. One possible data fact derived from this dataset is ($\{\{Country = China\}, \{Gender = Male\}, \{Year = *\}\}$, *Year*, *Income*, *Trend*, 0.9), which indicates that the income of males in China exhibits an increasing or decreasing trend over the years, with a fact score of 0.9.

DFCollection explores and visualizes all data facts that meet the defined thresholds, rather than limiting output to only the top-ranked results. Unlike existing fact-driven tools, we propose an Automatic Fact Extraction (AFE) function (**R1**) that applies Depth-First Search with impact-based pruning, rather than best-first prioritization (Ding et al., 2019). The extraction process is shown in algorithm 1. In practice, we set the impact threshold to 0.01, consistent with previous work to maintain comparable filtering sensitivity (Ding et al., 2019). The significance threshold is set to 0.80, slightly lower than prior studies (Ding et al., 2019; Tang et al., 2017), allowing the inclusion of a broader set of data facts while still ensuring statistical relevance. This choice balances the tradeoff between capturing sufficient insights and maintaining fact reliability in large, heterogeneous datasets.

The algorithm initializes three components: $filterSet$, *mask*, and k. The $filterSet$ stores the current subspace, whether complete or partial. The *mask* identifies rows in dataset S that match the $filterSet$, acting as a selection vector. The variable k records the number of applied filters, representing the search depth. To enable pruning, the algorithm computes *coverage*, which measures the influence of rows matching the $filterSet$. Branches with *coverage* below the threshold are pruned, since deeper branches introduce additional filters to the $filterSet$, meaning the number of matching records and the corresponding *coverage* cannot exceed that of the current branch. The monotonicity of the count aggregation method ensures the correctness of this pruning. When k equals n, the subspace is complete. The fast detection step, guided by each type's descriptive characteristics and the specifications (Ding et al., 2019; Li et al., 2025), as shown in Table 1, is then applied to quickly identify the fact type. The algorithm then assesses the fact's significance. If it meets or exceeds the threshold, the fact is emitted. Otherwise, if the subspace is incomplete, the algorithm extends the $filterSet$ with the next attribute's values and continues recursively until all branches are explored.

4.2. Fact Collection

After extracting a set of data facts, the next challenge is managing their potentially large volume to support scalable exploration. To tackle this, we introduce the concept of *fact collection* (**R2**). Before its formal definition, we first provide a preliminary one.

Definition 1 (Filter Attribute Set): Given a data fact $fact_k$, its *Filter Attribute Set* (FAS) is defined as:

$$FAS_k = \{\, a_i \in A \mid \{a_i = v_i\} \in fact_k.subspace,\ v_i \neq * \}. \quad (2)$$

In other words, the FAS of a data fact consists of all attributes that are constrained by specific values in the fact's subspace. It determines the subspace dimensions of a data fact. For example, the FAS of the data fact with subspace $\{\{Country = China\}, \{Gender = Male\}, \{Year = *\}\}$ is $\{Country, Gender\}$.

We organize data facts according to their FAS, which groups facts sharing the same filtered attribute dimensions. While this strategy resembles OLAP-style aggregation and group-by operations based on fact type or statistical properties, it differs substantially in both purpose and representation. OLAP methods primarily focus on summarizing numerical measures across multidimensional cubes for querying and analytical processing. Group-by operations based on fact type mainly categorize facts according to interestingness magnitude or predefined fact categories, rather than semantic relationships. Similarly, group-by operations based on statistical properties cluster facts according to shared numerical characteristics or statistical patterns, which may overlook the contextual meaning of the underlying subspaces. In contrast, FAS-based grouping organizes automatically extracted data facts into semantically consistent analytical units for visual exploration, preserving subspace semantics and relationships among facts to support interpretable similarity computation and hierarchical exploration.

Building on the concept of FAS, we organize data facts into fact collections, where each collection consists of facts that share the same FAS. This grouping supports structured exploration at a higher level, as all FAS-equivalent facts can be treated as a single analytical unit. In this way, relationships between collections can be studied holistically, while still allowing users to drill down into individual collections to examine the specific facts they contain. Organizing facts by FAS also ensures that similarity computations between subspaces remain meaningful, as comparisons are confined to consistent subspace dimensions rather than heterogeneous ones. Since the union of breakdowns or measures across the contained facts is nearly identical across collections, covering almost all categorical or numerical attributes, it provides limited analytical value and is therefore excluded from our definition. Formally, a fact collection is represented as a 3-tuple:

$$(dimension, composite, index).$$

Dimension specifies the FAS shared by all facts in a collection, representing its overall dimensionality. Since every fact shares the same FAS, the dimension of the collection is identical to the FAS of its contained facts. The number of distinctive attributes included in the dimension is called the *filter count*.

Composite describes the variety of fact types within a collection. It is defined as the union of the types of all contained facts, identifying which types are present. This property captures the diversity of patterns within the collection, and its size is called the *type count*.

Index measures the overall interestingness of a collection. It is computed by summing the scores of all contained facts, providing an aggregate measure of the collection's significance.

By organizing facts in this way, we move beyond isolated analysis and obtain a structured representation that summarizes the contained facts while maintaining consistent subspace dimensions. This representation facilitates scalable visualization and enables subsequent hierarchical exploration.

4.3. Similarity Metrics

With fact extraction and collection formulation completed, the next step is to define similarity metrics for fact collections, followed by those for data facts.

4.3.1. Metrics for Fact Collections

To compare the similarity between collections, we consider their three key components. Focusing on the dimension aspect, suppose two collections have dimensions $\{A, B, C\}$ and $\{B, C, D\}$. In this case, (i) both share the same filter count, and (ii) they overlap on the subset $\{B, C\}$. Therefore, both filter count similarity and dimension overlap serve as important indicators of similarity. To capture these aspects, we employ complementary measures for count similarity (Alzamili et al., 2025) and for dimension overlap (Simpson, 1960). Combining these views, the similarity between dimensions D_m and D_n is defined as:

$$S_{dimension} = \alpha \cdot \frac{\min(|D_m|, |D_n|)}{\max(|D_m|, |D_n|)} + (1-\alpha) \cdot \frac{|D_m \cap D_n|}{\min(|D_m|, |D_n|)}, \quad (3)$$

where $\alpha \in [0, 1]$ is a weighting parameter that balances the influence of filter count versus dimension overlap. A larger α increases the emphasis on the relative size of the dimension sets, while a smaller α increases the emphasis on their overlap. By default, we set $\alpha = 0.5$ to give equal weight to both aspects. A similar strategy is applied to compute the similarity between collections, denoted as $S_{composite}$.

To assess similarity in terms of index, we adopt an intuitive measure based on relative difference, which captures how close the index values of two collections are. The similarity between two indexes I_m and I_n is defined as:

$$S_{index} = 1 - \frac{|I_m - I_n|}{\max(I_m, I_n)}. \quad (4)$$

By integrating the three aspects, the overall similarity between $collection_m$ and $collection_n$ is expressed as:

$$S_{collection} = \gamma_1 \cdot S_{dimension} + \gamma_2 \cdot S_{composite} + \gamma_3 \cdot S_{index}, \quad (5)$$

where $\sum_{i=1}^{3} \gamma_i = 1$, and the weights can be adjusted to reflect user preferences or task requirements. In our implementation, we set them to 0.4, 0.4, and 0.2 respectively.

4.3.2. Metrics for Data Facts

Within each collection, we further assess the similarity between data facts. We begin with the subspace, which defines the scope of a data fact. As each subspace is defined by a set of filters applied to the dataset's categorical attributes (Ding et al., 2019; Tang et al., 2017; Shi et al., 2021), the similarity between subspaces can be computed by comparing the filter values of corresponding categorical attributes and aggregating the results. Previous research has rarely incorporated similarity metrics for categorical attributes into subspace comparison, often overlooking this aspect due to heterogeneous dimensions or addressing it only in limited ways (Li et al., 2024b; Dennig et al., 2024). In our framework, all facts within a collection share the same FAS, which eliminates ambiguous cases that are difficult to interpret, such as comparing $\{Country = China\}$ with $\{Country = *\}$, and ensures that similarity computation for categorical attributes remains both consistent and meaningful.

Numerous similarity metrics have been proposed for categorical data (Sulc and Rezanková, 2019; Boriah et al., 2008). After evaluating various options, we adopt Lin's measure (Lin, 1998), which provides an information-theoretic framework that assigns higher similarity to frequent matches and lower scores to rare mismatches in terms of frequency. However, because Lin's metric is better suited for ordinal or semantic data, we employ Lin1 (Sulc and Rezanková, 2019), an extension that orders values by frequency, offering broader and more consistent applicability across attributes. Formally, given two data facts $fact_m$ and $fact_n$ within the same collection, let D denote the collection's dimension, $A_k \in D$ denote an attribute in the collection's dimension, and v_k^m and v_k^n be the corresponding filter values drawn from V_k. The similarity between v_k^m and v_k^n is therefore defined as:

$$S_{Lin1}(v_k^m, v_k^n) = \begin{cases} 1 & \text{if } v_k^m = v_k^n \\ \frac{2\log \sum_{q \in Q} p_k(q)}{\sum_{q \in Q} \log p_k(q)} & \text{if } v_k^m \neq v_k^n \end{cases}, \quad (6)$$

Journal Pre-proof

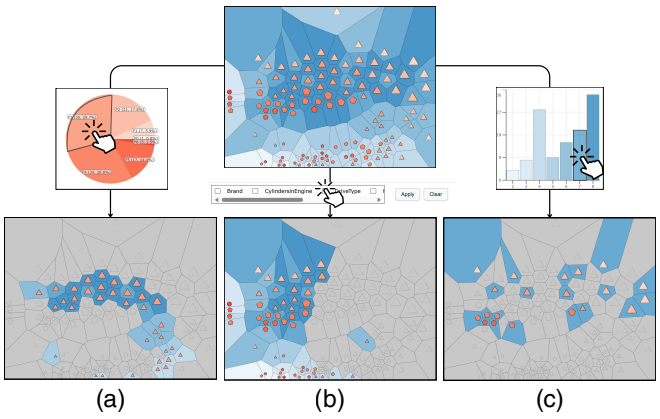


Figure 3: Interactive mechanisms of the Collection View. The system supports three main filtering mechanisms: (a) Filter Count filtering, (b) Attribute filtering, and (c) Type Count filtering.

where $\{Q \subseteq V_k \mid \forall q \in Q,\ p_k(v_k^m) \leq p_k(q) \leq p_k(v_k^n)\}$, assuming $p_k(v_k^m) \leq p_k(v_k^n)$. To maintain consistency within the similarity framework, we apply Lin1 uniformly to both nominal and ordinal attributes. Building on this, the similarity between two subspaces defined by filters can be calculated by aggregating attribute-level similarities. Then the normalized subspace similarity between subspaces of facts $fact_m$ and $fact_n$ is given by:

$$S_{subspace} = \frac{1}{|D|} \sum_{A_k \in D} S_{Lin1}(v_k^m, v_k^n). \tag{7}$$

We denote the subspace similarity as S_1. For breakdown, measure, and type, we apply a simple overlap rule: identical values have a similarity of 1, and different values have 0, denoted as S_2, S_3, and S_4 respectively. For the score component, we use the same computation as for S_{index}, replacing the collection index with the fact's score, denoted as S_5. The overall similarity between two data facts is then defined as a weighted sum of these components:

$$S_{fact} = \sum_{i=1}^{5} w_i \cdot S_i, \tag{8}$$

where $\sum_{i=1}^{5} w_i = 1$, and the weights can be adjusted according to preferences or requirements. In our implementation, these values are set to 0.5, 0.1, 0.1, 0.2, and 0.1 respectively, with greater emphasis placed on the subspace component.

5. Data Exploration

To align with the structure of our framework and facilitate effective user exploration, the visualization system is organized into three sequential components: the Collection View, Collection–Fact View, and Fact View, which together support exploration and help users locate facts of interest. An additional component, the Fact Gallery, allows users to review and record detailed information about selected facts. The overall system interface is shown in Figure 1.

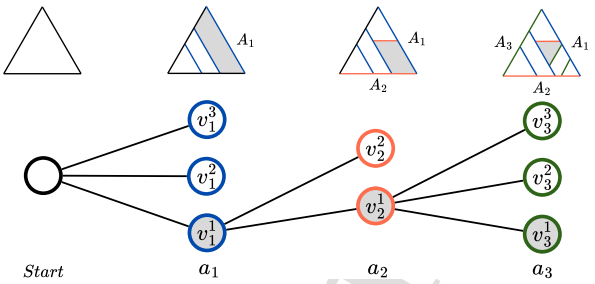


Figure 4: A collection with three filters begins as a triangle. The first attribute a_1 with three distinct values divides the triangle into three subpolygons, with the gray region representing $\{a_1 = v_1^1\}$. The second attribute a_2 further divides this region into two, forming $\{a_1 = v_1^1, a_2 = v_2^1\}$, and the third attribute a_3 splits it again into three, resulting in $\{a_1 = v_1^1, a_2 = v_2^1, a_3 = v_3^1\}$, thereby completing one splitting branch.

5.1. Collection View

In this view, as shown in Figure 1A, users can examine the overall distribution of fact collections, select a collection of interest, and further explore its details in the subsequent views (**R2**, **R3**), with additional features to support exploration.

Layout distribution. Using the pairwise distance matrix from similarity calculations, we apply Multidimensional Scaling (MDS) (Davison and Sireci, 2000; Saeed et al., 2018) to project collections into a two-dimensional space, providing an overview of their global distribution. Since collections occupy areas rather than points, overlaps may occur after projection. To address this, we implement an overlap removal algorithm inspired by force-directed layouts (Fruchterman and Reingold, 1991) and label placement methods (Christensen et al., 1995), modeling collections as repelling bounding boxes to maintain spatial clarity and visual interpretability.

Glyph design. Inspired by the Categorical Data Map (Dennig et al., 2024), we choose to use a Voronoi-style layout for better visual effect. Building on established glyph design principles (Borgo et al., 2013; Fuchs et al., 2017) and maintaining visual consistency with the Collection–Fact View, we represent each collection as a regular polygon. The number of polygon sides reflects the number of filters in a collection: for odd filter counts, the polygon has the same number of sides, while for even counts we use $n + 1$ sides (Reza and Watson, 2019). The fact collection generation step shown in Figure 2 further illustrates this design, demonstrating collections with filter counts ranging from 0 to 7. Different polygon shapes, together with a sequential color scheme, help distinguish collections with different filter counts. To ensure scalability across different datasets with varying numbers of collections, the size of each glyph is dynamically adjusted. Additional visual encodings further enrich the representation: polygon color and the number of solid edges indicate the number of dimensions, with darker tones representing higher filter counts. Surrounding Voronoi cells are shaded in blue to encode type counts, where darker shades indicate greater diversity. Together, these design choices allow users to quickly grasp the overall distribution of collections while also inspecting detailed attributes of individual collections, as illustrated in Figure 1A2.

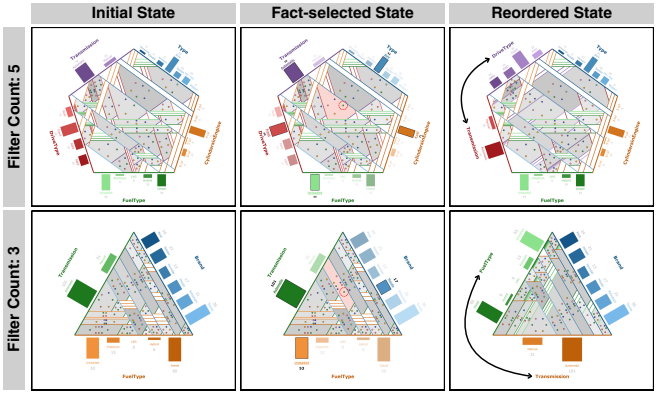


Figure 5: Three exploration states of the Collection–Fact View. The figure illustrates the initial, fact-selected, and reordered states for two example collections with filter counts of 5 and 3, respectively.

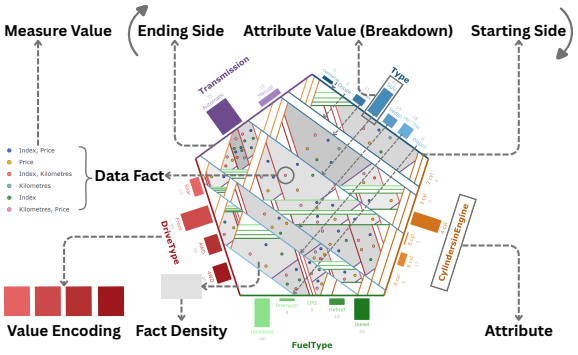


Figure 6: A detailed illustration of the Collection-Fact View, where different labels and visual components are annotated.

Interactive filtering mechanisms. The overview in Figure 1A1 displays the distributions of index, type count, and filter count for the collections spatially arranged in Figure 1A2. To support efficient exploration, the Collection View offers three interactive filtering mechanisms that let users focus on subsets of collections matching specific criteria while dimming others. As illustrated in Figure 3, these mechanisms include: (a) Filter Count filtering, which highlights collections with a selected filter count; (b) Attribute filtering, which highlights collections containing a chosen dimension; and (c) Type Count filtering, which highlights collections according to their type variety. Together, these tools help users refine the view, isolate relevant subsets, and efficiently locate collections of interest.

5.2. Collection-Fact View

Once a collection is selected, users can explore its contained data facts hierarchically, examine their multidimensional relationships, and observe their hierarchical distribution (**R4**); an example is shown in Figure 1B, with the corresponding view illustration shown in Figure 6.

Formulation. Inspired by Hi-D Map (Reza and Watson, 2019), the Collection–Fact View represents each collection as a polygon, where each side corresponds to one attribute of the collection's dimension. Attributes are distinguished by color. Starting from the top-right side, the polygon is recursively subdivided to reflect the filter values of each attribute. The subdivision proceeds in a near-side, clockwise order, and the area of each resulting subpolygon encodes the total score of the facts it contains. Figure 4 illustrates an example of this splitting process. Branches that contain no facts automatically stop subdividing. To reduce visual overlap caused by empty or very small regions, we slightly adjust the area ratios derived from scores (setting them to 0.05 for zero scores and 0.15 for nonzero scores) and then re-normalize the layout. Attribute values are encoded using a dark-to-light color gradient applied clockwise, making the subdivision order easy to follow. Individual data facts are shown as small circles inside the subpolygons. When an attribute has more than ten distinct values, they are grouped into three bins to improve readability.

Visual Design. To improve readability, each side of the polygon is paired with a bar chart that shows how many facts fall under each filter value of that attribute, enabling quick comparison at a glance. Attribute values are encoded using a dark-to-light color gradient applied in a clockwise order and are consistently mapped to both the subdivision boundaries and the bars in the charts. For example, in Figure 1B, the Type side shows that 18 facts satisfy the condition $\{Type = Ute/Tray\}$ within the subspace; correspondingly, the blue color gradually becomes lighter as it extends from the polygon side toward the interior. Subpolygons are shaded in grey, where darker shades indicate a larger number of contained facts. Individual facts are represented as circles inside the polygon and are encoded by their measure, with a legend provided in the top-left corner of the view to support interpretation. To maintain visual clarity, facts within each subpolygon are evenly distributed around the center while remaining within the subpolygon boundaries.

Interaction. The Collection–Fact View supports three exploration states, as illustrated in Figure 5, which shows examples for two collections with filter counts of three and five. In the *initial state*, attributes are ordered by the number of distinct values from largest to smallest, reducing visual clutter by preventing small subpolygons from being split into smaller sections. In the *fact-selected state*, selecting a fact highlights it with a red circle, emphasizes the corresponding subpolygon in red, and marks the relevant filters for each attribute using bar charts along the polygon's sides. In the *reordered state*, users can drag and drop attributes to modify their order, bringing attributes of interest to the forefront for more flexible exploration; in Figure 5, for instance, the last two attributes are swapped, and the layout updates accordingly.

5.3. Fact View

The Fact View provides an overview of the score and count distributions of all facts in the selected collection and arranges them according to their similarity (**R2**), as shown in Figure 1C. Figure 7 illustrates how the view is constructed and how to interpret its visual encodings. In Figure 7a, each data fact is represented by a circular glyph composed of an outer colored ring and a miniature chart icon. The ring color distinguishes the three fact categories shown in the legend: blue for Point,

Journal Pre-proof

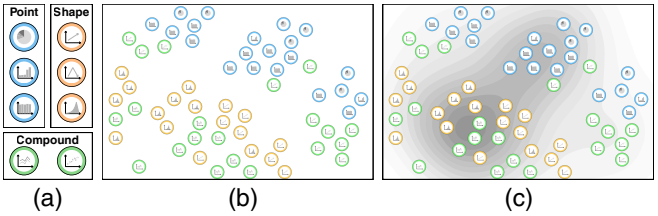


Figure 7: Construction process of the Fact View, consisting of three main stages: (a) fact glyph encoding, (b) dimension reduction, and (c) addition of score density contours.

orange for Shape, and green for Compound. Within each category, the inner icon indicates the specific fact type; for example, the icons depict different distribution patterns, trends, or relationships, allowing facts with the same ring color to be distinguished. In Figure 7b, the glyphs are placed on a two-dimensional canvas using the similarity-based layout strategy of the Collection View, so nearby glyphs represent more similar facts. Finally, Figure 7c adds grayscale density contours (Silverman, 1986) beneath the glyphs to summarize the spatial concentration of fact scores. Darker regions indicate greater score concentration, helping users identify high-score areas while retaining the individual glyphs for detailed inspection.

5.4. Fact Gallery

In the Collection–Fact View and Fact View, users can select facts of interest, examine their details, and record those with notable patterns in the Fact Gallery (**R5**), as shown in Figure 1D. The Fact Gallery offers a snapshot function that enables users to capture interesting data facts discovered during exploration. Once a fact is recorded, its position is highlighted in the Collection View, and the corresponding Collection–Fact View is displayed in the top-right corner for quick reference. Through this process, users can review detailed information, preserve meaningful findings, and progressively construct a personalized data story within the system (Li et al., 2024a).

5.5. Alternative Designs

The Collection–Fact View uses multiple visual channels to represent hierarchical data facts. While this multi-channel approach introduces complexity, it is necessary to support scalable exploration of large and heterogeneous fact spaces. The view follows an overview-to-detail reading path, allowing users to move from collection-level distribution to fact-level inspection.

We considered common hierarchical and multidimensional visualization techniques, including treemaps (Shneiderman, 1992), sunburst diagrams (Stasko and Zhang, 2000), and parallel coordinates (Inselberg, 1985), as potential alternatives for the Collection–Fact View. Treemaps efficiently use 2D space for hierarchical data, but they obscure attribute ordering and semantic relationships in multi-level hierarchies. Sunburst diagrams preserve hierarchical containment radially, yet radial distortion and uneven angular allocation reduce interpretability for dense data facts. Parallel coordinates represent multidimensional relationships well but lack hierarchical structure and semantic grouping. Moreover, these techniques each convey only limited aspects of analytical information for fact exploration.

In contrast, the Collection–Fact View jointly encodes hierarchical containment, semantic consistency, fact density, and progressive drill-down in a polygonal subdivision layout. This design introduces complexity but preserves relationships across heterogeneous subspaces, supports interpretable similarity computations, and provides a clear visual hierarchy for overview-to-detail exploration. Although multiple visual channels are used simultaneously, each channel serves a distinct role. Integrating these encodings within a single view helps maintain contextual awareness during exploration and avoids the fragmentation caused by multiple disconnected views. While simpler alternatives exist, they do not simultaneously support hierarchical navigation, semantic grouping, density overview, and local fact inspection, which are central to DFCollection's analytical goals. The resulting visual complexity is therefore an intentional design choice aimed at supporting DFCollection's core analytical and exploratory tasks.

6. Usage Scenarios

To evaluate the usefulness and effectiveness of our system, we present two usage scenarios that illustrate how a data analyst interacts with DFCollection in real-world settings, using two public datasets that were preprocessed for system compatibility.

6.1. Scenario 1: Australian Car Price

The first dataset contains records of second-hand car sales in Australia from 2010 to 2024[1]. From this dataset, a total of 8,811 data facts were generated. It contains 7,104 rows and 10 columns, with 3 measure attributes and 7 subspace attributes. The analyst explores how vehicle characteristics affect resale prices, moving from collections with fewer filters to those with more, gradually refining the analysis without applying custom filters in the Collection View. The exploration follows collection similarity, proceeding step by step through nearby collections. Notably, the variable *Year* indicates the car's purchase year rather than its resale year.

Given such amount of data facts, the analyst expresses that the system remained very scalable and navigable even when viewing all fact collections that contain 8,811 facts in total in the overview. The exploration begins with Collection 1, which contains no filters. At this broad level, the analyst observes a clear negative correlation between kilometers driven and sale price across years. Moving to Collection 3 with one filter, diesel vehicles with 5-cylinder engines dominate sales, while unleaded cars lead in the 6-cylinder category. In Collection 4, defined by two filters, a distinct pattern appears for 4-cylinder automatic cars, whose sales rose steadily until peaking in 2018. With three filters in Collection 5, brand differences become evident: Holden and Toyota outperform others within the AWD, unleaded, 6-cylinder segment. Finally, in Collection 6

[1] `https://www.kaggle.com/datasets/nelgiriyewithana/australian-vehicle-prices`

Journal Pre-proof

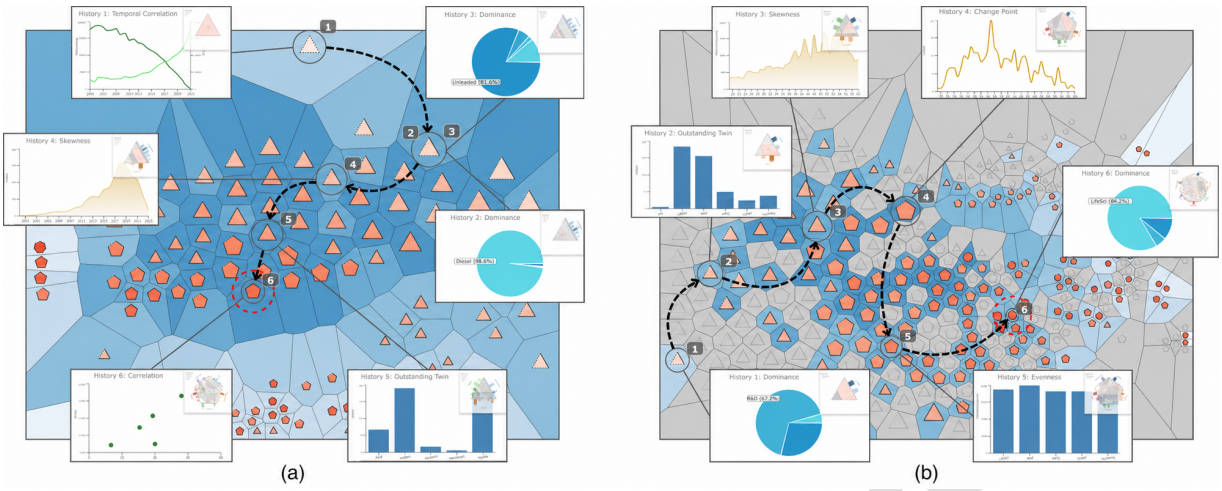


Figure 8: Two data stories are constructed in the usage scenarios: (a) an exploration of second-hand car sales in Australia (2010–2024) and (b) an analysis of employee records related to attrition and HR analytics. The figure combines the Collection View and Fact Gallery, with the identified data facts displayed alongside their corresponding fact collections for a more intuitive presentation. Scenario (b) applies attribute-based filtering on Attrition, as illustrated in Figure 3b.

with five filters, the analyst finds that higher-priced hatchbacks with manual transmission, unleaded fuel, and 4-cylinder engines achieve the strongest sales. Through this open-ended exploration, the analyst constructs a coherent data story, as shown in Figure 8(a).

6.2. Scenario 2: HR Employee Analytics

The second dataset contains employee records related to attrition and human resource analytics[2]. From this dataset, 13,566 data facts were generated. It consists of 1,470 rows and 11 columns, with 2 measure attributes and 9 subspace attributes. In this scenario, the analyst focuses on the *Attrition* attribute to distinguish findings by attrition status. He limits the analysis to employees who have not attrited, using an attribute-filtering function mentioned in subsection 5.1 that greys out collections lacking *Attrition*. Unlike the first case, the analyst does not follow collection similarity but instead targets collections defined by specific dimensions, placing *Attrition* first in the filter sequence in each collection for clarity.

Within the system's hierarchical visual design, the analyst can quickly locate data facts with a clear focus on *Attrition* across each fact collection. The analyst begins with the *Attrition* filter, focusing on retained employees, and observes that the Research and Development department dominates the workforce. Adding *Gender*, he finds Life Sciences and Medicine dominant among female employees. Introducing *BusinessTravel* and narrowing to females who rarely travel, he relates age to monthly income and notes that middle-aged and older employees earn more than younger ones. Filtering further by *Department*, he analyzes males in Research and Development who rarely travel and identifies a sharp change around age thirty-five, where counts rise and then decline. Adding *Education*, he observes that male Sales employees with an education level 2 who rarely travel show no strong income concentration across education fields. Finally, with the *JobRole* filter applied to male Research Scientists in Research and Development with education level 2 who rarely travel and have not attrited, he finds Life Sciences to be the dominant education field. The analyst builds a focused data story for the second dataset, as shown in Figure 8(b), using the attribute-filtering function on *Attrition*.

7. Interview

In this section, due to the lack of existing work on fact collections and hierarchical exploration of data facts, we adopt user interviews as our primary evaluation method. We conducted a semi-structured interview study with 13 participants to assess the effectiveness and usability of DFCollection.

7.1. Participants and Apparatus

We recruited 13 participants (P1–P13) from universities, companies, and research institutes (3 females, 10 males; M_{age} = 25.38, SD = 2.47), all with normal vision and prior experience in data analysis or visualization. P1 and P2 were PhD experts with over five years of experience, P8 was a data analyst intern, P7, P10, P11, and P13 were master's students specializing in data analysis and visualization, and P9 was an algorithm engineer focusing on data processing. The remaining participants (P3, P4, P5, P6, P12) were undergraduate or PhD students in computer-related fields who had extensive coursework and hands-on experience in data analysis or visualization, ensuring they could effectively engage with DFCollection and provide informed feedback. While the participant pool consisted primarily of students, all participants had relevant expertise in visualization, data analysis, or related domains, allowing them to provide informed feedback on the system's usability and exploratory capabilities. Interviews were conducted online

[2] https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset

Journal Pre-proof

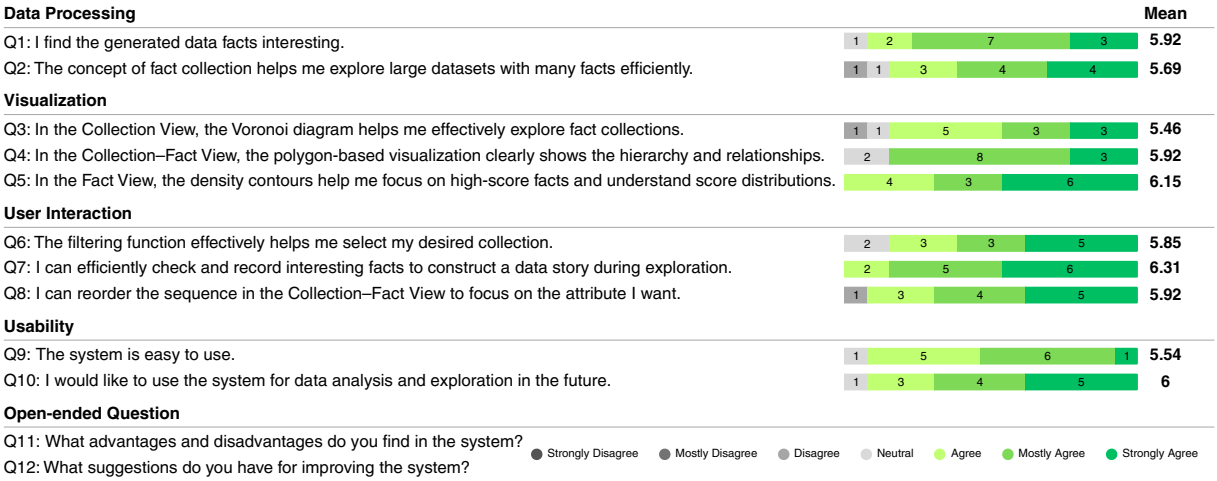


Figure 9: The questionnaire used in the interview is structured into five sections: Data Processing, Visualization, User Interaction, Usability, and Open-ended Questions. Each closed-ended question includes a corresponding rating, with detailed information provided for each scoring component. The questionnaire comprises a total of 10 closed-ended questions and 2 open-ended questions.

via Zoom, with DFCollection deployed on a public server accessible from participants' own computers. The study was conducted under ethical approval, and participation was voluntary for all interviewees.

7.2. Procedure

The interview began with an overview of DFCollection, its background, and limitations of prior work, followed by explanations of the system framework, key concepts, and visual design (10 minutes). A demonstration using a simple dataset from prior work (Ding et al., 2019) introduced two exploration approaches: similarity-based (subsection 6.1) and filter-based (subsection 6.2) (10 minutes). Participants subsequently explored two additional datasets independently (20 minutes). During this phase, they were instructed to find and record 5 interesting data facts in the Fact Gallery during their exploration. This task was intentionally open-ended to encourage natural exploration behaviors, although it may limit direct comparison across participants or systems. Finally, they completed a questionnaire with 10 closed-ended (Q1–Q10) and 2 open-ended (Q11–Q12) questions, as shown in Figure 9. The closed-ended questions, rated on a 7-point Likert scale (Joshi et al., 2015), evaluated data processing, visualization, interaction, and usability, while the open-ended questions gathered feedback on DFCollection's advantages, limitations, and suggestions (10 minutes). An open discussion session was then conducted to allow participants to share additional reflections on the system (10 minutes). The evaluation primarily focused on subjective feedback and exploratory behaviors, with limited use of quantitative metrics to assess system efficiency.

7.3. Results

Figure 9 presents the participants' responses to the closed-ended questions (Q1–Q10), showing the score distribution and the calculated mean (M) for each question. The questionnaire results indicate overall positive feedback on the system. The detailed analysis of each question is discussed in this section.

Data Processing. Most participants agreed that the generated facts were interesting (Q1), confirming the effectiveness of our algorithm. Likewise, most found that organizing data facts into collections improved efficiency when exploring large datasets (Q2). P1 and P7 highlighted the value of grouping facts by subspaces, while P9, who rated Q2 with a "3," noted that her daily work rarely involves datasets with such a volume of facts, but suggested it could be valuable in financial analysis.

Visualization. This section evaluated the visual design of the Collection View (Q3), Collection–Fact View (Q4), and Fact View (Q5). Overall, participants responded positively. P1 and P10 praised the polygon-based design of the Collection–Fact View for effectively visualizing multidimensional relationships and hierarchical distribution, and P13 agreed it was well-suited for the system, though locating specific fact sections could be inconvenient. In contrast, P2 found the Collection–Fact View less visually intuitive at first, requiring familiarity with the concepts before effective use, leading to lower scores.

User Interaction. This section used three questions to assess the system's main interaction functions (Q6–Q8). Overall, participants gave positive feedback, with most agreeing that the interactive features effectively supported data exploration. P7 and P10 noted that the filter function might be less effective for users unfamiliar with a dataset, as they may not know where to start. P13 added that while the reordering feature was helpful, the subsequent recalculation and re-evaluation steps could increase cognitive load.

Usability. This section evaluated the system's overall usability and participants' willingness to use it for future data analysis. Overall, participants found the system helpful and expressed interest in applying it to future tasks. However, several participants noted that it takes some time to become familiar with the system.

Open-ended questions. In this section, participants dis-

Journal Pre-proof

cussed the system's advantages, drawbacks, and potential improvements. Overall, they appreciated its ability to handle many data facts and support flexible, interest-based filtering. However, several found it challenging to learn initially, particularly in understanding its concepts and visual encodings. To improve usability, P5, P7, and P9 suggested adding an onboarding tool or interactive guide, P6 and P10 proposed integrating large language models (Zhao et al., 2024) for smarter analysis, and P8 recommended a scoring algorithm to suggest starting points for exploration.

8. Discussion

In this section, we reflect on key lessons learned, acknowledge limitations in our current design, and highlight potential directions for future work.

Computational Complexity. We designed a customized fact extraction algorithm that integrates DFS pruning with a fast detection mechanism to accelerate exploration. Its complexity depends on the number of attributes and their distinct values; given N attributes and up to M distinct values per attribute, the worst-case complexity is $O((M+1)^N)$. While efficient for moderate datasets, performance decreases as M and N grow. In future work, we plan to explore better optimization strategies to maintain responsiveness with large datasets, or develop a real-time approach that computes facts and collections dynamically during user exploration rather than precomputing them all.

To reduce visual congestion, additional preprocessing of the dataset and extracted data facts is often required, such as limiting measure attributes, applying value binning, and simplifying highly heterogeneous subspaces. In future work, we plan to develop a more automated preprocessing framework to improve scalability, efficiency, and usability.

Learnability. DFCollection is effective and useful for scalable and hierarchical exploration, and the Collection-Fact View integrates multiple visual channels to support exploration from overview to detail. However, this design may also increase the learning burden for first-time users, particularly when interpreting the hierarchical subdivision structure and multiple visual encodings simultaneously.

Guided Exploration. Currently, users navigate collections and facts independently to construct data stories based on their exploration paths. However, analysts often face time constraints that prevent them from examining every collection in detail. To address this, future work will incorporate automatic summarization of key facts within each collection to provide concise overviews and speed up exploration while preserving user control. Another direction, as noted by P13, is automatic story generation to offer recommended narratives for user reference.

Recognition of Rare Patterns. Exploration with DFCollection revealed that prior work often emphasizes top-scored facts while overlooking rare but insightful patterns. Many relationships, such as the negative correlation between price and mileage in car datasets, appear repeatedly and offer limited novelty. In contrast, uncommon patterns, such as a positive correlation, can yield more valuable insights due to their rarity. Future work will integrate measures of rareness, derived from statistical or other methods, into the interestingness scoring and visual design to better highlight infrequent yet meaningful patterns.

9. Conclusion

We presented DFCollection, a scalable and hierarchical visual analytics system for exploring a vast number of data facts. By introducing the concept of fact collection, the system organizes data facts by subspace dimensionality and supports multi-level exploration through coordinated visual interfaces. Evaluations through two usage scenarios and in-depth user interviews with 13 participants demonstrate that DFCollection effectively enables scalable, structured, and interpretable exploration of data facts. The evaluation also demonstrated the strengths of the system while revealing several limitations of the current design, which will be addressed in future work.

Acknowledgements

This project is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (NTU Tier 1, RG104/25) and the NTU Start Up Grant awarded to Yong Wang. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore.

References

Alzamili, S.L., Baawi, S.S., Kadhim, M.N., Al-Shammary, D., Ibaida, A., 2025. Efficient feature selection based on ruzicka similarity for EEG diagnosis. International Journal of Information Technology 17, 3373–3387. doi:10.1007/s41870-025-02572-3.

Aodeng, G., Li, G., Feng, Y., Chen, Q., Zhang, Y., Liu, C.H., 2025. InReAcTable: LLM-powered interactive visual data story construction from tabular data, in: ACM Symposium on User Interface Software and Technology, ACM. pp. 139:1–139:16.

Bikakis, N., Papastefanatos, G., Skourla, M., Sellis, T., 2016. A hierarchical aggregation framework for efficient multilevel visual exploration and analysis. Semantic Web 8, 139–179. doi:10.3233/SW-160226.

Borgo, R., Kehrer, J., Chung, D.H.S., Maguire, E., Laramee, R.S., Hauser, H., Ward, M., Chen, M., 2013. Glyph-based visualization: Foundations, design guidelines, techniques and applications, in: 34th Annual Conference of the European Association for Computer Graphics, Eurographics Association. pp. 39–63.

Boriah, S., Chandola, V., Kumar, V., 2008. Similarity measures for categorical data: A comparative evaluation, in: SIAM International Conference on Data Mining, SIAM. pp. 243–254. doi:10.1137/1.9781611972788.22.

Journal Pre-proof

Chambers, J.M., Cleveland, W.S., Kleiner, B., Tukey, P.A., 2018. Graphical Methods for Data Analysis. Chapman and Hall/CRC. doi:10.1201/9781351072304.

Chen, Q., Chen, N., Shuai, W., Wu, G., Xu, Z., Tong, H., Cao, N., 2024. Calliope-Net: Automatic generation of graph data facts via annotated node-link diagrams. IEEE Transactions on Visualization and Computer Graphics 30, 562–572.

Cheng, S., Mueller, K., 2016. The data context map: Fusing data and attributes into a unified display. IEEE Transactions on Visualization and Computer Graphics 22, 121–130. doi:10.1109/TVCG.2015.2467552.

Chernoff, H., 1973. The use of faces to represent points in k-dimensional space graphically. Journal of the American Statistical Association 68, 361–368.

Chow, S., Ruskey, F., 2003. Drawing area-proportional venn and euler diagrams, in: Graph Drawing, 11th International Symposium, Springer. pp. 466–477.

Christensen, J., Marks, J., Shieber, S.M., 1995. An empirical study of algorithms for point-feature label placement. ACM Transactions on Graphics 14, 203–232.

Davison, M.L., Sireci, S.G., 2000. Multidimensional scaling, in: Tinsley, H.E.A., Brown, S.D. (Eds.), Handbook of Applied Multivariate Statistics and Mathematical Modeling. Academic Press, pp. 323–352.

Demiralp, Ç., Haas, P.J., Parthasarathy, S., Pedapati, T., 2017. Foresight: Rapid data exploration through guideposts. CoRR arXiv:1709.10513. arXiv:1709.10513.

Dennig, F.L., Joos, L., Paetzold, P., Blumberg, D., Deussen, O., Keim, D.A., Fischer, M.T., 2024. The categorical data map: A multidimensional scaling-based approach, in: Visualization in Data Science (VDS) Workshop at IEEE VIS, IEEE. pp. 25–34. doi:10.1109/VDS63897.2024.00008.

Dibia, V., 2023. LIDA: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), Association for Computational Linguistics, Toronto, Canada. pp. 113–126.

Ding, R., Han, S., Xu, Y., Zhang, H., Zhang, D., 2019. QuickInsights: Quick and automatic discovery of insights from multi-dimensional data, in: ACM SIGMOD International Conference on Management of Data, ACM. pp. 317–332. doi:10.1145/3299869.3314037.

Eskin, E., Arnold, A., Prerau, M.J., Portnoy, L., Stolfo, S.J., 2002. A geometric framework for unsupervised anomaly detection, in: Barbará, D., Jajodia, S. (Eds.), Applications of Data Mining in Computer Security. Springer. volume 6 of *Advances in Information Security*, pp. 77–101. doi:10.1007/978-1-4615-0953-0_4.

Fruchterman, T.M.J., Reingold, E.M., 1991. Graph drawing by force-directed placement. Software: Practice and Experience 21, 1129–1164.

Fuchs, J., Isenberg, P., Bezerianos, A., Keim, D.A., 2017. A systematic review of experimental studies on data glyphs. IEEE Transactions on Visualization and Computer Graphics 23, 1863–1879.

Goodall, D.W., 1966. A new similarity index based on probability. Biometrics 22, 882–907. doi:10.2307/2528080.

Hornof, A.J., 2004. Cognitive strategies for the visual search of hierarchical computer displays. Human–Computer Interaction 19, 183–223.

Huang, Y., 2026. Explanova: Automatically discover data insights in $N \times M$ table via XAI combined LLM workflow. CoRR doi:10.48550/arXiv.2601.12317, arXiv:2601.12317. arXiv:2601.12317.

Inselberg, A., 1985. The plane with parallel coordinates. The Visual Computer 1, 69–91.

Jankun-Kelly, T.J., Ma, K., Gertz, M., 2007. A model and framework for visualization exploration. IEEE Transactions on Visualization and Computer Graphics 13, 357–369. doi:10.1109/TVCG.2007.28.

Jebb, A.T., Parrigon, S., Woo, S.E., 2017. Exploratory data analysis as a foundation of inductive research. Human Resource Management Review 27, 265–276.

Jiang, Q., Sun, G., Dong, Y., Pan, L., Chang, B., Jiang, L., Liang, H., Liang, R., 2025. FactExplorer: Fact embedding-based exploratory data analysis for tabular data. International Journal of Human-Computer Interaction 41, 12361–12379. doi:10.1080/10447318.2025.2459172.

Jones, K.S., 2004. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 60, 493–502.

Joshi, A., Kale, S., Chandel, S., Pal, D.K., 2015. Likert scale: Explored and explained. British Journal of Applied Science & Technology 7, 396–403. doi:10.9734/BJAST/2015/14975.

Kandogan, E., 2001. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM. pp. 107–116.

Li, G., He, P., Wang, X., Li, R., Liu, C.H., Ou, C., He, D., Wang, G., 2025. InsigHTable: Insight-driven hierarchical table visualization with reinforcement learning. IEEE Transactions on Visualization and Computer Graphics 31, 4462–4479. doi:10.1109/TVCG.2024.3404454.

Journal Pre-proof

Li, G., Li, R., Feng, Y., Zhang, Y., Luo, Y., Liu, C.H., 2024a. CoInsight: Visual storytelling for hierarchical tables with connected insights. IEEE Transactions on Visualization and Computer Graphics 30, 3049–3061.

Li, J., Lai, C., Yuan, X., 2024b. Subspace-map: Interactive visual analysis for subspace data with a map metaphor. IEEE Transactions on Visualization and Computer Graphics doi:10.1109/TVCG.2024.3368094.

Li, S., Chen, X., Song, Y., Song, Y., Zhang, C., 2024c. Prompt4Vis: Prompting large language models with example mining and schema filtering for tabular data visualization. CoRR doi:10.48550/arXiv.2402.07909, arXiv:2402.07909. arXiv:2402.07909.

Lian, Y., Hao, J., Zeng, W., Luo, Q., 2025. A survey of visual insight mining: Connecting data and insights via visualization. Visual Informatics 9, 100271. doi:10.1016/j.visinf.2025.100271.

Lin, D., 1998. An information-theoretic definition of similarity, in: International Conference on Machine Learning, Morgan Kaufmann. pp. 296–304.

Lin, Y., Li, H., Yang, L., Wu, A., Qu, H., 2024. InkSight: Leveraging sketch interaction for documenting chart findings in computational notebooks. IEEE Transactions on Visualization and Computer Graphics 30, 944–954.

Ma, P., Ding, R., Han, S., Zhang, D., 2021. MetaInsight: Automatic discovery of structured knowledge for exploratory data analysis, in: ACM SIGMOD International Conference on Management of Data, ACM. pp. 1262–1274.

Ma, P., Ding, R., Wang, S., Han, S., Zhang, D., 2023. Demonstration of InsightPilot: An LLM-empowered automated data exploration system. CoRR arXiv:2304.00477. arXiv:2304.00477.

Müller, N.H., Liebold, B., Pietschmann, D., Ohler, P., Rosenthal, P., 2017. Hierarchy visualization designs and their impact on perception and problem solving strategies, in: International Conference on Advanced Computer-Human Interactions, pp. 93–101.

Ng, R.T., Lakshmanan, L.V.S., Han, J., Pang, A., 1998. Exploratory mining and pruning optimizations of constrained association rules, in: ACM SIGMOD International Conference on Management of Data, ACM Press. pp. 13–24.

Pienta, R., Abello, J., Kahng, M., Chau, D.H., 2015. Scalable graph exploration and visualization: Sensemaking challenges and opportunities, in: 2015 International Conference on Big Data and Smart Computing, IEEE. pp. 271–278. doi:10.1109/35021BIGCOMP.2015.7072812.

Qin, X., Luo, Y., Tang, N., Li, G., 2020. Making data visualization more efficient and effective: a survey. The VLDB Journal 29, 93–117. doi:10.1007/s00778-019-00588-3.

Reza, R.M., Watson, B.A., 2019. Hi-D Maps: An interactive visualization technique for multi-dimensional categorical data, in: IEEE Visualization Conference, IEEE. pp. 216–220.

Saeed, N., Nam, H., Ul Haq, M.I., Bhatti, D.M.S., 2018. A survey on multidimensional scaling. ACM Computing Surveys 51, 47:1–47:25. doi:10.1145/3178155.

Shen, S., Lin, Z., Mei, H., Liu, W., Xin, C., Dai, W., Chen, S., Wen, X., Lan, X., 2026. DashChat: Interactive authoring of performance dashboard design prototypes through conversation with LLM-powered agents, in: Extended Abstracts of the CHI Conference on Human Factors in Computing Systems, ACM. pp. 199:1–199:7. doi:10.1145/3772363.3798634.

Shi, D., Xu, X., Sun, F., Shi, Y., Cao, N., 2021. Calliope: Automatic visual data story generation from a spreadsheet. IEEE Transactions on Visualization and Computer Graphics 27, 453–463.

Shneiderman, B., 1992. Tree visualization with tree-maps: 2-d space-filling approach. ACM Transactions on Graphics 11, 92–99.

Silverman, B.W., 1986. Density Estimation for Statistics and Data Analysis. Springer. doi:10.1007/978-1-4899-3324-9.

Simpson, G.G., 1960. Notes on the measurement of faunal resemblance. American Journal of Science 258, 300–311.

Stasko, J.T., Zhang, E., 2000. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations, in: INFOVIS, IEEE Computer Society. pp. 57–65.

Sulc, Z., Rezanková, H., 2019. Comparison of similarity measures for categorical data in hierarchical clustering. Journal of Classification 36, 58–72. doi:10.1007/s00357-019-09317-5.

Tang, B., Han, S., Yiu, M.L., Ding, R., Zhang, D., 2017. Extracting Top-K insights from multi-dimensional data, in: ACM SIGMOD International Conference on Management of Data, ACM. pp. 1509–1524. doi:10.1145/3035918.3035922.

Tukey, J.W., 1977. Exploratory data analysis. Addison-Wesley series in behavioral science : quantitative methods, Addison-Wesley.

Vosough, Z., Hogräfer, M., Royer, L.A., Groh, R., Schulz, H., 2018. Parallel hierarchies: A visualization for cross-tabulating hierarchical categories. Computers & Graphics 76, 1–17. doi:10.1016/j.cag.2018.07.009.

Wang, H.W., Gordon, M., Battle, L., Heer, J., 2025. DracoGPT: Extracting visualization design preferences from large language models. IEEE Transactions on Visualization and Computer Graphics 31, 710–720.

Wang, L., Zhang, S., Wang, Y., Lim, E., Wang, Y., 2023. LLM4Vis: Explainable visualization recommendation using ChatGPT, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track, Association for Computational Linguistics, Singapore. pp. 675–692.

Wang, W., Wang, H.H., Dai, G., Wang, H., 2006. Visualization of large hierarchical data by circle packing, in: ACM CHI Conference on Human Factors in Computing Systems, ACM. pp. 517–520. doi:10.1145/1124772.1124851.

Wang, Y., Sun, Z., Zhang, H., Cui, W., Xu, K., Ma, X., Zhang, D., 2020. DataShot: Automatic generation of fact sheets from tabular data. IEEE Transactions on Visualization and Computer Graphics 26, 895–905.

Weng, L., Wang, X., Lu, J., Feng, Y., Liu, Y., Feng, H., Huang, D., Chen, W., 2025. InsightLens: Augmenting LLM-powered data analysis with interactive insight management and navigation. IEEE Transactions on Visualization and Computer Graphics 31, 3719–3732.

Wu, S., Luan, W., Wang, Y., Zeng, D., Shen, Q., Tang, B., 2025. Data insights as data: Quick overview and exploration of automated data insights, in: Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems, ACM. pp. 167:1–167:7.

Zhang, H., Jing, Y., Zhang, W., He, Z., Zhang, K., Wang, X.S., Chen, Z., Zhang, C., 2025. InsightCube: Accelerating interactive insight discovery in exploratory data analysis, in: IEEE International Conference on Big Data and Smart Computing, IEEE. pp. 299–306.

Zhao, H., Chen, H., Yang, F., Liu, N., Deng, H., Cai, H., Wang, S., Yin, D., Du, M., 2024. Explainability for large language models: A survey. ACM Transactions on Intelligent Systems and Technology 15, 20:1–20:38.

Journal Pre-proof

Journal Pre-proof

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Journal Pre-proof