Visual Insight Recommendation: From Ranking Insight Visualizations to Insight Types

Camille Harris
Georgia Institute of Technology
charris320@gatech.edu

Ryan A. Rossi
Adobe Research
ryrossi@adobe.com

Sana Malik
Adobe Research
sanmalik@adobe.com

Jane Hoffswell
Adobe Research
jhoffs@adobe.com

Fan Du
Databricks
dufan2013@gmail.com

Tak Yeon Lee
KAIST
takyeonlee@kaist.ac.kr

Eunyee Koh
Adobe Research
eunyee@adobe.com

Handong Zhao
Adobe Research
hzhou@adobe.com

Abstract—Visualization recommendation systems make understanding data more accessible to users of all skill levels by automatically generating visualizations for users to explore. However, most existing visualization recommendation systems focus on ranking all possible visualizations based on the attributes or encodings, which makes it difficult to find the most interesting or relevant insights. We therefore introduce a novel class of visualization recommendation systems that automatically rank and recommend both groups of related insights and the most important insights *within* each group. Our approach combines results across different learning-based methods to discover insights automatically and generalizes to a variety of attribute types (*e.g.*, categorical, numerical, and temporal), including non-trivial combinations of these attribute types. We then implemented a new insight-centric visualization recommendation system, SpotLight, which ranks annotated visualizations in visual insight groups. Finally, we conducted a user study which showed that users are able to quickly understand and find relevant insights in unfamiliar data.

Index Terms—Insight-centric visualization recommendation, insight-type recommendation, insight-type ranking, insight recommendation

Fig. 1. SpotLight is a web-based system for visual insight recommendation that automatically recommends visualizations for the most relevant insights in the data: (a) insights are automatically discovered; (b) insights are grouped into rows, scored, and ranked based on the *insight-type*; (c) these rows are then globally ranked and sorted; (d) users can interactively query recommendations through attribute filters; and (e) users can bookmark the most important insights.

I. INTRODUCTION

Visualization recommendation systems automatically suggest visualizations to support exploratory data analysis (EDA) and facilitate the visualization authoring process. These systems often leverage principles of expressiveness and effectiveness to produce recommendations that are agnostic to the specific data set. In this paper, we introduce a novel class of *insight-centric visualization recommendation systems* that automatically discover, rank, and visualize both specific insights and overarching insight-types (or classes of insights) in an arbitrary data set. Instead of displaying all visualizations together, they are grouped based on meaningful and intuitive insight-types and displayed to the user as rows of visualizations ordered by importance. The key idea here is that the system scores and recommends the top insight-types, in addition to ranking the individual visualizations within each insight-type. This ranking enables users to quickly find the most interesting and relevant insights.

There are two different but fundamentally important recommendation tasks in the proposed class of insight-centric visualization recommendation systems: (1) recommendation of the important top-r insight-types, and (2) ranking of the top-k insights within each insight-type. In essence, once we obtain the top-k recommended insights, the system can derive the top-r recommended insight-types that best summarize the most valuable insights that were identified in the data. Given an arbitrary data set, the proposed class of insight-centric recommendation systems (i) automatically discovers the important insights for each insight-type using many different learning and statistical models, (ii) scores and recommends the top-r insight-types, (iii) scores and recommends the top-k insights for every insight-type, (iv) infers an appropriate visualization for each insight, and (v) recommends a relevant insight annotation.

To demonstrate the utility of this approach, we contribute

a new insight-centric visualization recommendation system: SpotLight. SpotLight currently supports 21 different insight-types such as skew, multivariate outliers, and nonlinear correlations (Table I). The insights for every insight-type are discovered automatically using many different machine learning methods, including both supervised and unsupervised methods. Each insight-type is carefully designed to be meaningful to the user, therefore enabling them to quickly understand the relevant insights and their relative importance with respect to any arbitrary user-uploaded data set.

The main contributions of this work are summarized as follows:

Insight-centric Visualization Recommendation. We propose a novel class of insight-centric recommendation systems that enables users to quickly understand, explore, and visually identify relevant insights in their data. We further contribute a new system of this type, SpotLight, which automatically discovers, scores, and ranks the insights as well as the insight-types that are most relevant in the data set of interest. We evaluated SpotLight with 12 users to demonstrate the efficacy, and show that users can quickly uncover a diverse and useful set of insights in an unfamiliar data set.

Insight-Type Ranking. We introduce the notion of insight-type ranking and describe a general approach for recommending the most relevant insight-types for an arbitrary data set. SpotLight leverages this approach to display the top-r recommended insight-types to the user as rows of visualizations showing insights of that type, with the rows sorted based on the relevance of the underlying insights. The recommendation of insight-types (as opposed to just insights) improves the user experience making it significantly faster and easier for the user to identify the most important insights.

Insight Ranking. We propose an insight recommendation approach that derives a score for each of the discovered insights within an insight-type. This ranking enables us to derive the top-k insight recommendations for each of the different insight-types. A key advantage of our proposed insight ranking approach is that it can naturally leverage multiple insight discovery methods, attribute combinations, attribute types, visualizations, and so on.

Non-Trivial Insights and Insight-Types. The individual insights for every insight-type are discovered automatically using many different machine learning methods, including both unsupervised and supervised techniques. SpotLight can therefore recommend non-trivial insights that have complex attribute combinations. Furthermore, SpotLight naturally handles a wide variety of different attribute types including numerical, categorical, and temporal. The recommended insights for a specific insight-type can also have a varying number of attributes and attribute types, thus allowing SpotLight to produce a diverse set of recommended insights.

II. RELATED WORK

In this section, we introduce related work on insight discovery and visualization recommendation, which informs our classifi-

cation of visualization recommendation systems introduced in Section III.

A. Insight Discovery

There has been some related work on insight discovery, focusing both on a specific use case and type of insight, or on the general problem [1]. For example, prior work has contributed novel approaches to discovering insights including correlations [2]–[5], outliers and anomalies [6]–[11], and peaks [12], as well as more general insight discovery techniques [13], [14]. Related work often focuses on specific use cases, such as insight discovery for email data [15] or geovisualization [16]. For example, NewsViz [16] offers an automatic system to generate interactive geovisualizations for news articles. QuickInsights [17] proposes an algorithm to extract the most useful insights from data and uses pruning to remove obvious insights and best-first prioritization. Agrawal et al. [18] propose a general framework for efficiently discovering association rules in data. Amar et al. [13] propose a set of 10 analysis task types which visualization recommendation systems can use to produce effective visualizations. Scorpion [19] proposes an approach to reveal qualities in the data that may explain outliers. Voder [20] helps people explore data through manual view specification. Once users create a visualization, Voder utilizes a set of predefined heuristics to generate a list of related data facts (insights). DataSite [21], offers a system that identifies and visualizes data facts based on pre-defined methods. However, all of this related work focuses on specific insight-types, customized insight discovery methods, or particular data types. In this paper, we contribute a general-purpose solution to insight discovery and ranking, which leverages many of the techniques proposed by related work. Section IV further outlines the specifics of our problem formalization.

B. Visualization Recommendation

In this paper, we consider three primary types of visualization recommendation systems, namely, rule-based [22]–[28], ML-based [21], [29]–[34], and insight-based systems [35]. Section III describes the general system properties considered for this classification. Related work has also discussed frameworks for choosing the ideal visualizations for specific data [36] and providing a general overview of how to approach visualization recommendation based on other recommendation system techniques [37].

Rule-based Systems. Most previous work on visualization recommendation has focused on rule-based systems [38]–[40] where visualizations are recommended based on a series of manually defined if-then statements [26]–[28], [41]. Voyager [41] proposes a hybrid approach that incorporates rule-based decisions with support for user selected variables (columns). In addition, some related work has proposed different types of visualization generation tools [22]–[24] as well as tools to help users produce their own visualizations, such as VizAsisst [42] and Tableau [43].

ML-based Systems. VizML [30] focuses entirely on design choice classification (*e.g.*, predicting the chart type),

and not other more challenging problems of visualization recommendation, such as selecting the appropriate variables for the visualization. Data2Viz [44] uses an end-to-end neural translation model to map data specifications to visualizations. DeepEye [45] combines rule-based visualization generation techniques similar to work in the previous paragraph with classification models to rank visualizations and classify them as "good" or "bad." DataShot [46] creates a data fact sheet by analyzing the data for the most interesting data facts and generating visualizations from them. Draco [47] extends the rule-based system of Voyager by learning weights for manually defined constraints. More recently, Xin et al. [29] proposed an end-to-end learning-based visualization recommendation model that is trained using a corpus of hundreds of data sets and visualizations; the learned model can be applied to recommend visualizations given any arbitrary unseen data set of interest. Other recent work has focused on learning a personalized visualization recommendation model called PVisRec [34] that recommends interesting and highly relevant visualizations for a user based on their past data and visualization preferences. However, none of this related work focuses on recommending insights, and thus solves a different problem.

Insight-based Systems. Insight-based systems recommend visualizations based on properties of the specific data set that will be visualized. Our system SpotLight is an example of this type. This space has largely been under explored, but the work closest to our own is Foresight [35]. Foresight is an early example of an insight-based system that employs a single heuristic insight method to identify insights based on the marginal distribution of the attributes. Foresight then selects a manually defined visualization based on the insight-type. In contrast, we propose a more general approach to insight-centric recommendation and ranking, and instantiate the techniques in our own system, SpotLight. SpotLight automatically creates a novel two-dimensional ranking of insight-types and individual insights, that leverages many different learning-based insight discovery methods to produce a robust ranking. Furthermore, our proposed approach generalizes to a wider variety of attribute types and a diverse set of visualization designs. Notably, this work contributes a novel approach for ranking and recommending both individual insights and overarching classes of insight-types to facilitate rapid identification of relevant insights in an arbitrary data set.

III. CLASSES OF RECOMMENDATION SYSTEMS

In this paper, we introduce a novel class of *insight-centric* visualization recommendation systems, that goes beyond the existing *design-centric* and *preference-centric* classes explored in prior work. As a whole, visualization recommendation systems aim to automatically generate visualizations, often based on some form of user input. However, each specific class of visualization recommendation system employs unique characteristics to control the recommendation process. In this section, we describe the key differences between the classes of visualization recommendation systems and propose the unique benefits posed by insight-centric approaches.

Fig. 2. Recommending visual insights from each insight-type. Insight-types are also ranked and displayed accordingly to make it easy for the user to quickly understand the most important insights in their data.

Design-centric systems leverage information about the underlying data types (*e.g.*, nominal, ordinal, quantitative) to recommend visualizations based on principles of visualization effectiveness and expressiveness. The goal in such systems is to create "good" visualization designs by default to help users rapidly create visualizations. However, such systems are based entirely on principles of visualization design, and remain agnostic to the underlying data distributions aside from a basic consideration of data types.

Preference-centric systems go one step further to incorporate new knowledge that has been learned about the visualizations and data preferences. Such systems may employ machine learning techniques to assign weights based on properties like the underlying chart and/or data types, or to create a global ranking of visualization designs. While details of the underlying data set may surface during this process, the data itself is still not considered directly.

Insight-centric systems aim to incorporate details of the specific underlying data that will be used to create the visualizations. Whereas the other classes of visualization recommendation systems are largely agnostic to the underlying data values, insight-centric systems incorporate the specific data properties as an essential feature of the recommendation process. Such systems can therefore highlight not only the most relevant visualizations, but also summarize the most important classes of insight-types to provide a high-level overview of the shape of the underlying data. We provide additional details about the exact problem formulation for insight-centric recommendation systems in the next section.

TABLE I
WE SUMMARIZE THE INSIGHT-TYPES, AND CORRESPONDING ATTRIBUTE TYPE COMBINATIONS AND CHART TYPES SUPPORTED FOR EACH INSIGHT. WE ALSO SUMMARIZE THE TECHNIQUES USED IN THE PROPOSED FRAMEWORK FOR AUTOMATICALLY REVEALING THESE INSIGHTS, AND COLOR EACH ONE BASED ON THE APPROACH USED FOR DISCOVERING THE INSIGHTS (information theoretic, statistical, supervised learning, unsupervised learning). N=NUMERICAL, C=CATEGORICAL, T=TIME.

Insight Type	Attribute Type Combinations	Visualization Type	Methods
Timeseries Outliers	$N \times T$	timeseries	autocorrelation · IForest · one-class SVM
	$N \times N \times T$	multiline timeseries	IForest · one-class SVM · LOF
	$C \times N \times T$	bar timeseries	temporal entropy · IForest · one-class SVM
Single Variable Outliers	N	boxplot	z-score · IQR · DBSCAN · LOF · one-class SVM
	C	bar	entropy
Two Variable Outliers	$N \times N$	scatterplot	LOF · DBSCAN · k-means · IForest · one-class SVM
	$C \times C$	stacked bar	PMI · ks-stat · IForest
Multivariate Outliers	$N \times ... \times N$	colored scatterplot	CBLOF · k-means · IForest · one-class SVM
	$C \times N \times ... \times N$	stacked bar	CBLOF · IForest
Timeseries Correlation	$N \times T$	timeseries	autocorrelation
Multivariate Timeseries Correlation	$C \times N \times T$	stacked bar timeseries	temporal entropy
	$N \times N \times T$	multiline timeseries	cross-correlation · cosine tdiff · Spearman tdiff
Two Variable Correlation	$N \times N$	scatterplot	Pearson correlation · PCA · decision tree regression
	$N \times C$	violin plot	logistic regression · decision tree classification
	$C \times C$	stacked bar	Theil's U · Cramer's V
Nonlinear Correlation	$N \times N$	scatter with LOESS	decision tree regression
	$N \times C$	bar	decision tree classification · logistic regression
	$C \times C$	stacked bar	Theil's U · Cramer's V
Rank Correlation	$N \times N$	scatterplot	Spearman · Kendall · Goodman-Kruskal's γ
Timeseries Causality	$N \times N$	multiline timeseries	Granger causality
Stability	$N \times N$	timeseries	augmented Dickey-Fuller
Seasonality	N	timeseries	CH test
Spikes	N	timeseries	max local differences
Peaks	N	timeseries	k peaks of support
Irregularity	N	timeseries	augmented Dickey-Fuller
Skew	C, N	histogram	unbiased skew
Heavy Tails	C, N	histogram	kurtosis
Multimodality	C, N	histogram	bimodality coefficient
Dispursion	C, N	histogram	variance
Missing Values	C, N	pie	percent missing vaules
Joint Missing Values	$(C, N) \times (C, N)$	pie	percent missing vaules per attribute pair

IV. PROBLEM FORMULATION

An insight is a new piece of knowledge identified in the data while performing an analytical task. In other words, an insight is a property of the data that is unexpected, complex, deep, or relevant to the analyst, such as a strong linear correlation between two variables or a set of points that are anomalies with respect to time. Providing insights is arguably the main goal of information visualization [14]. However, none of the existing visualization recommendation systems described in prior work focus on the recommendation of classes of insight-types or the individual ranking of visual insights.

Definition 1 (VISUAL INSIGHT RECOMMENDATION).
Let $\mathcal{I} = \{I_1, I_2, \ldots, I_{|\mathcal{I}|}\}$ denote the set of insight-types. Given an arbitrary data set $\mathbf{X}$, we define $\mathcal{F}_{I_i} = \{f_1, f_2, \ldots\}$ as the set of insight discovery methods for insight-type $I_i \in \mathcal{I}$ and $\rho_{I_i}(\mathbf{X}_1, \mathbf{X}_2, \ldots)$ as the set of insights found for that insight-type. An insight-centric visualization recommender system (i) automatically discovers the important insights for each insight-type $I_i \in \mathcal{I}$ using many different learning and statistical models $\mathcal{F}_{I_i}$ for many different attribute types and attribute type combinations; (ii) recommends the important top-r insight-types; (iii) scores and recommends the top-k insights $\rho_I(\mathbf{X}_1, \ldots, \mathbf{X}_k)$ for every insight-type $I_i \in \mathcal{I}$ and attribute type combination; and (iv) infers an appropriate visualization for each insight.

Every insight-type $I_i \in \mathcal{I}$ has a set of attribute type combinations denoted as $\mathcal{A}_I$. For each attribute type combination $A \in \mathcal{A}_I$, we have a set of insight discovery methods $\mathcal{F}_I$ as shown in Table I. For example, the "Two Variable Outliers" insight-type has two different attribute type combinations $\mathcal{A}_I = \{N \times N, C \times C\}$; each attribute type combination then supports several methods for detecting insights (*e.g.*, for $N \times N$, the supported methods include DBScan and IForest, among others). Furthermore, we also recommend many different visualizations for every insight type $I \in \mathcal{I}$, as shown in Table I.

V. Visual Insight Recommendation

Given the problem formulation from Sec. IV, we now describe our proposed approach for visual insight recommendation.

A. Auto-Insight Discovery

The first step in the recommendation process is to automatically detect insights in an arbitrary data set. To effectively identify insights for any input data set, we employ a variety of different insight discovery methods as described next.

1) Multiple Insight Discovery Methods: Given an arbitrary data set, each insight-type (and attribute combination) has many potential algorithms and the best algorithm depends on the data and its characteristics [48], [49]. The above is due to the no-free lunch theorems by Wolpert *et al.* [48], [49], which essentially means that any two optimization algorithms are equivalent when averaged across all possible problems (data sets) [50], [51]. Since the "best" insight discovery method depends on the data and its characteristics [52], we instead use multiple insight discovery methods for each insight-type and attribute combination to ensure that our approach can find important insights, regardless of the user selected data set.

2) Learning-based Insight Discovery Methods: In addition to the simple heuristic-based insight methods used in Foresight (*e.g.*, IQR), we use learning-based insight methods for many of the different insight-types as shown in Table I. Many of the machine learning methods in this framework can be configured with different kernel and distance functions such as One-Class SVM, Kernel PCA, k-means, among many others. These methods also have many other real-valued hyperparameters that can be optimized. Therefore, we sample a few models from each method to use in the ranking of the different insights (*e.g.*, One-Class {linear, RBF, polynomial}-SVM).

3) Insight-Type Granularity: As the number of different insight-types in visual insight recommendation systems becomes large, the insight-type ranking and recommendation that this work proposes becomes even more important. Intuitively, the more insight-types the visual insight recommendation system incorporates, the higher the probability of finding relevant insights that are important to the user and data set of interest. SpotLight also detects insights at a fine-granularity that are more meaningful and intuitive to the user. For instance, instead of considering a general insight-type called "outliers" as done in prior work [35], we distinguish outliers into meaningful subcategories, such as time-series outliers, multivariate outliers, single variable outliers, and so on (Table I).

4) Multi-Attribute Types and Combinations: For each insight-type shown in Table I, we consider multiple attribute combinations and many different attribute types. The expressiveness of this approach contrasts with prior work [35], which only considers a single attribute combination and only supports numerical attributes.

5) Multi-faceted Insight Types: Many insight-types support different attribute combinations ($N \times C$, $C \times C$), as well as combinations consisting of a different number of attributes, *e.g.*, $N \times N$ and $N \times N \times C$. One such example in Table I is timeseries outliers.

6) Methods: In this work, we use many different methods to automatically discover insights for each insight-type. We provide a summary of these methods in Table I. We categorize these methods into four main classes, including (i) **information theoretic**, (ii) **statistical**, (iii) **unsupervised learning**, and (iv) **supervised learning**. We adapt and leverage many of these methods in nonstandard ways for learning and automatically discovering insights from a wide range of different insight-types. To do this, we often have to first derive a different representation of the data, modify or change entirely the scoring function, etc. As such, we focus our discussion on the fundamentally different and novel ways these techniques were used in our insight recommendation framework. Importantly, the proposed insight recommendation framework is flexible as the insight discovery methods used for each insight-type and attribute type combination are completely interchangeable. Furthermore, this approach can generalize to any arbitrary set of methods that users may want to employ in the future.

B. Insight Ranking

Once insights have been discovered, the next step is to create a ranking of insights to surface for the user. Previous work used a single heuristic for each insight-type and therefore did not require any scoring or ranking metric. In contrast, our approach automatically reveals insights of a specific type using multiple learning-based methods, which makes scoring and ranking the insights non-trivial. Ranking becomes even more complex given that the insights from an insight-type can leverage a different number of attributes of different types and combinations as shown in Table I, which makes comparing and scoring individual insights difficult.

Given an insight-type $I \in \mathcal{I}$ and set of methods $\mathcal{F}_I$ for that insight type, the goal is to assign a score to each insight detected using $\mathcal{F}_I$. Suppose $p_I = |\mathcal{F}_I|$, then we may have p_I scores for every insight detected for insight type $I \in \mathcal{I}$. We define the utility function ϕ as:

$$\phi : \mathcal{X} \times \mathcal{F}_I \to \mathbb{R} \qquad (1)$$

where $\mathcal{X}$ is the space of attribute combinations used by the methods $\mathcal{F}_I$ for insight-type $I \in \mathcal{I}$. We derive a final score as:

$$\phi(\mathbf{X}_k, \mathcal{F}_I) = \frac{1}{Z} \sum_{f_i \in \mathcal{F}_I} \sum_{j=1}^{n} \big[g\big(f_i(\mathbf{X}_k, \Lambda_i)\big) \big]_j \qquad (2)$$

where $\mathbf{X}_k$ is an arbitrary attribute matrix consisting of one or more attributes from $\mathbf{X} \in \mathbb{R}^{n \times m}$. In other words, $\mathbf{X}_k$ is an attribute combination matrix that may consist of one or more attributes. Further, $\big[g\big(f_i(\mathbf{X}_k)\big) \big]_j$ is the jth value from $\mathbf{s} = g\big(f_i(\mathbf{X}_k)\big) \in \mathbb{R}^n$. We define $g : \mathbb{R} \to [0, 1]$ and Λ_i as the set of hyperparameters for the learning-based insight method $f_i \in \mathcal{F}_I$. Suppose $f_i \in \mathcal{F}_I$ is one-class SVM, then Λ_i may include a kernel function $\mathbb{K}$ such as the non-linear RBF kernel or polynomial kernel along with other hyperparameters, such as γ or the degree of the polynomial kernel. In Eq. 2, $Z = |\mathcal{F}_I| n$.

Our formulation in Eq. 2 assumed that the output of each insight method was the same. However, in general, some insight
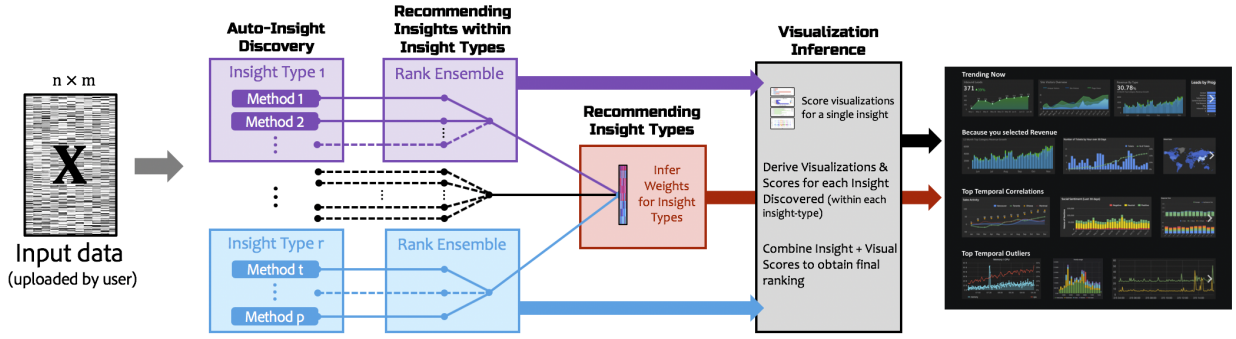
Fig. 3. Overview of the proposed class of Insight-centric Visualization Recommendation Systems.

methods may return scores for only the most relevant data points (as opposed to all n data points), or even a single score for the attribute combination. Thus, we can rewrite Eq. 2 as

$$\phi(\mathbf{X}_k, \mathcal{F}_I) = \frac{1}{|\mathcal{F}_I|} \sum_{f_i \in \mathcal{F}_I} \frac{1}{n_i} \sum_{j=1}^{n_i} \left[g\big(f_i(\mathbf{X}_k, \Lambda_i)\big) \right]_j \quad (3)$$

where n_i denotes the number of scores returned by f_i: Eq. 3 assigns an insight score to the attribute combination $\mathbf{X}_k$ for insight type $I \in \mathcal{C}$ using the methods $\mathcal{F}_I$. Now, we obtain a ranking of the insights within the insight type $I \in \mathcal{C}$ as follows:

$$\rho_I\big(\{\mathbf{X}_1, \dots, \mathbf{X}_k, \dots\}\big) = \underset{k}{\arg\operatorname{sort}} \; \phi(\mathbf{X}_k, \mathcal{F}_I) \quad (4)$$

There are a number of novel aspects in the above formulation for ranking insights within an insight-type. More specifically, we are the first to (i) leverage learning-based insight methods, (ii) use more than a single method per insight, and (iii) propose and require an insight scoring and ranking function. Moreover, we describe a general mathematical framework for recommending visual insights.

Instead of using the normalized scores from every method to derive a final score for a given insight-type and attribute combination, we derive a final score for each data point (row of $\mathbf{X}_k$) based on the rankings given by each method. Using the ranks as opposed to the weights can help avoid biasing certain methods depending on the distribution of inferred weights. For a single j and attribute combination $\mathbf{X}_k$, we can generate a rank-based score as follows:

$$R_j(\mathbf{X}_k, \mathcal{F}_I) = \frac{1}{|\mathcal{F}_I|} \sum_{f_i \in \mathcal{F}_I} \pi_j\big(f_i(\mathbf{X}_k, \Lambda_i)\big) \quad (5)$$

where $\pi_j\big(f_i(\mathbf{X}_k, \Lambda_i)\big)$ is the position of the jth object in the ranking obtained from attribute combination $\mathbf{X}_k$ with method $f_i \in \mathcal{F}_I$. Therefore, $R_j(\mathbf{X}_k, \mathcal{F}_I)$ is the average rank of j across all methods and every method can be seen as having equal weight. Notice that Eq. 5 gives an overall ranking for each data point j (row in $\mathbf{X}$) whereas Eq. 4 provides a ranking of the overall attribute combinations $\{\mathbf{X}_1, \dots, \mathbf{X}_k, \dots\}$ across all n data points and $|\mathcal{F}_I|$ methods.

The rank-based score given by Eq. 5 can be used to appropriately annotate the visualizations for better visual insight

recommendation. As an example, suppose we use the set of two variable outlier methods and use the methods as an ensemble to obtain an overall ranking of the data points by how much of an outlier each point appears to be. This ensemble ranking can be accomplished using the average rank of the data points given by the set of methods.

Note that if g is set to the min-max norm for each insight-type and attribute type combination, then by definition we are guaranteed to have a diverse ranking of visualizations for each insight-type. Intuitively, since min-max norm is applied to each attribute type combination independently, then one of the insights with that attribute type combination is guaranteed to score 1. Hence, if there are three attribute type combinations for a given insight-type, then the first three insights will be of different attribute type combinations.

1) Space & Scalability: To ensure the approach is space-efficient and scalable for large datasets with millions or more data points, we do not store the individual scores of each data point (for all methods). Notably, we can simply compute the final score on-the-fly while taking a scan over the (sampled) attribute values. For large data where even a linear scan over the data points is considered too costly, we can use sampling or sketching techniques.

C. Scoring and Ranking Insight Types

The insight-types are scored in a completely automatic and data-driven fashion. In particular, the ranking of the different insight-types (rows of insights and their visualizations) is driven by the amount of information captured by each insight-type across all the discovery methods. Let $Q_I = \{\mathbf{X}_1, \dots, \mathbf{X}_k, \dots\}$ denote the set of potential insights being scored for insight type $I \in \mathcal{I}$, then

$$\Psi(I) = \frac{1}{|Q_I|} \sum_{\mathbf{X}_k \in Q_I} \phi(\mathbf{X}_k, \mathcal{F}_I) \quad (6)$$

where $\Psi(I)$ is the overall score assigned to insight type $I \in \mathcal{I}$ for the dataset as a whole. As defined by Eq. 6, if an insight-type $I \in \mathcal{I}$ receives a relatively high score $\Psi(I)$, then there must be many important and highly weighted insights of that insight-type. The overall insight-type score is based on the insights discovered in the specific data set and their corresponding

scores previously derived in Section V-B. Using Eq. 6, we derive a global ranking of the insight types $\mathcal{I}$ by simply sorting them based on their overall scores:

$$\rho_{\mathcal{I}}\big(\{I_1, I_2, \ldots, I_{|\mathcal{I}|}\}\big) = \underset{I \in \mathcal{I}}{\arg\operatorname{sort}} \ \Psi(I) \qquad (7)$$

The insight-types are displayed to the user according to the ranking from Eq. 7. This ranking enables the user to quickly find the most relevant insights for the data set of interest. Furthermore, the overall insight-type ranking for a data set can be used to better understand the data quickly, *e.g.*, if the most important insight-types for a specific data set are all related to time-series (such as time-series outliers, time-series causality, and so on), then the user immediately knows that the temporal dimension in the data is important.

D. Additional Features of SpotLight

We now discuss a few other important components.

1) Multiple Visualizations per Insight-Type: The next step is to derive weights (either learned or rule-based) for the potential visualizations for each of the top-k insights of a specific insight-type learned by our proposed approach. For each insight, we derive weights for the potential visualizations and select the best visualization (rank-1). With this approach, notice that we could easily select and display more than one visualization for a given insight. This option may be preferable if there is an insight that is significantly more important than others. In this case, showing several different visualizations for the same insight may be more useful to the user than showing less relevant insights or visualizations.

2) Data Complexity: Given an insight from an arbitrary insight-type, we add a penalty to overly complex and data-intensive insights. For instance, an insight involving time, a categorical attribute, and two numerical attributes will obviously be more difficult for a user to understand and visualize than one involving only two attributes.

3) Human-in-the-Loop: The system also leverages intuitive UI features that allow the user to specify attributes of interest and perform many other types of interactive insight queries as well. Importantly, the recommended insight-types and visual insights within each type are automatically adapted based on the user input such as the attributes of interest.

VI. EVALUATION

To understand the efficacy, accuracy, and generalizability of the proposed class of insight-centric recommendation systems, we conducted a user study of the system with twelve participants. The user study is designed to answer the following questions:

RQ1: Does SpotLight help users generate hypotheses, find insights, and understand the data better?

RQ2: Are the supported insight-types and attribute types useful, diverse, and comprehensive?

RQ3: Are the recommended rankings (of both insight-types and insights) useful and do they provide diversity?

RQ4: Are the visualizations used for the recommended insights useful, diverse, aesthetic, and easy-to-understand?

RQ5: Does the UI allow users to find the most interesting insights and does it support different workflows for EDA?

A. Method

Participants. Participants were recruited by word-of-mouth and internal mailing lists at a software company and two universities. In total, there were 12 participants aged 25–51 years ($mean = 34$). Participants consisted of two graduate students, five researchers/data scientists, four technical professionals, and one visualization designer. Participants had 3–20 years of data analysis experience and 1–20 years visual analytics experience ($mean = 8$ and 7 respectively). Two participants were female and ten were male.

Procedure. Sessions were conducted remotely, through video conference and screenshare, with two researchers (one administered, one took notes). Participants were first trained on SpotLight's UI and features using a training data set. Participants were then asked to perform a free-form exploration of two data sets from the UCI ML Repository: `Weather` data and `Wholesale` customer profile data. No participant was familiar with either data set ahead of time and the order of the data sets was counterbalanced between participants to avoid fatigue and learning effects. During exploration, participants were encouraged to think-aloud and bookmark interesting charts that helped them understand the data. After exploring the data, participants were asked to verbally present their findings. Afterwards, participants completed a post-questionnaire with twenty-nine 7-point Likert questions and three questions about the usability and utility of SpotLight. The bookmarked insights were also logged for each participant. Participants were required to use a desktop or laptop computer instead of mobile or tablet devices, for a consistent experience. Each session lasted 60–75 minutes.

Data sets. The `Weather` data set consisted of weather patterns for two US cities, Seattle and New York, over four years (2014–2017). Each row represented one day for each city, with measurements for: precipitation (*in*), wind (*mph*), minimum and maximum recorded temperatures (*C*). Lastly, there was a categorical variable indicating the type of weather seen on that day, such as *sunny, fog, drizzle, rain,* or *snow.* The `Wholesale` data set consisted of 440 businesses' annual spending at a wholesale company. Each row represented one customer, with a categorical variable representing the type of business (*hospitality* or *retail*) and six numerical values representing annual spending broken down by department: *fresh, frozen, grocery, detergent and paper, dairy,* and *deli.*

Analysis. Sessions were analyzed by two researchers to observe participants' patterns in workflow and their comments about their use and understanding of the system. The insights users bookmarked were saved and analyzed based on frequency and ranking in the recommendations. In particular, since the insights within every insight-type are ordered automatically by SpotLight, we compared the ranking of insights given

by SpotLight to the "ground-truth" ranking given by the insights users bookmarked. We used the bookmarked insights to quantitatively evaluate the insight-type ranking derived by SpotLight compared to the ranking of insight-types given by the bookmarked insights. Finally, the post-questionnaire Likert responses were analyzed with descriptive statistics and the open-ended responses were thematically coded.

B. System Usage

To gather preconceptions about the data and to confirm their understanding of the data sets, we asked participants what they expected or would find interesting to see in each of the data sets. Participants tended to have more hypotheses about the weather data set than the wholesale data set because it was more natural to have experience with weather. For example, nine participants hypothesized city-based comparisons (*e.g.*, "Seattle is rainier than New York") and half of the participants were interested in variable-based correlations (*e.g.*, whether temperature and precipitation have a relationship). Additionally, the types of hypotheses people had were different based on the data set. Wholesale hypotheses were less focused on comparisons and were more related to the relationships between different spending variables in general, though some did also focus on comparing customer types (*e.g.*, hospitality vs. retail).

The hypotheses also impacted participants' workflow when using the system. Most participants (10/12) followed a general exploratory workflow: they began by reviewing the variables in the data set using the variable search bar then explored the recommended insights row-by-row. Most opted to explore the results from top to bottom, citing trust in the recommendations: *"Since the system suggested this as the most insightful, I'll go ahead and start with that"* (P1). After reviewing the top recommendations and generating additional hypotheses, some participants filtered the results based on specific variables they were interested in analyzing. Conversely, many participants (5/12) with clear hypotheses began their analysis by first filtering by the variables relevant to their ideas to find insights that confirmed or denied their hypotheses. Two participants (P1, P5) systematically searched for variables one-by-one to analyze subgroups of the results and to understand the data set better; for example, one participant noted that *"Maybe I need to first look at one variable at a time, so I can understand the dataset first. It's a lot cleaner and easier to see initially with one variable"* (P1).

C. Insight and Insight-type Results

We analyzed the rank of users' bookmarked insights to quantitatively evaluate the effectiveness of the insight and insight-type rankings of our proposed approach.

Insight Ranking. Across all insight-types, 68% (weather) and 61% (customer wholesale) of the insights bookmarked by the users came from the top-5 insights recommended by SpotLight in their respective insight-type rows. This finding indicates that the ranking of insights (within each insight-type) from SpotLight were indeed useful and validates the effectiveness of the ranking of insights and the top-recommended insights

within each insight-type given by the proposed approach. Considering the top-10 insights recommended by SpotLight (as opposed to only the top-5), we find that among all the bookmarked insights from the participants, 91% (weather) and 86% (customer wholesale) of them can be found in the top-10 ranked insights from SpotLight. This observation is important, since there is clearly an exponential amount of visual insights that can be recommended for every different insight-type, and the above finding demonstrates that SpotLight is able to accurately score and recommend the best insights with the highest quality/relevance from each insight-type. In addition, users bookmarked multiple chart-types per insight-type, indicating that the diversity of charts was useful. This observation is particularly important since the proposed insight ranking function biased charts of different types towards the top of the ranking. We posit that this behavior was welcomed by the users for many reasons. For instance, if they do not understand certain chart types, then they can immediately see the variety of different charts used for the insight-type. Furthermore, if users found a particular chart type useful for a given insight-type, they could quickly find the insights that had such a chart.

Insight-Types. We leverage the insights bookmarked by the users to (i) investigate the utility of the insight-types, and (ii) quantitatively evaluate the *insight-type ranking*. We begin by first evaluating whether the automatic insight-type ranking proposed in this work is important or not. To do this, we compare the top-5 insight-types ranked by the number of insights bookmarked from each insight-type across both weather and customer wholesale data. We use Kendall's τ rank correlation to compare the top-5 insight-types from either data set. Strikingly, we find the ranking of the top-5 insight-types for weather data are significantly different at p-val=0.05 from those preferred by participants in the customer wholesale data. Hence, ranking the insight-types is indeed useful, since the insight-types for either data set were ranked differently by the users. In other words, if the ranking of insight-types did not matter, then the ranking should not depend on the data set of interest, and thus be the same. This result indicates the importance of ranking the insight-types differently depending on the user-selected data set, the important insights discovered in the data, as well as the behavior and preferences of the user. In addition, we found that users bookmarked insights from every insight-type but one (heavy tails). This result holds across both data sets and indicates that nearly all insight-types proposed in SpotLight are useful.

Insight-Type Ranking. To quantitatively evaluate the effectiveness of the insight-type ranking, we compare the ranking of insight-types from SpotLight to the ranking given by the participants (based on their bookmarked insights) for the wholesale data. The ranking of insight-types from the participants are used as ground-truth. To test whether the insight-type ranking given by SpotLight is similar to the ranking given by the participants in the study, we use Kendall's τ rank correlation. Importantly, we find the insight-type ranking from SpotLight to be significantly correlated with the ground-
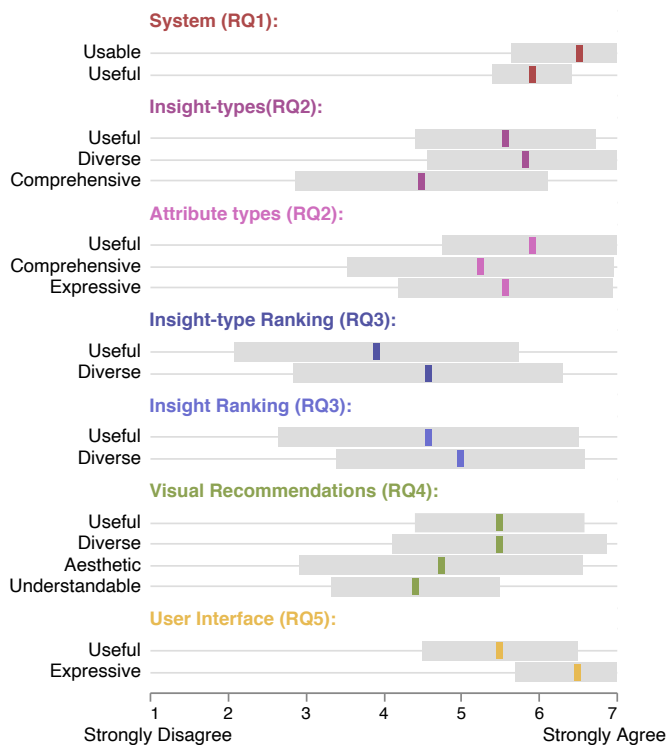
Fig. 4. Overall ratings of SpotLight on a 7-point Likert scale, with one standard deviation shown in gray.

truth ranking from the users in the study (with a p-val<0.01). Notably, the rank correlation coefficient $\tau = 0.87$ indicates that the two rankings are extremely correlated with a p-value of 0.0008. This result quantitatively demonstrates the effectiveness of the insight-type ranking as the ranking given by SpotLight is correlated significantly with the user-preferred ranking.

D. Participant Feedback

Overall feedback based on the questions is provided in Fig. 4. Participants were generally positive about the system and appreciated that it was comprehensive and automated when recommending insights. Even experts appreciated the ease-of-use, noting that *"I do a lot of this by hand using Python so I definitely can see value in some immediate analysis. I end up generating similar graphs but each takes time (and learning in Python)"* (P3) and *"I like that the visualizations are automated for me, so it's great to just bookmark and view them"* (P1). Overall, participants felt the system would be **useful (mean = 5.9)** and **usable** for improving participants' understanding of the data (**mean = 6.3**). P5 noted that it was *"a good way to start my orientation in a dataset"*. P3 explained how the system provided *"immediate insights"* and could *"save a lot of time exploring the dataset to understand it"*.

Participants were mixed about the **comprehensiveness** of the insight-types (**mean = 4.5**). Most were able to find insights from the data based on the available insight-types, especially for insights that they would not have otherwise found. P2 noted that the system included insight-types that they would not think

to use: *"I wouldn't think to correlate the numerical values with each other off the top of my head, but that makes sense"*. P7 similarly explained that *"a number of analyses listed were very useful in parsing out trends about the overall data"*. Some participants expressed interest in insights related to specific slices of the data; P2 suggested adding *"comparisons between the categorical variables"* and P8 expressed a need for filtering by categorical values: *"I don't know how I could select data from only variable (like city) and then see the results"*.

The visualizations were rated highly (**mean = 5.5**), especially with the inclusion of annotations, which *"help a lot to explain [the insights]"* (P2). Many participants commented about the **diversity of chart types**, citing this feature as a major strength of the system; P11 explained that *"The system provide more advanced visualizations that other tools provide"* and P7 noted that *"Some of the graphs were very visually appealing and indicated trends across different categories very clearly. Definitely worth using this tool for those graphs alone"*.

The most common suggestion for improvement was to include more descriptive statistics and explanation of the insights used. Participants felt that the use of insight-type were too technical for people without significant statistics background: *"Explanations about visualizations or statistical analysis would help me better explain what each chart is trying to tell me"* (P3). Participants also suggested common improvements on the visualizations themselves, such as enforcing consistency between visualizations (*e.g.*, color), providing interactions such as zooming, and details-on-demand to allow for analysis beyond EDA. These features in particular would help users in the next step of their investigative analysis, especially as they generate additional hypotheses when comparing between multiple charts. Lastly, many participants (6/12) wanted more control over the visualizations (*e.g.*, customizing the binning and configuring chart variables and axes).

VII. CONCLUSION

In this work, we proposed a visual insight recommendation system that makes it easy and fast for users to find interesting insights in their data visually. Given an arbitrary data set, we first recommend the top-k insights and accompanying visualizations for each insight-type (*e.g.*, time-series outliers). The visual insights for each insight-type are ranked and displayed to the user in a single row. More importantly, to enable the user to find interesting insights quickly, we also score and rank the insight-types where each insight-type corresponds to a row of visual insights. By recommending the top-r insight-types (rows of visual insights), users can easily identify important and relevant insights in their data quickly. Finally, future work will investigate personalizing the insight-type ranking and ranking of insights within each insight-type based on user behavior logs (*e.g.*, bookmarked insights, the insight-types of those bookmarked insights, insights a user clicked on, and so on).

REFERENCES

[1] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry, "Task taxonomy for graph visualization," in *Proceedings of the BELIV Workshop*, 2006, p. 1–5.

[2] W. Xiao, "An online algorithm for nonparametric correlations," *arXiv:1712.01521*, 2017.

[3] Y. Wang, S. Romano, V. Nguyen, J. Bailey, X. Ma, and S.-T. Xia, "Unbiased multivariate correlation analysis," in *AAAI*, 2017.

[4] Q. Wang, Y. Shen, and J. Q. Zhang, "A nonlinear correlation measure for multivariable data set," *Physica D: Nonlinear Phenomena*, vol. 200, no. 3-4, pp. 287–295, 2005.

[5] Y. Wang, Y. Li, X. Liu, W. Pu, X. Wang, J. Wang, M. Xiong, Y. Y. Shugart, and L. Jin, "Bagging nearest-neighbor prediction independence test: an efficient method for nonlinear dependence of two continuous variables," *Sci. Reports*, vol. 7, no. 1, pp. 1–12, 2017.

[6] Z. He, X. Xu, and S. Deng, "Discovering cluster-based local outliers," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1641–1650, 2003.

[7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[8] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier detection for temporal data: A survey," *TKDE*, vol. 26, no. 9, pp. 2250–2267, 2013.

[9] M. B. Duggimpudi, S. Abbady, J. Chen, and V. V. Raghavan, "Spatio-temporal outlier detection algorithms based on computing behavioral outlierness factor," *Data & Knowledge Engineering*, vol. 122, pp. 1–24, 2019.

[10] A. Duraj and Ł. Chomatek, "Outlier detection using the multiobjective genetic algorithm," *Journal of Applied Computer Science*, vol. 25, no. 2, pp. 29–42, 2017.

[11] Q. Liu, R. Klucik, C. Chen, G. Grant, D. Gallaher, Q. Lv, and L. Shang, "Unsupervised detection of contextual anomaly in remotely sensed data," *Remote Sensing of Environment*, vol. 202, pp. 75–87, 2017.

[12] G. Palshikar *et al.*, "Simple algorithms for peak detection in time-series," in *Proc. 1st Int. Conf. Advanced Data Analysis*, vol. 122, 2009.

[13] R. Amar, J. Eagan, and J. Stasko, "Low-level components of analytic activity in information visualization," in *INFOVIS*, 2005, pp. 111–117.

[14] R. Chang, C. Ziemkiewicz, T. M. Green, and W. Ribarsky, "Defining insight for visual analytics," *IEEE CG&A*, vol. 29, no. 2, pp. 14–17, 2009.

[15] L. Dey, H. S. Bharadwaja, G. Meera, and G. Shroff, "Email analytics for activity management and insight discovery," in *WI-IAT*, vol. 1, 2013, pp. 557–564.

[16] T. Gao, J. R. Hullman, E. Adar, B. Hecht, and N. Diakopoulos, "Newsviews: An automated pipeline for creating custom geovisualizations for news," in *CHI*, 2014, p. 3005–3014.

[17] R. Ding, S. Han, Y. Xu, H. Zhang, and D. Zhang, "Quickinsights: Quick and automatic discovery of insights from multi-dimensional data," in *SIGMOD*, 2019.

[18] R. Agrawal, R. Srikant *et al.*, "Fast algorithms for mining association rules," in *VLDB*, vol. 1215, 1994, pp. 487–499.

[19] E. Wu and S. Madden, "Scorpion: Explaining away outliers in aggregate queries," 2013.

[20] A. Srinivasan, S. M. Drucker, A. Endert, and J. Stasko, "Augmenting visualizations with interactive data facts to facilitate interpretation and communication," *TVCG*, vol. 25, no. 1, pp. 672–681, 2019.

[21] Z. Cui, S. K. Badam, A. Yalçin, and N. Elmqvist, "Datasite: Proactive visual data exploration with computation of insight-based recommendations," *CoRR*, vol. abs/1802.08621, 2018. [Online]. Available: http://arxiv.org/abs/1802.08621

[22] E. D. Petajan, Y. D. Jean, D. Lieuwen, and V. Anupam, "Dataspace: An automated visualization system for large databases," in *Visual Data Exploration and Analysis IV*, vol. 3017, 1997, pp. 89–98.

[23] G. Wills and L. Wilkinson, "Autovis: automatic visualization," *Information Visualization*, vol. 9, no. 1, pp. 47–69, 2010.

[24] T. Siddiqui, A. Kim, J. Lee, K. Karahalios, and A. Parameswaran, "Effortless data exploration with zenvisage: an expressive and interactive visual analytics system," *arXiv preprint arXiv:1604.03583*, 2016.

[25] S. F. Roth, M. C. Chuah, S. Kerpedjiev, J. A. Kolojejchick, and P. Lucas, "Toward an information visualization workspace: combining multiple means of expression," *Human–Computer Interaction*, vol. 12, no. 1-2, pp. 131–185, 1997.

[26] J. Mackinlay, P. Hanrahan, and C. Stolte, "Show me: Automatic presentation for visual analysis," *TVCG*, vol. 13, no. 6, pp. 1137–1144, 2007.

[27] D. B. Perry, B. Howe, A. M. Key, and C. Aragon, "Vizdeck: Streamlining exploratory visual analytics of scientific data," 2013.

[28] J. Seo and B. Shneiderman, "A rank-by-feature framework for interactive exploration of multidimensional data," *Information visualization*, vol. 4, no. 2, pp. 96–113, 2005.

[29] X. Qian, R. A. Rossi, F. Du, S. Kim, E. Koh, S. Malik, T. Y. Lee, and J. Chan, "Ml-based visualization recommendation: Learning to recommend visualizations from data," *arXiv:2009.12316*, 2020.

[30] K. Hu, M. A. Bakker, S. Li, T. Kraska, and C. Hidalgo, "Vizml: A machine learning approach to visualization recommendation," in *CHI*, 2019, p. 1–12.

[31] J.-F. Kassel and M. Rohs, "Online learning of visualization preferences through dueling bandits for enhancing visualization recommendations." in *EuroVis*, 2019.

[32] G. Leban, B. Zupan, G. Vidmar, and I. Bratko, "Vizrank: Data visualization guided by machine learning," *DMKD*, vol. 13, no. 2, pp. 119–136, 2006.

[33] Q. Liu, M. Vorvoreanu, K. P. C. Madhavan, and A. F. McKenna, "Designing discovery experience for big data interaction: A case of web-based knowledge mining and interactive visualization platform," in *Design, User Experience, and Usability. Web, Mobile, and Product Design*, A. Marcus, Ed., 2013, pp. 543–552.

[34] X. Qian, R. A. Rossi, F. Du, S. Kim, E. Koh, S. Malik, T. Y. Lee, and N. K. Ahmed, "Personalized visualization recommendation," *arXiv:2102.06343*, 2021.

[35] Ç. Demiralp, P. J. Haas, S. Parthasarathy, and T. Pedapati, "Foresight: Recommending visual insights," *arXiv:1707.03877*, 2017.

[36] S. Few and P. Edge, "Data visualization effectiveness profile," 2017.

[37] M. Vartak, S. Huang, T. Siddiqui, S. Madden, and A. Parameswaran, "Towards visualization recommendation systems," *SIGMOD*, vol. 45, no. 4, p. 34–39, May 2017.

[38] F. Viégas, M. Wattenberg, D. Smilkov, J. Wexler, and D. Gundrum, "Generating charts from data in a data table," Mar. 29 2018, uS Patent App. 15/719,425.

[39] S. M. Casner, "Task-analytic approach to the automated design of graphic presentations," *ACM Trans. Graph.*, vol. 10, no. 2, p. 111–151, Apr. 1991.

[40] M. Vartak, S. Rahman, S. Madden, A. Parameswaran, and N. Polyzotis, "Seedb: Efficient data-driven visualization recommendations to support visual analytics," in *VLDB*, vol. 8, no. 13, 2015, p. 2182.

[41] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *TVCG*, vol. 22, no. 1, pp. 649–658, 2015.

[42] F. Bouali, A. Guettala, and G. Venturini, "Vizassist: an interactive user assistant for visual data mining," *The Visual Computer*, vol. 32, no. 11, pp. 1447–1463, 2016.

[43] A. Deardorff, "Tableau (version. 9.1)," *JMLA*, vol. 104, no. 2, pp. 182–183, 2016.

[44] V. Dibia and Ç. Demiralp, "Data2vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks," *CG&A*, vol. 39, no. 5.

[45] Y. Luo, X. Qin, N. Tang, and G. Li, "Deepeye: towards automatic data visualization," in *ICDE*, 2018, pp. 101–112.

[46] Y. Wang, Z. Sun, H. Zhang, W. Cui, K. Xu, X. Ma, and D. Zhang, "Datashot: Automatic generation of fact sheets from tabular data," *TVCG*, vol. 26, no. 1.

[47] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, and J. Heer, "Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco," *TVCG*, vol. 25, no. 1, pp. 438–448, 2018.

[48] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *TEVC*, vol. 1, no. 1, pp. 67–82, 1997.

[49] D. H. Wolpert, W. G. Macready *et al.*, "No free lunch theorems for search," SFI-TR-95-02-010, Tech. Rep., 1995.

[50] Y.-C. Ho and D. L. Pepyne, "Simple explanation of the no-free-lunch theorem and its implications," *Journal of opt. theory and appl.*, vol. 115, no. 3, pp. 549–570, 2002.

[51] T. Joyce and J. M. Herrmann, "A review of no free lunch theorems, and their implications for metaheuristic optimisation," in *Nature-inspired algorithms and applied optimization*, 2018, pp. 27–51.

[52] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.