EVALUATION OF VISUAL ANALYTICS WITH APPLICATION TO SOCIAL SPAMBOT LABELING

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Mosab A. Khayat

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL

STATEMENT OF DISSERTATION APPROVAL

Dr. Arif Ghafoor, Chair

School of Electrical and Computer Engineering

Dr. Alexander J. Quinn

School of Electrical and Computer Engineering

Dr. David S. Ebert

School of Electrical and Computer Engineering

Dr. Walid G. Aref

School of Computer Science

Approved by:

Dr. Dimitrios Peroulis

Head of the School Graduate Program

To my parents Abdulaziz Khayat and Zaheda Alama.

ACKNOWLEDGMENTS

First, I would like to thank Allah, the Almighty, for His guidance and uncountable blessings. Then, I would like to thank my father, Dr. Abdulaziz Khayat, and my mother, Zaheda Alama, for allowing me to start my education journey abroad and for their prayers, the source of my success. I also thank my Aunt, Afaf Khayat, for her warm prayers and sacrifices she made to keep me focused on my study. I want to thank my wife, Hadeel Alaama, and my two daughters, Lamees and Yousra, for enduring and accomodating my busy lifestyle during my study. My sincere thanks to my advisor Professor Arif Ghafoor for his great support and the valuable time he generously devoted to developing my research ideas and skills that led to this work. Many thanks to my respected advising committee members, Professor David Ebert, for his invaluable collaboration and contributions, Professor Alex Quinn and Professor Walid Aref, for their advice and recommendations. I would like to thank all of my professors at Purdue University for developing my background with their courses and projects. I thank all of my respected co-authors who helped in evolving my research ideas with their constructive feedback, especially Professor Morteza Karimzadeh, and I thank my friends and lab mates for all the discussions and brainstorming. Finally, I would like to sincerely thank the Umm Al-Qura University (UQU) and the Government of Saudi Arabia for graciously sponsoring my scholarship.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABBREVIATIONS

abbr	abbreviation
Ch	Chapter
G	Modeled Knowledge
HCI	Human Computer Interaction
JA	Judgment Analysis
JAVA	Judgment analysis evaluation framework for Visual Analytics
KGM	Knowledge Generation Model
r_a	Achievement (accuracy)
TAS	Total Analytical Support
VA	Visual Analytics
VDAR	Visual Data Analysis and Reasoning
VAST	Visual Analytics Science and Technology
VASSL	Visual Analytics Toolkit for Social Spambot Labeling

ABSTRACT

Khayat, Mosab A. PhD, Purdue University, August 2020. Evaluation of Visual Analytics With Application to Social Spambot Labeling. Major Professor: Arif Ghafoor.

Visual analytics (VA) solutions emerged in the past decade and tackled many problems in a variety of domains. The power of combining the abilities of human and machine creates fertile ground for new solutions to grow. However, the rise of these hybrid solutions complicates the process of evaluation. Unlike automated solutions, VA solutions behavior depends on the user who operates them. This creates a dimension of variability in measured performance. The existence of a human, on the other hand, allows researchers to borrow evaluation methods from domains, such as sociology. The challenge in these methods, however, lies in gathering and analyzing qualitative data to build valid evidence of usefulness.

This thesis tackles the challenge of evaluating the usefulness of VA solutions. We survey existing evaluation methods that have been used to assess VA solutions. We then analyze these methods in terms of validity and generalizability of their findings, as well as the feasibility of using them. Subsequently, we propose an evaluation framework which suggests evaluating VA solutions based on judgment analysis theory. The analysis provided by our framework is capable of quantitatively assessing the performance of a solution while providing a reason for the captured performance.

We have conducted multiple case studies in social spambot labeling domain to apply our theoretical discussion. We have developed a VA solution that tackles social spambot labeling problem, then use this solution to apply existing evaluation methods and showcase some of their limitations. Furthermore, we have used our solution to show the benefit yielded by our proposed evaluation framework.

1. INTRODUCTION

Visual analytics (VA) is an emerging problem-solving approach that combines the power of humans and machines to gain a deeper understanding of complex problems to reach optimal solutions that are not achievable by humans or machines working independently. This intertwined environment is created by designing interactive tools that employ automatic data analysis and visualization techniques to improve the awareness of expert users and support their decisions in their domains. The visual analytics approach has been utilized in many areas to solve concrete domain problems including, healthcare [1], finance [2], weather forecasting [3], cybersecurity [4] and many others. Visual analytics solutions have also been tackling abstract problems that serve multiple different fields such as anomaly detection [5], event analysis [6], and data clustering problems [7].

Researchers in the visual analytics community acknowledge the importance and complexity of evaluating VA solutions. To clarify the difficulty of evaluation in VA, it is important to understand what VA is and what makes it special. We start this dissertation by providing necessary background information about visual analytics in Chapter 2. The chapter also provides a survey of the work related to the problems tackled by this dissertation.

The first problem we tackle in this dissertation is to determine which evaluation method should a researcher use to evaluate a VA solution. Because of the diverse evaluation contexts in the field of visual analytics, many researchers start to confuse the applicability of different evaluation methods in different evaluation contexts. This calls for a systematic methodology to analyze current evaluation methods to provide an abstract prescription, which assists researchers in choosing the right evaluation method and improve the validity of evaluation studies in general.

Multiple challenges face researchers who desire to propose a prescription of evaluation methods. Identifying the superiority of one method over another requires an accurate abstract description of the process of evaluation in these methods. This description needs to show the effect of different evaluation activities on the validity of the findings of the evaluation. Furthermore, the description must permit comparing the process of evaluation across all the methods used in the VA field. Developing this description of evaluation methods is challenging because of the diversity of evaluation goals, which are typically mixed up with each other. Moreover, missing metrics capable of quantifying the validity of study findings create yet another challenge for prescribing evaluation methods. The final challenge to tackle when prescribing evaluation methods is to make the prescription flexible to adapt to different evaluation instances in VA.

Considering the summative evaluation context, in which a researcher is trying to judge and prove the usefulness of a developed solution, we find eight different categories of evaluation methods that have been used with summative intention despite the suitability of methods only for formative or exploratory assessment and not summative. To explain differences among summative evaluation methods in terms of their quality of proving usefulness, we analyze the validity and the generalizability (external validity) of the evidence of usefulness the eight categories generate. Our analysis utilizes a taxonomy, which we build to represent the activities employed in each category of evaluation. By focusing on the activities of the methods, we highlight *risk factors* that could affect the validity and generalizability of the generated evidence. We also analyze the feasibility of applying a method, which is also affected by the method's activities. Then, we propose quantitative metrics that measure the summative quality of evaluation methods and the feasibility of using them. We use these metrics to analyze current evaluation methods. From our analysis, we propose a ranking of the eight categories of evaluation in terms of their quality of proving usefulness as well as the feasibility of using them. Chapter 3 present more details on the problem of prescribing summative evaluation method for different evaluation instances and how we tackle this problem.

The second problem we are tackling in this dissertation is to minimize the validity risk in existing evaluation methods. The prescription we propose in Chapter 3 can increase the validity of evaluation studies in the field as it guides practitioners in selecting an evaluation method with the least validity risks. However, our prescription point to some risk factors in every current evaluation methods. These risk factors are inherent limitations in current methods that ought to be overcome to increase the validity of studies findings. A challenge needed to be considered while solving this problem is to maintain feasibility, which reversely relates to validity.

Our solution to the problem of minimizing validity risks in current evaluation methods starts from selecting methods that balance validity and feasibility. We examined the risk factors in these methods and search for a systematic way to minimize validity risks in them while maintaining an acceptable level of feasibility. Two of the evaluation methods that well-balancing validity and feasibility are the performance-based and insight-based frameworks, which are assessing VA solutions in different ways. In the performance-based framework, VA solutions are evaluated using their capabilities to tackle certain decision problems. The insight-based framework, on the other hand, evaluates the support provided by a VA solution to generate new knowledge about analyzed data. The support is assessed using the number of insights reached while interacting with the VA solution.

To reduce the validity risks we find in performance-based, and insight-based frameworks, we propose a new evaluation framework based on judgment analysis (JA) theory [8]. Based on statistics and information theory, our framework improves the validity of the insight-based framework by providing a mechanism and a metric that measures the significance of insights to replace traditional insight-count metric used to justify the usefulness. Furthermore, our JA framework identifies clusters in the population of potential users, which are systematically different, thus need to be studied independently. Following this methodology improves the validity of the traditional performance-based framework, which always treats the user population as a whole. More details about our integration of visual analytics evaluation and judgment analysis theory to tackle the problem of minimizing validity risk are discussed in Chapter 4.

The previous two theoretical problems tackled by this dissertation need to be demonstrated practically. To achieve this, **we choose to apply our contributions in multiple case studies in the domain of social spambots labeling.** A social spambot is a computer algorithm that automatically produces content and interacts with humans on social media, trying to emulate and possibly alter their behavior [9]. These types of bots have been used to propagate harmful content, such as spreading radicalism [10]. Many automated solutions have been designed to detect spambot accounts. However, the nature of the problem requires the continuous tracking of the performance of these models. As in many cybersecurity topics, malicious actors dynamically change and evade existing solutions. This change in the behavior of the attackers creates a challenge to automated solutions that needs to adapt to unknown new behavior. Meanwhile, the scale of the problem creates a challenge for analyzing and labeling social media accounts manually.

To tackle the problem of social spambot labeling, we present VASSL, a visual analytics system that expedites and facilitates the process of spambot labeling. VASSL leverages multiple integrated computational and visual features to support human annotators in inspecting accounts from different angles and at different aggregation levels. Notably, it enables behavior analysis of multiple accounts as groups, instead of analyzing accounts individually, enabling the detection of spammers using multiple accounts, as well as providing users with insights into the collective and dynamic behavior of spambots. VASSL also allows users to conduct analyses at a lower resolution, using views that reveal detailed information about a selected account.

We apply our theoretical contributions of Chapters 3 and 4 to improve the validity of evaluating the visual analytics approach of labeling social spambots. Following the prescription proposed in Chapter 3, we evaluate VASSL using the quantitative user testing method to assess its usefulness objectively. The guidelines in Chapter 3 guide us to apply the quantitative user opinion method as a secondary evaluation to increase the validity of the study findings. The results indicate a statistically significant improvement in the performance of human annotators when they use VASSL.

Our judgment analysis evaluation framework proposed in Chapter 4 is feasible in the social spambot labeling context. We conducted a second case study to apply our framework to show the advantages of using the proposed evaluation framework over traditional evaluation frameworks. In this case study, we evaluate a version of VASSL, which utilizes three of the five views of VASSL. We compare this version to the traditional manual labeling approach when analyzing one social media account at a time. The results highlight differences between groups of tested subjects in terms of performance and analytical support when using the two evaluated approaches. In the context of this case study (i.e., analyzing one twitter account at a time), some subjects favor the VA approach while most of them do not benefit much from the added visualizations and show no improvements compared to the traditional manual labeling approach.

Chapter 5 provides more discussion about the problem of labeling social spambots. It gives detailed information about VASSL and the two empirical experiments we perform to demonstrate our theoretical arguments made in Chapters 3 and 4.

After presenting our contributions to increase the validity of evaluation studies in VA, we conclude the dissertation with a summary of its content. The summary provides a brief descriptions of our solutions to the problem of prescribing evaluation methods (Chapter 3), the problem of minimizing risk validity in evaluation method (Chapter 4), and the application of the theoretical contributions to the problem of labeling social spambots (Chapter 5).

2. RELATED WORK

In this chapter, we present background materials about the field of visual analytics. The information provided in this chapter is essential to understand different aspect of VA systems, which is will be relied on in the chapters to follow. We also present a literature survey for a set of work related to this dissertation.

Thomas and Cook [11] give the most well-known definition of visual analytics as “the science of analytical reasoning facilitated by interactive visual interfaces”. This definition clearly states the role of both human and machine in VA. Human consumes the information provided by the machine and make sense of it using humans’ analytical reasoning capabilities. The machine, on the other hand, communicates a large volume of information to the human by utilizing automated analytical procedures and information visualization techniques.

VA has been seen as a process that transforms raw data into units of knowledge which are commonly called insights. This process has been described by Sacha *et al.* [12] in their Knowledge Generation Model “KGM” (Figure 2.1). The process starts with a set of raw data. VA solutions transform this data into interactive visualizations to enable users to absorb as much information from it as possible. The data is also modeled or summarized using automated procedures such as data mining methods, to generate more pieces of information from the data. The human part includes the set of mental processes that transform observable information into findings, make sense of the findings to reach a set of insights, generate new hypotheses from reached insights or from previous knowledge, search for more findings and evidence through interaction with the VA solution, confirm or reject generated hypothesis and eventually generate new knowledge.

From the description of the VA process, one can observe the nature of VA as a research area that integrates multiple disciplines. This includes information visualization, data management and cleaning, statistics and data analysis (including data mining and machine learning), Human-computer interaction “HCI”, human perception, and cognition science.

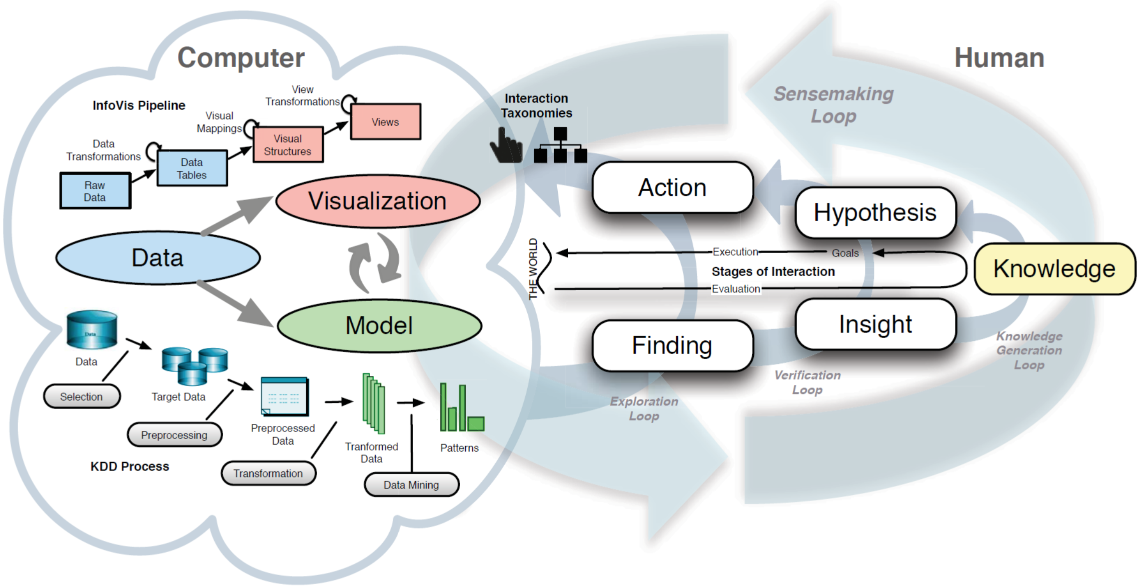


Fig. 2.1.: The Knowledge generation model proposed in [12]. The model describes the process of generating knowledge which is considered the conceptual output of VA

The multidiscipline nature of VA introduces many challenges for researchers to tackle. A set of challenges has been highlighted in multiple studies in the fields [11, 13]. One of these challenges is the challenge of evaluating this complicated system which incorporates automated processes and human-dependent processes.

In this section, we provide a literature survey of the work related to this dissertation. This survey can be broadly be categorized into: *a*) surveys of evaluation practices, *b*) analysis of evaluation methodologies, *c*) prescription of evaluation methods, *d*) judgment analysis and its applications, and *e*) Social Spambot detection solutions.

2.1 Surveys of Evaluation Practices

Multiple studies have surveyed existing evaluation practices. Lam *et al.* [14] suggest that it is reasonable to generate a taxonomy of evaluation studies by defining scenarios of evaluation practices that are common in the literature. Their extensive survey is unique and provides many insights for researchers. Specifically, seven scenarios of evaluation practices are discussed along with the goals of each, with examplar studies and methods used in each scenario. Isenberg *et al.* [15] continue this effort by extending the number of surveyed studies and introducing an eighth scenario of evaluation practices. These studies helped us build the backbone of our taxonomy as explained in Section 3.3.1. The initial code to group evaluation methods in our survey was derived from Lam *et al.* and Isenberg *et al.*. We then gradually modified the coding of evaluation methods according to the studies we surveyed. In contrast to the grouping approach according to common evaluation practices taken by previous surveys, we focus on grouping evaluation methods based on the similarities in each method's (sub)activities, with the ultimate goal of analyzing the potential risks associated with them, rather than simply describing the existing evaluation practices.

2.1.1 Evaluating Knowledge-building in VA

The research and development agenda for visual analytics presented by Thomas and Cook [11] highlights the necessity of developing evaluation methods suitable for assessing VA solutions as knowledge-building tools. VA solutions are developed to facilitate the analytical reasoning of users, a process that generates knowledge from the analyzed data. An evaluation method designed to assess such solutions needs to consider users knowledge as an abstract objective. Many VA researchers formulate the definition of the knowledge resulted from VA process [16–18]. In chapter 4, we select the description chosen by Chen *et al.* [19], which defines knowledge in the perceptual and cognitive space as the ability to use available data and information to answer explanation questions (e.g., *how* to distinguish spammer accounts in social media).

Several contributions have been devoted to tackling the challenge of evaluating the knowledge-building support provided by VA solutions. A common name for this group of evaluation methods is “Evaluating Visual Data Analysis and Reasoning (VDAR)” as suggested by [20, 21]. We found five different types of related work that belong to VDAR.

The first type is the value-driven type, which aims at explaining how to identify the value of VA or visualization solutions. The work in this type acknowledge the limitation of oversimplifying real world tasks in traditional performance-based evaluation, and propose alternative ways to measure the value of VA solutions. An early study that belongs to this type is conducted by Van Wijk [22] who propose an economic model to measure the value of visualization. The model is build based on an iterative process of interacting with visualization and change in user knowledge over time. Stasko [23] proposes another value-driven evaluation of visualization. His work defines the value in terms of four capabilities: minimizing the time to answer diverse questions, spurring the generation of insights, conveying the essence of the data, and generating confidence and knowledge about the data's domain and context. These works contribute valuable conceptual models to understand the importance of knowledge-building to determine the value of VA and visualization solutions. However, these works are not intended to be practical evaluation frameworks as they have not identify how to assess the knowledge-building practically.

The second type of VDAR evaluation methods is qualitative case studies. In this type, evaluators try to identify the amount of knowledge-building support provided by a VA solution by interacting with the users. A well-known framework of this type is the Multi-dimensional In-depth Long-term Case studies (MILC) proposed by Shneiderman and Plaisant [24]. In this framework, evaluators get involved with domain expert users in their work environment for an extended period and try to understand the value of visualization or VA solutions from various collected qualitative data. Another framework of this type is the pair analytics method proposed by Hernandez *et al.* [25]. In this method, the evaluators work in close collaboration with the expert users on an actual analysis problem while trying to understand how the evaluated tool support the analysis. These methods have high ecological validity as they tend to assess a solution in situ. However, the ethnographic, qualitative nature of these methods could limit their precision and generalizability.

Another related work to VDAR is the Knowledge Task-based framework proposed by Amar and Stasko [26]. The authors commence their contribution by defining *analytical gaps*: a group of issues that distance users of a visualization system from understanding a phenomenon explained by the system. After breaking down the analytic gaps into two groups, i.e., the rationale gap and the worldview gap, the authors suggest a set of heuristics to bridge the gaps. Then, they propose the use of these heuristics to evaluate a visualization system, following heuristic evaluation methodology [27]. Analyzing the roadblocks in the knowledge-building process is unique in this framework. However, the usage of heuristic evaluation requires an exhaustive list of finer heuristics to assess knowledge-building accurately. Building such a list is challenging for VA solutions, which are problem-dependent and tackle different analysis problem at different abstraction levels. An effort in building a heuristics list for evaluating VA solutions has been made by Scholtz [28], who emphasize the need for more work in this area to have a reliable list of heuristics.

The fourth type of VDAR evaluation is the work conducted by Chang *et al.* who propose a learning-based evaluation framework. This framework seeks to quantitatively evaluate the capability of VA solutions in supporting the knowledge-building process. The pipeline proposed by the authors suggests post-testing of users after they explore data with a VA solution. Testing is recommended to be performed using some analytical tasks and a semantic questionnaire to measure the amount of knowledge gained from studying the data with a VA solution. The quantitative approach taken by this framework is useful to build generalizable evidence of usefulness for the tested VA solutions. However, a certain level of domain expertise is required to perform the testing with an acceptable level of ecological validity.

The last type of VDAR evaluation work, which is the closest to our work, is the insight-based evaluation framework proposed by Saraiya *et al.* [29]. Insight is defined as "an individual observation about the data by the participant, a unit of discovery" [30], which distinguish the term from the definition of insight in cognitive science [31]. The insight-based framework measures the amount of knowledge-building by asking users to report, in a qualitative manner, any insight they reached. Subsequantly, the reported insights allow the evaluators to define quantitative metrics representing the amount of knowledge gained. The mixed-methodology employed in this framework is unique and well-balancing precision, generalizability, and realism.

There are concerns regarding the validity of the quantitative metrics used in the insight-based framework. Traditionally, insight-based studies rely on **insight count** to measure the amount of support provided by a solution in terms of knowledge-building. In a previous study [32], we explain that the usage of such a metric introduces risk to the validity of the framework since it is likely for insights to be of varying significance. This observation has been reported in multiple studies in the literature, including Sumc [33] who define three levels of insight significance according to the Skill-rule-knowledge loop developed by Rasmussen [34]. Another study conducted by Tan and Chan [35] suggests different insight significance based on Endlsly's levels of situation awareness [36]. Similar to these studies, we argue that it is essential to acknowledge differences in insight significance when using the insight-based framework. Our work, however, suggests an objective quantitative way to measure insights significance according to their contribution to the decision-making process.

2.2 Analysis of Evaluation Methodologies

The next set of related work focuses on explaining and analyzing evaluation methodologies. Evaluation research in visualization and VA can be divided into two types from the perspective of human-involvement: human-dependent evaluation and human-independent evaluation. The methodology of the first type draws on behavioral and social science methodologies to study the effect of visual artifacts on the human operator. One of the most well-known taxonomies for classifying behavioral and social science methodologies that has been ported to the Human-Computer Interaction (HCI) community is proposed by McGrath [37]. This taxonomy was built based on the three main dimensions that any behavioral study seeks to maximize, which are *a*) generalizability, *b*) realism, and *c*) precision. Generalizability of a study determines the extent of applicability of the study findings to any observable cases in general. It is related to the concept of external validity of results. Realism is the representativeness of studied cases to situations that can be observed in the real world; i.e., it determines the level of ecological validity of the findings. Finally, the precision of a study measures the level of reliability and internal validity of the findings. McGrath argues that these dimensions cannot be maximized simultaneously, since increasing one adversely affects the others. He then reviews common methodologies in behavioral science and assigns them to a position in the space defined by the three dimensions. Our analysis of evaluation methods relies on many of the arguments made by McGrath. A key difference between our work and that of McGrath lies in the intention of targeted studies. Our work focuses on studies that have a summative intention of proving usefulness. Unlike the general view of McGrath's work, summative evaluation studies have unique characteristics that permit ranking according to the quality of proving usefulness, as we explain in Section 3.4.

An early study that introduces McGrath's work to the information visualization evaluation context is done by Carpendale [38], who provides a summary of different quantitative, qualitative and mixed methodologies along with a discussion about their limitations and challenges. A more recent work by Crisan and Elliott [39] revisits quantitative, qualitative and mixed methodologies and provides guidance on when and how to correctly apply them. Instead of taking a general view of behavioral methodologies, we use a unified lens to identify limitations in evaluation methods used to prove usefulness, which may follow different methodologies, but are indeed used with summative intentions. Similar to Crisan and Elliott, we use validity and generalizability as our analysis criteria and add the feasibility criterion to the analysis to determine the level of applicability of the methods.

The second type of evaluation in visualization and VA is human-independent. In this type of evaluation, researchers follow a quantitative methodology to assess visualization or VA systems without considering the human element. This includes computer science methods of evaluating automated algorithms [40] and statistical methods for assessing machine learning models (e.g. [41]). A unique quantitative methodology that has been used to evaluate visualization and VA solutions is the information theoretic framework proposed by Chen and Heike [42]. This framework treats the pipeline of generating and consuming visual artifacts as a communication channel that communicates information from raw data, as the sender, to human perception as the receiver. Information theory framework has been used to define objective metrics such as the cost-benefit ratio [43], which has been recently used to build an ontological framework that supports the design and evaluation of VA systems [44]. We include human-independent methods in our analysis because they are summative by nature.

2.3 Prescription of Evaluation Methods

The next set of related work focuses on the prescription of evaluation methods by providing guidelines on what evaluation methods are suitable for different evaluation instances. Andrews [45] proposes four evaluation stages during the development cycle of a system: *a*) before the design, *b*) before the implementation, *c*) during implementation, and *d*) after implementation. Andrews suggests that the purpose, as well as the method of evaluation, is defined by the stage. For example, evaluation studies conducted after the implementation are summative in purpose and usually use methods such as formal experiments or guideline scoring. A more sophisticated prescription of evaluation methods is proposed by Munzner [46], who defines four nested levels, each having a set of unique problems and tasks. During the design stage, developers face multiple problems on their way to the inner level, which requires validation of the design choices. After implementation, a sequence of validation must be performed at each level to validate the implementation on the way out of the nest. Munzner then prescribes different evaluation methods to be used in each validation step. Meyer *et al.* [47] expand this model by focusing on each of the nested levels and proposing the concepts of blocks and guidelines. Blocks describe the outcomes of design studies at each level, and guidelines explain the relationship between blocks at the same level or across adjacent levels in the nest. Another extension to Munzner's work is Mckenna *et al.* [48] who link the nested model to a general design activity framework. The framework breaks down the process of developing a visualization into four activities of understand, ideate, make and deploy.

One argument made by Munzner [46] was the necessity of summative evaluation during each stage of design studies to evaluate the outcome of that individual stage. Sedlmair *et al.* [49] and Mckenna *et al.* [48] made similar arguments while describing the process of design studies. They make the case for considering non-quantitative methods, such as heuristic evaluation, for summative purposes. While the Munzner's nested model [46] essentially prescribes evaluation methods based on the development stage, we focus our analysis and prescription based on the activities performed during evaluation, and judge the quality of evaluation findings (evidence of usefulness) based on the amount of risk introduced by the involved activities. Further, our approach adapts to different evaluation instances and prescribes relatively smaller number of potential evaluation methods, compared to [46].

Another form of prescription studies is the study of correctly adopting existing evaluation methods in the context of VA. Most evaluation methods that have been applied in visualization and VA have been borrowed from the field of human-computer interaction (HCI). Scholtz [50] explains the main factors that need to be added or modified in existing HCI methods to increase their utility in VA research. In addition, she prescribes potential evaluation metrics that have been successfully applied to assessing VA solutions. Still, the necessity of searching for suitable evaluation metrics for visual analytics persists [50–53].

2.4 Judgment Analysis and Its Applications

Judgment analysis (JA) is a theory developed to evaluate individuals' decision policies. The core of the theory was proposed in the mid-20th century by Brunswik [54], who emphasized the importance of understanding variability in individuals' behavior. To generalize findings about a human sample to the population (e.g., weather forecasting accuracy of a group of meteorologists), Brunswik argues that it is essential to initially capture behavior differences among them using multiple instances of the studied decision task. These instances should represent the natural variability in the task (formally called the environment in JA) for accurately studying the behavior of each individual human in that task [55]. For example, testing meteorologists' forecasting skills requires forecasting the weather for multiple days representing the variability of environmental factors such as radiation level, sea surface temperature, and cloud pattern in satellite imagery. Brunswik also points out that it is suitable to use correlational statistics to analyze the variability in human's behavior and environmental factors over the tested instances. This idea was then applied to social judgment studies and eventually given the name judgment analysis [8].

A conceptual model that expresses Brunswik ideas and explains the different components in judgment analysis studies is the **Lens Model** (see the evaluation part of figure 4.1). This model describes the relationships among the three main components in judgment analysis studies: **(1)** human *judgments*, **(2)** an environment *criterion*, and **(3)** a set of *cues* utilized to make the judgments. An example of these three components when studying the skills of a meteorologist is the meteorologist predictions of the days' conditions (judgment), the actual observed conditions of these days (criterion), and the parameters used by the meteorologist to make the predictions (cues). A key concept in judgment analysis studies is to acknowledge the probabilistic nature of the relationship among the three components, and thus, utilize random sampling for all three components to represent the studied cases accurately.

The relationship between human judgment and a criterion represents the performance of that individual human subject in the tackled environment. Collecting n observations of the subject's *judgment* and the environment *criterion* results in two n-dimensional vectors Y_s and Y_e, respectively. In the weather forecasting example, Y_s represents the meteorologist's predictions of n days, whilst Y_e represents the actual conditions in these n days. Measuring the correlation between these two vectors is an example of a commonly used performance metric [56]. Another performance metric commonly used in meteorology is the Skill Score (SS) [57], which is defined based on the distance between Y_s and Y_e instead of the correlation.

In judgment analysis studies, the set of *cues* X is used as a way to capture differences among individuals' decision behaviors. Cues are the factors observed by an individual when making decisions, thus they are used as parameters to model individuals' judgment policies. For example, meteorologist rely on cues such as sea surface temperature and cloud pattern in satellite imagery to forecast the conditions of future days. Unlike studies such as [58], which provide a user with knobs to adjust a decision policy explicitly, JA studies seek to *infer* the decision policy from the relationship between an individual's decision vector Y_s and the set of cues X collected over multiple instances. This relationship can explain the significance of each cue to that individual. Moreover, JA studies rely on the relationship between the *cues* and the environment *criterion* vector Y_e to validate the significance of each cue in the targeted environment.

Many researchers have advanced the judgment analysis field to analyze the performance of individuals [8]. One of these advances is the Lens Model Equation (LME), which decomposes the correlation between Y_s and Y_e to describe the correlation in terms of modeled behaviors [59, 60]. A similar decomposition for the Skill Score metric has been proposed in [57, 61]. These decompositions add tools and measures to analyze the performance of individuals by connecting the performance to captured policies defined in terms of the set of cues X. In Chapter 4, we adopt these measurements and formulas to quantify the significance of insights generated through the VA process.

For a smooth adaptation of JA in VA evaluation, it is essential to illustrate the core difference between JA's representative design and the commonly-used systematic designs [55]. The dispute appears in the way of building a generalizable knowledge about the studied human population. The representative design (of JA) seeks to discover behavior differences among human subjects tested in a realistic environment without controlling environment variables. The rationale of this design stresses the importance of not creating any general conclusion about a group of individuals without ensuring similarity in their behaviors. Representative design claims that each individual has a behavior model that needs to be found to decide if an individual can be grouped with other individuals or not, according to the similarities in their behavior models. This logic differs from the traditional systematic design, which does not consider differences among individuals, i.e., considering all human subjects as a homogeneous group with one behavior model that has some variance. Behavioral models in this context refer to the cognitive processes that generate decisions (i.e., the decision policies), which are identified in a given problem environment. Adapting the representative design to evaluate the analytical support of VA solutions is needed since several researchers have shown systematic behavior differences among VA users [62, 63]. Moreover, generalizing performance to the problem domains rather than the users domain is desired in many VA systems such as those designed to serve particular, relatively small potential users. Such generalization is achievable by the representative design, which primarily focuses on studying the general behavior of individuals in the tested problem environment.

JA theory has been applied to evaluate different factors that concern VA, including the level of situation awareness (SA). According to Scholtz [64], evaluators need to consider the level of SA as an essential metric to assess the usefulness of VA solutions. There are many ways proposed in the literature to evaluate SA level. A unique quantitative framework is proposed in [65, 66], which utilizes JA theory to assess the level of SA. Unlike this framework, which assumes the availability of the set of cues prior to human testing, our work proposes extracting cues during the testing session in the form of findings and insights, as we explain in Section 4.3.

Another application of JA theory to visualization evaluation is proposed by Miller *et al.* [67] who validate the usefulness of visual-aid in judgment problems. Unlike their contribution, which validates the value of visualization in reducing decision bias with the help of JA formulation, our work applies JA theory to understand the value of insights reached with the help of VA solutions, which allows us to define a knowledge-based metric more suitable for VA evaluation.

2.5 Social Spambot Detection Solutions

The problem of detecting spambots in social media applications such as Twitter is an active area of research [68]. Researchers have proposed automatic algorithms to tackle this problem. These studies mainly differ in *a*) the process of generating labeled data which is used in training or testing, *b*) the features that are extracted and engineered to distinguish spambots from genuine accounts, and *c*) the models that are used for the detection process.

Ground-truth, annotated (labeled) data to experiment is essential for designing and evaluating spambot detection methods. For this purpose, researchers use different methods to generate labeled data. As in many machine learning contexts, researchers can manually tag social media accounts by carefully inspecting the information available about them [69]. This method is not scalable, which may force researchers to rely on crowdsourcing solutions to generate the ground truth [70]. However, achieving an acceptable level of reliability through crowdsourcing can be costly [9], especially if the annotation tools are not efficient. Another method is to rely on sites' suspension mechanisms and crawl accounts that have been suspended by the sites' administrators [71]. This method may not be accurate since it is possible for site administrators to ban accounts that are not spambots, e.g. for violating the site's rules. The last common method for building labeled data is by using what is known as a social honey-pot [72, 73]. In this method, researchers create inactive accounts, the honey-pots, which do not interact with social media users or initiate any attractive posts. These characteristics reduce the probability of attracting genuine users and allow researchers to inspect accounts that interact with the honey-pots and possibly tag them as spambots.

Studies that propose automatic detection solutions commonly start with feature extraction and engineering. In this step, researchers define a set of features that allow automated detection models to separate genuine users from spambots. A rich set of features has been suggested in the literature and shown to be useful for the detection tasks. These features can be extracted from accounts' metadata, accounts' social networks, tweets' content, activity in time, sentiment, etc. [70, 71, 74].

After extracting this set of features, researchers propose different models that automate the detection of spambots. Most of these models are developed according to machine learning approaches, which can be categorized into supervised and unsupervised models. Supervised solutions utilize a set of labeled data to learn a discriminant function defined in the feature space. This function defines a boundary between spambots and genuine accounts, which is used afterwards to classify accounts according to their place in the defined feature space [75, 76]. On the other hand, unsupervised solutions build boundaries in the feature space according to similarities among accounts without utilizing predefined account labels [77, 78]. These boundaries define clusters of accounts that are considered similar, thus helping separate spambots from genuine accounts. Unsupervised models need additional human efforts to accurately label members of clusters. This requires an understanding of the characteristics of clusters produced by exploring their members, which may not be intuitive. Our visual analytics approach differs from these solutions, as it incorporates a human into the process of generating the initial clusters.

The most relevant work to ours was conducted by Cao et. al. [79], who proposed TargetVue, a visual analytics solution for detecting and analyzing anomalous user behavior in social media. The computation module in TargetVue utilizes the Time-adaptive Local

Outlier Factor “TLOF” model to rank accounts according to their abnormal behavior. These accounts are visualized using multiple interactive views that allow the exploration of accounts and facilitate manual labeling. One shortcoming of this work appears in the process of identifying the new type of spambot, which propagates malicious content using multiple accounts serving a single actor, rather than the older individual spambot accounts. In such cases, anomaly detection models working on the account-level, such as TLOF, are less effective in identifying spambots, as they could potentially create a *rare category* [80]. Our work emphasizes visualizing algorithmically derived information that helps in grouping accounts, enabling users to search for visual patterns that highlight malicious clusters rather than individual malicious accounts.

3. ANALYSIS OF SUMMATIVE EVALUATION METHODS IN VA

Many evaluation methods have been used to assess the usefulness of Visual Analytics (VA) solutions. These methods stem from a variety of origins with different assumptions and goals, which cause confusion about their proofing capabilities. Moreover, the lack of discussion about the evaluation processes may limit our potential to develop new evaluation methods specialized for VA. In this chapter, we present an analysis of evaluation methods that have been used to summatively evaluate VA solutions. We provide a survey and taxonomy of the evaluation methods that have appeared in the VAST literature in the past two years. We then analyze these methods in terms of validity and generalizability of their findings, as well as the feasibility of using them. We propose a new metric called summative quality to compare evaluation methods according to their ability to prove usefulness, and make recommendations for selecting evaluation methods based on their summative quality in the VA domain.

3.1 Introduction

Visual analytics (VA) solutions emerged in the past decade and tackled many problems in a variety of domains. The power of combining the abilities of human and machine creates fertile ground for new solutions to grow. However, the rise of these hybrid solutions complicates the process of evaluation. Unlike automated algorithmic solutions, the behavior of visual analytics solutions depends on the user who operates them. This creates a new dimension of variability in the performance of the solutions that needs to be accounted for in evaluation. The existence of a human in the loop; however, allows researchers to borrow evaluation methods from other domains, such as sociology [81], to extract information with the help of the user. Such evaluation methods allow developers to assess their solutions even when a formal summative evaluation is not feasible. The challenge in these methods, however, lies in gathering and analyzing qualitative data to build valid evidence.

Many methods have been used to evaluate VA solutions, each originally developed to answer different questions with different evaluation intentions, including formative, summative and exploratory [45]. Nevertheless, many of these methods have been extensively applied in summative evaluation despite the fact that some are only suitable for formative or exploratory assessment, not summative evaluation.

In this chapter, we survey and analyze the evaluation methods commonly used with summative intentions in VA research. Specifically, we survey the papers presented at VAST 2017 and 2018, resulting in a seven-category taxonomy of evaluation methods. We identify the activities typically performed within each category, focusing on the activities that could introduce risks to the validity and the generalizability of the methods' findings, and we use both of these factors to define summative quality. We also define feasibility based on the identified activities and the limitations in applying evaluation methods in various scenarios. Finally, we use summative quality and feasibility to compare summative evaluation methods. Unlike existing problem-driven prescriptions [46], we analyze the risks to the validity, generalizability, and feasibility of each evaluation method by focusing on the activities employed in each method. We then provide a prescription of evaluation methods based on (a) their ability to prove usefulness and (b) their feasibility.

The contributions of this chapter can be summarized as follows:

- A survey, taxonomy and risk-based breakdown and analysis of evaluation methods used in summative evaluation of VA solutions.
- A summative quality metric to assess the summative quality of evaluation methods based on the potential risks to the validity and the generalizability of the methods' findings.
- An analysis and prescription of summative evaluation methods in terms of their summative quality and feasibility.

This chapter is organized as follows: In Section 3.2, we provide important definitions. In Section 3.3, we present our taxonomy of evaluation methods used for summative assessment. We analyze these methods in Section 3.4, followed by a set of recommendations for practitioners in Section 3.5. Finally, we conclude the chapter in Section 3.6.

3.2 Usefulness and Summative Evaluation Definitions

The term "summative assessment" has roots in the field of education, which distinguishes it from formative assessment [82]. The former assesses students objectively at the end of a study period using standardized exams, while the latter focuses on the learning process and the students' progress in meeting standards. In visualization and VA literature, summative evaluation has been traditionally referred to as the type of studies that measures the quality of a developed system using methods such as formal lab experiments [46]. This is in contrast with formative assessment, which seeks to inform the design and development processes by applying techniques such as expert feedback [83].

There have been some suggestions in the literature that evaluation intention should be unlinked from evaluation methods. Ellis and Dix [84] argue that a formal lab experiment of a completely developed system can be conducted with formative intentions to suggest improvement. On the other hand, Munzner [46] argues that formative methods such as expert feedback can be used with summative intentions to validate the outcome of different design stages. We agree with these arguments and believe that it is essential to give a formal definition of summative evaluation as an intention rather than an evaluation stage.

From the discussions in [82], we define summative evaluation as a systematic process which generates evidence about the degree of accomplishment of the given objectives (standards) for an assessed object (a solution) at a point in time. We use the term"solution" throughout this dissertation to refer to different approaches for tackling a problem. VA research covers different types of solutions ranging from algorithms, to visualizations, to the integration of these in a holistic system (See [85] for design study contributions). "Standards" refer to benchmarks that are used to distinguish useful solutions from non-useful ones. These are commonly determined during the requirement elicitation stage, e.g. by conducting qualitative inquiries with domain experts.

Summative evaluation is used to determine the usefulness (i.e. the value) of a solution. From a technology point of view, usefulness is based on two main factors: effectiveness and efficiency [86]. The former can be defined as the ability of a solution to accomplish the desired goals (i.e. doing the right things). The latter concerns the ability of a solution to optimize resources, such as time or cost, while performing its tasks (i.e. doing things right). Most existing summative evaluations assess one or both of these two factors.

Effectiveness and efficiency could be assessed differently according to the nature of the evaluated solution and the problem it tackles. Some solutions can be assessed in a straightforward manner because of the availability of explicit objectives they seek to achieve. An example of such solutions is a classification algorithm, which can be evaluated by objective metrics such as *accuracy*. On the other hand, some solutions require extra effort to define valid objectives that can be used to assess their usefulness. Such effort can be seen in previous work targeted at finding valid objectives to determine the value of holistic visualization and visual analytics systems [86–88].

Usefulness of human-in-the-loop solutions can also be assessed by utility and usability objectives. A Useful system has the needed functionalities (utility) designed in a manner that allows users to use them correctly (usability) [89]. The question of whether to prioritize utility or usability has been discussed in previous work [90]. We focus on the objectives used in utility and usability evaluation and view them with a broader lens as ways to assess effectiveness and efficiency.

We consider effectiveness and efficiency as generic objectives of summative evaluation. This permits us to put all the methods used to assess these two factors in the same plate and compare them in terms of the quality of their evidence and the feasibility of generating them.

3.3 Survey of Summative Evaluation Methods

In this section, we present our survey and taxonomy of methods used by other researchers for the summative evaluation of VA solutions. Our goal in developing a taxonomy is to identify their limitations in terms of their validity, generalizability, and feasibility. Because of our objective of analyzing evaluation methods themselves, it is important to note that we abstract the evaluated solutions and the problems they solve. For example, we do not distinguish between a study that reports a holistic evaluation of a complete VA system and another study that evaluate a part of the system, as long as they both use the same evaluation method. This abstraction is discussed in Section 3.4.

We focused our survey on papers that were published in VAST-17 and VAST-18. The initial number of papers we considered was 97 papers (52 papers from VAST17 and 45 papers from VAST18). We excluded papers that only included usage scenarios or did not report any evaluation at all. Usage scenarios are excluded since they only exemplify the utilization of solutions rather than systematically examining their usefulness. They differ from case studies and inspection methods, which have been used to systematically determining the usefulness of a solution as we explain next. The final number of papers we include in our taxonomy is 82. Some of these papers report more than one type of assessment. The total number of evaluation studies we found in these 82 papers is 182. The number of included papers are relatively small compared to existing surveys [14, 15]. However, our deductive approach to identify evaluation categories requires a smaller sample size compared to inductive approaches which develop concepts by grounding them to data. We built on previous taxonomies [14, 15] to layout ours, and then surveyed recent papers to guide the grouping, activity breakdown and risk analysis.

3.3.1 Survey Methodology

We followed a deductive approach to build our taxonomy, starting with an initial code based on the previous surveys [14, 15], and then progressively changing the concepts in the code by considering new dimensions that help highlight factors that affect validity, generalizability, and feasibility of evaluation methods.

Phase 1: Building the initial concepts We based our taxonomy on two extensive surveys of evaluation practices in visualization and VA literature [14, 15]. The descriptive concepts developed in these works (i.e. the evaluation scenarios) are built for different objectives than our diagnostics. However, these works include the set of evaluation methods used in each scenario, which allowed us to determine our initial code. We consider each reported evaluation method as a concept in this phase and categorize the studies accordingly.

Phase 2: Selecting grouping dimensions We looked for new dimensions that are key for diagnosing evaluation methods' validity, generalizability, and feasibility. By examining the process of evaluation in each method, we identified four dimensions that are useful in grouping the evaluation methods to simplify our analysis: epistemology, methodology, human-dependency, and subjectivity. These dimensions can be seen as titles for each level of our taxonomy depicted in Figure 3.1 and are explained in more detail in the following sections.

Phase 3: Redefining concepts We iteratively refined the taxonomy, which resulted in merging some concepts and splitting others. For example, one of our initial codes was "Quantitative-objective assessment" which included both "Quantitative User Testing" and "Quantitative Automation Testing" in our final code. The dimension responsible for splitting these two concepts is the "Human-dependency" dimension. On the other hand, we decided to merge "quantitative-subjective assessment test" and "quantitative-subjective comparison test" concepts into the single concept "Quantitative User Opinion", because both concepts are similar in every grouping dimension that we considered.

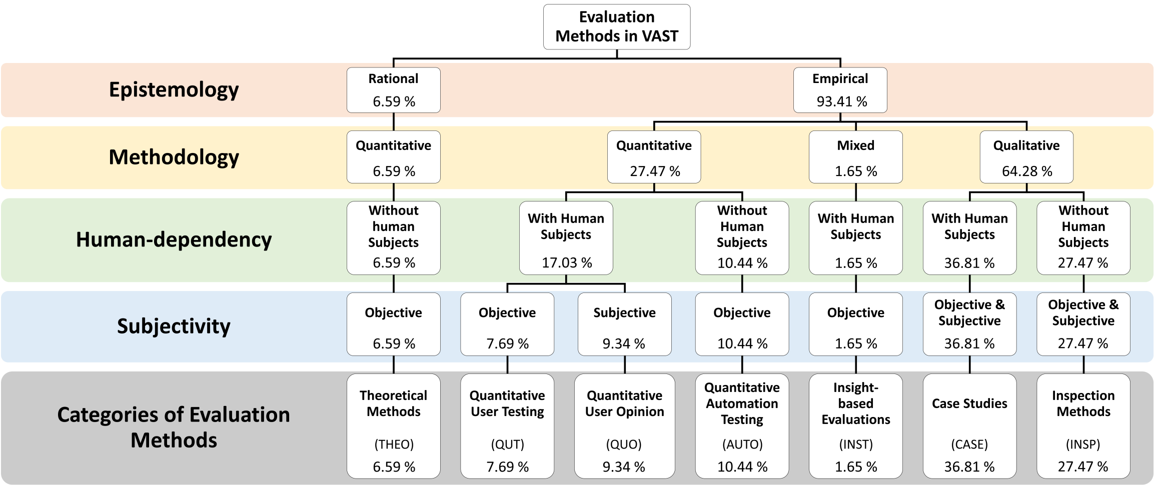

Evaluation Methods in VAST
Epistemology
Rational 6.59 %
Empirical 93.41 %
Methodology
Quantitative 6.59 %
Quantitative 27.47 %
Mixed 1.65 %
Qualitative 64.28 %
Human-dependency
Without human Subjects 6.59 %
With Human Subjects 17.03 %
Without Human Subjects 10.44 %
With Human Subjects 1.65 %
With Human Subjects 36.81 %
Without Human Subjects 27.47 %
Subjectivity
Objective 6.59 %
Objective 7.69 %
Subjective 9.34 %
Objective 10.44 %
Objective 1.65 %
Objective & Subjective 36.81 %
Objective & Subjective 27.47 %
Categories of Evaluation Methods
Theoretical Methods (THEO) 6.59 %
Quantitative User Testing (QUT) 7.69 %
Quantitative User Opinion (QUO) 9.34 %
Quantitative Automation Testing (AUTO) 10.44 %
Insight-based Evaluations (INST) 1.65 %
Case Studies (CASE) 36.81 %
Inspection Methods (INSP) 27.47 %

Fig. 3.1.: A taxonomy of summative evaluation methods based on surveying 82 papers published in VAST-17 and 18. The leaves represent categories of evaluation methods distinguished by the dimensions shown in the left. The percentages show the distribution of surveyed studies.

3.3.2 Taxonomy

Figure 3.1 summarizes our taxonomy. The dimensions are independent and can be used separately to classify evaluation methods. Therefore, the order of dimensions in Figure 3.1 is not important. However, we chose to present a breakdown leading to our identified seven categories. In this section, we explain the dimensions that differentiate the seven categories of evaluation methods and the distribution of the surveyed papers in each level. The following section focuses on the analysis of the processes and activities in each method category.

Epistemology Dimension

Evaluation methods produce evidence that justifies our beliefs about the value of the evaluated solution. The process of justification in the evaluation methods can be categorized, according to epistemological views, into two classes: rational and empirical. Rational evaluation methods use deductive reasoning by relying on logically true premises. For

Table 3.1.: A summary of our survey of the evaluation studies reported in VAST-2017 and 2018. The table provides a brief description of our seven categories and the distribution of the surveyed studies within those categories. The total number of categorized studies is **182** reported in 82 papers.

Abb	Category	Description	Frequency	%	Examples
THEO	Theoretical Methods	Rational, objective, quantitative methods which do not rely on human subjects to generate evidence of usefulness. These methods rely on deductive reasoning to logically derive evidence.	12	6.59%	[91–93]
QUT	Quantitative User Testing	Empirical methods that are objective, quantitative and estimate the performance of human subjects for assessment or comparison reasons.	14	7.69%	[4,94,95]
QUO	Quantitative User Opinion	Similar to the previous category; except, it assesses subjective aspects instead of measuring objective performance. A conventional method in this category is structured questionnaires which use measurable scales, e.g. Likert scale [96], to evaluate user satisfaction and opinion.	17	9.34%	[1,97,98]
AUTO	Quantitative Automation Testing	Empirical methods used to quantitatively and objectively assess human-independent solutions such as machine learning models. This includes evaluation methods such as cross-validation and hold-out test set to predict the performance of supervised machine learning models [41].	19	10.44%	[99–101]
INST	Insight-based	A mixed method which relied on human subjects to qualitatively identify a set of insights that can be reached with the help of a solution. Insight-based methods map identified insights to measurable metrics, e.g. insights count, which are used for quantitative reasoning [88].	3	1.65%	[2, 102, 103]
CASE	Case Studies	Qualitative methods which allow researchers to determine objective values and subjective opinions about the evaluated solution by interacting with human subjects who are typically domain experts. This category encompasses different variants of case studies including Pair analytics [104] and Multidimensional In-depth Long-term Case studies "MILC" [105].	67	36.81%	[106–108]
INSP	Inspection Methods	Methods which assess objective or subjective potentials of a solution without testing or recruiting human subjects. Inspection methods help in checking the satisfaction of predefined requirements that characterize objective or subjective features needed in useful solutions [50, 83, 109–111].	50	27.47%	[112–114]

example, the analysis of algorithms complexity as reported in [80, 91] is rational. This method is used to evaluate the efficiency of an algorithm by determining the time required to execute its instructions. Another rational method of evaluation is the information-theoretic framework [42] that is used in [93] to study the cost-benefit of visualization in a virtual environment. Both of these methods are built on top of a set of basic premises that are assumed to hold, such as the assumption of unit execution time per the algorithm's instruction and the axioms of probability, respectively.

Empirical evaluation methods, on the other hand, follow inductive reasoning by collecting and using practical evidence to justify the value of the solution. Most categories of evaluation methods are empirical. An example of an empirical method is the estimation of automated models' performance as reported in [99, 100]. Such estimations are performed empirically by measuring the performance of a solution in a number of test cases.

Our survey shows that 12 out of 182 evaluation studies (6.59%) were conducted using rational methods. Only one (0.58%) of these 12 studies uses the information-theoretic framework. The other 11 studies (6.08%) applied the traditional analysis of algorithms. Empirical evaluation methods are reported as the method of evaluation in the remaining 170 studies (93.41%).

Methodology Dimension

Evaluation methods are categorized, according to the methodology they follow, into three classes: quantitative, qualitative and mixed methods [38, 39]. Quantitative methods rely on measurable variables to interpret the evaluated criteria. They collect data in the form of quantities and analyze it using statistical procedures to generalize their findings. The evidence generated by these methods has high precision but a narrow scope, i.e. rejection of a hypothesis by measuring particular metrics. Thus, these methods are preferable for problems that are well-abstracted to a set of measurable objectives. Controlled experiments are examples of quantitative methods, used extensively in comparative evaluations such as the studies reported in [115] and [94]. These studies aim to justify the value of a solution by comparing it to counterpart solutions.

Qualitative methods, on the other hand, have fewer restrictions on the type of data that can be collected from a study. They evaluate the usefulness of solutions which tackle less abstract, concrete problems using data that is less precise but more descriptive, such as narratives, voice/screen recordings, and interaction logs. Such data can be generated as a result of observation, or with the active participation of human subjects such as in interviews and self-reporting techniques. The case studies reported in [106] and [107] are examples of qualitative methods used with a summative intention.

Mixed methodology integrates both quantitative and qualitative methods to produce better comprehensive studies [39]. The most common way of following this methodology is to perform multiple complementary studies that are independent but serve the same summative intention (called a convergence mixed method design [116]). For example, the authors of [117] report a controlled experiment as well as a case study with domain experts used to evaluate ConceptVector, a VA system that guides users in building lexicons for custom concepts. The results of both studies can be compared to support each other in proving the value of ConceptVector. Another way of mixing quantitative and qualitative methods is to connect the two types of data prior to analysis such as in an insight-based evaluation method [88]. This method starts by collecting qualitative data in the form of written or self-reported insights, then transforms this data into quantity, e.g. insight count, for analysis, such as the evaluation reported in [2]. Since our taxonomy categorizes evaluation methods at individual resolution, we only categorize methods which follow embedded and merging designs, e.g. the insight-based method, as mixed methods.

According to our survey, 62 studies (34.07%) out of 182 were conducted using quantitative methods. 117 studies (64.29%) were conducted using qualitative methods, and only 3 studies (1.65%) were conducted using the mixed method. According to this, qualitative methods constitute the majority of evaluations in VAST-17 and 18. 21 out of 82 (25.61%) apply the convergence mixed method design.

Human-dependency Dimension

Visual analytics solutions combine both human and automated processes to tackle problems [13]. Researchers may evaluate different components independently. For example, researchers may evaluate the efficiency of an automated algorithm [118, 119], or inspect the requirements of a user interface [120]. Another option is to assess human-related tasks such as estimating the performance of the users [121] or gathering expert feedback about the value of a VA system holistically [122].

The human-dependency dimension in our taxonomy affects all the factors we aim to analyze (i.e. validity, generalizability, and feasibility); therefore, we include it as a dimension in the taxonomy.

Our survey shows that 81 (44.51%) out of 182 studies summatively evaluated a solution without utilizing any human subjects. Among these studies, 31 studies (17.03%) used quantitative methods and 50 (27.47%) used qualitative methods in the form of inspection. On the other hand, 101 studies out of 182 (55.49%) used methods that rely on human subjects. This includes 31, 67, and 3 studies using quantitative, qualitative, and mixed methods respectively (17.03%, 36.81%, and 1.65% respectively). We remind the reader that the word solution is an abstract concept, which can represent automated algorithms, user interfaces or a complete VA system.

Subjectivity Dimension

The usefulness of a solution can be determined by assessing the objective level of accomplishments. However, the objectives are sometimes defined as abstract ideas that cannot be directly or independently assessed. For example, VA systems have a general objective of generating insights about available data [13]. Such an abstract objective may not always be assessable by defined measures. From another angle, a correlation between subjective assessment such as user satisfaction in information systems and the usefulness of these systems has been shown [123]. Therefore, researchers include subjective assessment methods as ways of determining a solution's usefulness. Subjective assessment can be performed quantitatively [1, 97] or qualitatively [124], and can be done with the help of human subjects [125] or through inspecting the design without relying on human subjects [126]. Qualitative methods have the flexibility to assess both objective and subjective aspects.

There is a clear difference between summative evaluation methods that use objective versus subjective scopes. Objective methods assess effectiveness and efficiency of a solution in tackling the targeted problem, whereas subjective methods assess factors that correlate with that solution's capabilities (indirect assessment of usefulness). This led us to include the subjectivity dimension in our taxonomy, to highlight the differences between objective and subjective categories in terms of validity, generalizability, and feasibility.

Our survey shows that 48 (26.37%) studies out of 182 applied objective evaluation methods. 17 studies (9.34%) applied subjective evaluation and 117 studies (64.29%) applied qualitative methods that are not restricted to a narrow scope and can assess both objective and subjective aspects.

The Seven Categories of Summative Evaluation Methods

Table 3.1 summarizes the surveyed evaluation studies in our seven categories of summative evaluation. The most reported evaluation category in VAST-17 and 18 is case studies, followed by the inspection category. These two types are used significantly more than other evaluation categories. The high feasibility of case studies and inspections could be the reason for their popularity, as we explain in Section 3.4.2. On the other hand, the least utilized evaluation category is the insight-based methods. Many of the reported studies that capture subjects' insights do not perform the second stage of defining quantitative measures from captured insights, and thus, end up in the case studies category in our taxonomy.

3.4 An Analysis of Summative Evaluation Methods

We analyze the identified seven evaluation categories in terms of validity, generalizability and feasibility, in order to compare their capability of proving usefulness, which is the objective of summative evaluation. Some of these methods are originally designed to address different evaluation requirements, such as formative or exploratory questions. However, we include them here, since they have been used by others to prove usefulness. Our focus is to analyze the process of evaluation itself regardless of the type of solutions they evaluate.

Table 3.2.: The source of validity, generalizability and feasibility risks encountered when conducting summative evaluation studies.

Activity	Relevant categories	Description of the Risk
Defining the objectives and the objective metric(s)	**THEO**, **QUT**, **AUTO**	Some tasks do not have a clear objective, e.g. exploratory tasks (feasibility risk).
Abstracting the evaluated solution by a formal language	**THEO**, **AUTO**	Some solutions cannot be automated with our current knowledge, e.g. human-dependent solutions. (feasibility risk)
Deductively inferring the performance of the evaluated solution using a formal system	**THEO**	Building a new formal system requires extraordinary work and high abstraction skills. Reusing a formal system requires skills of mapping abstract problems and performing mathematical deduction. (feasibility risk)
Sampling problem instance(s)	**QUT**, **QUO**, **AUTO**, **INST**, **CASE**	Relying on unrepresentative problem instances. (validity, generalizability risk)
Sampling human subject(s)	**QUT**, **QUO**, **INST**, **CASE**	Relying on unrepresentative target users. (validity, generalizability risk)
Sampling competing solution(s)	**QUT**, **QUO**, **AUTO**, **INST**, **CASE**	Bias in selecting competing solutions included in a comparative evaluation study.(validity, generalizability risk)
Identifying the ground-truth	**QUT**, **AUTO**	Unavailable ground-truth for a representative number of problem instances.(feasibility risk)
Organizing studied treatments	**QUT**, **QUO**, **INST**	Fail to eliminate confounders. (validity, generalizability risk)
Statistical testing	**QUT**, **QUO**, **AUTO**, **INST**	A potential reduction to the risk as a result of testing the statistical significance of quantitative analysis findings. (validity, generalizability risk reduction)
Qualitatively identifying insights	**INST**	Subjects potential miss-reporting of reached insights / researcher potential miss-collecting of reached insights. (validity, generalizability risk)
Defining quantity from insights	**INST**	Defining a metric that do not reflect the value of solutions. (validity risk)
Collecting and interpreting qualitative data	**CASE**	Missing essential pieces of information / misinterpreting the value of a solution evaluated using collected information. (validity, generalizability risk)
Identifying the requirements / heuristics sources	**INSP**	Relying on a source which provides less than needed requirements/heuristics to distinguish a useful solution from another. (validity, generalizability risk)
Requirements / heuristics elicitation	**INSP**	Mis-eliciting requirements / heuristics from the identified source. (validity, generalizability risk)
Judging the satisfaction of the requirements / heuristics	**INSP**	Inspector subjectivity in checking the accomplishment of requirements / heuristics. (validity, generalizability risk)
Indirect inference of usefulness	**QUO**, **CASE**, **INSP**	Inferring the value of a solution from measures or findings that do not directly test the solution objectively. (validity, generalizability risk)

3.4.1 Analysis Criteria

Validity and generalizability are well-known properties of generated evidence in scientific studies and have been broken down into many types. The primary types influencing our analysis are internal validity and external validity as defined in experimental quantitative studies [127], as well as credibility and transferability as defined in qualitative studies literature [128]. We view validity as the property of correctness of study findings, while we see generalizability as the extent to which study findings can be applied to similar but unstudied (unevaluated) cases.

By examining the findings of each evaluation method, we found four types of summative evidence which assess effectiveness or efficiency:

a) quantities that represent the objective performance (measured or estimated by a method from **THEO**, **QUT**, **AUTO**, or **INST**),

b) quantities that represent subjective satisfaction (estimated by a method from the **QUO** category),

c) qualitative information about objective or subjective value of a solution (gathered by **CASE** methods),

d) accomplishment of requirements/heuristics (inspected by a method belonging to **INSP** category).

Each evaluation method includes a set of activities resulting in one of the aforementioned four types of evidence. In our analysis, we outline the activities for each method and highlight risk factors associated with each activity. We rely on the definition of risk found in the software engineering literature [129], which defines exposure to risk as the probability-weighted impact of an event on a project (evaluation in our case). The identified risk factors may affect the validity and generalizability of the outcome of each method. For example, the generalizability of empirical evidence is affected by the sampling of cases for the study. Thus, in our analysis, we designate sampling as an activity for empirical evaluation methods and associate it with potential generalizability risk. On the contrary, some activities may reduce risks to validity or generalizability. For example, a typical activity to maintain the validity of quantitative empirical evidence is to apply inferential statistical tests [130]. Such testing activity is an example of what we call a risk reducer.

Besides validity and generalizability, feasibility is the third criterion we consider in our analysis. We include this criterion to reason about researchers' decisions to evaluate solutions using methods with less summative quality. Table 3.2 describes the potential validity, generalizability and feasibility risks we identify for each of the summative evaluation category, along with the source of these risks.

3.4.2 Evaluation Process Breakdown

We break down the (sub)activities common to the methods in each category of our taxonomy. Then, we highlight the risks introduced or reduced as a result of performing these activities. The process of identifying the activities and highlighting their associated risks was performed based on our personal experience, validated and by the survey we report in Section 3.3. Figure 3.2 presents a summary of our analysis, along with risks highlighted on each activity.

Theoretical Methods (THEO)

This category includes complexity Analysis of Algorithms & information-theoretic framework. These rational methods start by defining an objective metric, e.g time complexity, which is a useful measurement for assessment or comparison tasks. To measure the metric, researchers are required to abstract the behavior of the solution using a formal language, e.g. a programming language (Figure 3.2). This explicitly means full knowledge about the behavior of the solution. The last activity is to build a formal system, e.g. Turing machine [131], and use the premises in that system, e.g. unit execution time per instruction, to deductively measure the defined objective metric. Most rational studies captured in our survey apply the analysis of algorithm method to measure the time complexity of algorithms that are abstract by nature, and thus do not require the second activity. Moreover, the Turing machine is an applicable formal system that can be used to perform the deduction in this context. Another set of rational studies, which are more sophisticated, rely on information theory premises [132]. Most remarkably, these works present an abstraction activity for solutions that are not abstract by nature [44, 93].

The three activities we report for rational methods do not introduce any risk to the validity and generalizability criteria. They are rigorous activities that always measure what

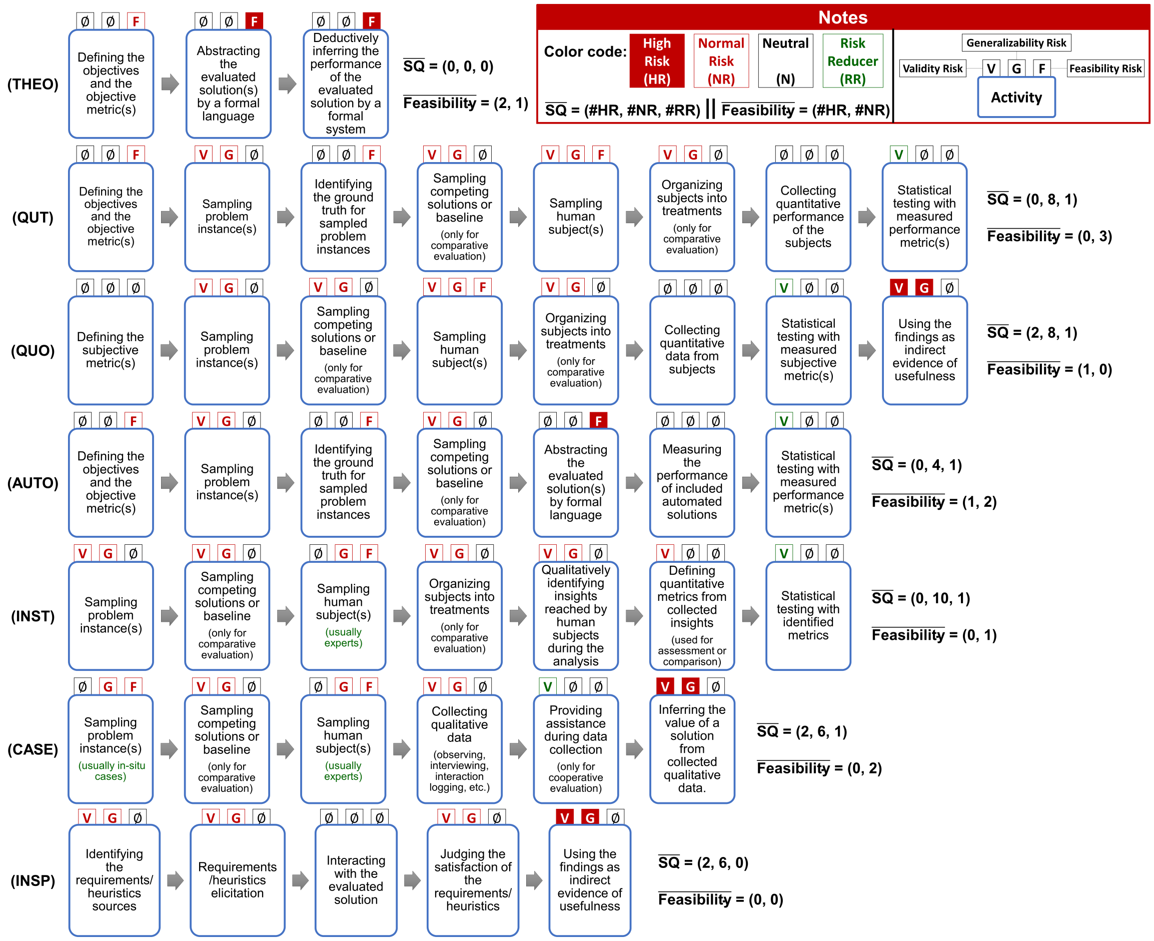

Notes
Color code:
High Risk (HR)
Normal Risk (NR)
Neutral (N)
Risk Reducer (RR)
SQ = (#HR, #NR, #RR) || Feasibility = (#HR, #NR)
Generalizability Risk
Validity Risk
V G F
Feasibility Risk
Activity
(THEO)
Defining the objectives and the objective metric(s)
Abstracting the evaluated solution(s) by a formal language
Deductively inferring the performance of the evaluated solution by a formal system
SQ = (0, 0, 0)
Feasibility = (2, 1)
(QUT)
Defining the objectives and the objective metric(s)
Sampling problem instance(s)
Identifying the ground truth for sampled problem instances
Sampling competing solutions or baseline (only for comparative evaluation)
Sampling human subject(s)
Organizing subjects into treatments (only for comparative evaluation)
Collecting quantitative performance of the subjects
Statistical testing with measured performance metric(s)
SQ = (0, 8, 1)
Feasibility = (0, 3)
(QUO)
Defining the subjective metric(s)
Sampling problem instance(s)
Sampling competing solutions or baseline (only for comparative evaluation)
Sampling human subject(s)
Organizing subjects into treatments (only for comparative evaluation)
Collecting quantitative data from subjects
Statistical testing with measured subjective metric(s)
Using the findings as indirect evidence of usefulness
SQ = (2, 8, 1)
Feasibility = (1, 0)
(AUTO)
Defining the objectives and the objective metric(s)
Sampling problem instance(s)
Identifying the ground truth for sampled problem instances
Sampling competing solutions or baseline (only for comparative evaluation)
Abstracting the evaluated solution(s) by formal language
Measuring the performance of included automated solutions
Statistical testing with measured performance metric(s)
SQ = (0, 4, 1)
Feasibility = (1, 2)
(INST)
Sampling problem instance(s)
Sampling competing solutions or baseline (only for comparative evaluation)
Sampling human subject(s) (usually experts)
Organizing subjects into treatments (only for comparative evaluation)
Qualitatively identifying insights reached by human subjects during the analysis
Defining quantitative metrics from collected insights (used for assessment or comparison)
Statistical testing with identified metrics
SQ = (0, 10, 1)
Feasibility = (0, 1)
(CASE)
Sampling problem instance(s) (usually in-situ cases)
Sampling competing solutions or baseline (only for comparative evaluation)
Sampling human subject(s) (usually experts)
Collecting qualitative data (observing, interviewing, interaction logging, etc.)
Providing assistance during data collection (only for cooperative evaluation)
Inferring the value of a solution from collected qualitative data.
SQ = (2, 6, 1)
Feasibility = (0, 2)
(INSP)
Identifying the requirements/ heuristics sources
Requirements /heuristics elicitation
Interacting with the evaluated solution
Judging the satisfaction of the requirements/ heuristics
Using the findings as indirect evidence of usefulness
SQ = (2, 6, 0)
Feasibility = (0, 0)

Fig. 3.2.: A summary of our analysis of evaluation methods. We capture the main activities taken by evaluation methods which could introduce risk to evidence validity, generalizability and feasibility. We assign 3 risk categories for these criteria per activity, classify each risk factor to high, normal or reducer class, then compare the methods using their summative quality (SQ) and feasibility.

they claim to measure. Rational methods also evaluate abstract problems and solutions with well-defined behavior, and thus are completely generalizable to any untested cases. For example, finding the worst case time complexity for an algorithm as $O(n)$ means no observable case of input size n will ever take longer than linear execution time.

The issue of rational methods appears in the feasibility criterion. The first feasibility risk is introduced by the first activity, which defines an objective metric. In many problems, the objective metric might not be feasibly defined. For example, the general goal of VA systems is to generate insights about data, a goal that may not be easily assessed by measurable factors. The second activity introduces much more sever risk to the feasibility. Abstracting the evaluated solution's behavior using a formal language requires sufficient knowledge about that solution's behavior, which may not be possible for some types of solutions. For example, it is challenging to develop a formal language representation of human analytical processes, which practically limits the applicability of this type of evaluation on human-in-the-loop solutions. Since a human in these solutions controls their behavior, and that we cannot replace a human with a completely automated machine, it is not feasible to describe the human user's behavior using a formal language. If the behavior of the solution cannot be abstracted, the third activity becomes infeasible since it cannot be performed in a formal manner without an abstract, well-defined solution. Moreover, building a formal system to deductively infer the performance of a solution is challenging and requires high abstracting skill.

Quantitative User Testing (QUT)

In these empirical quantitative methods, researchers study human-in-the-loop solutions by either conducting a formal comparative experiment or measuring the performance of the solutions independently. The latter can be considered a special case of the former. These methods start by defining objective metrics, similar to rational methods. However, a typical activity in all empirical methods is to sample test cases. These cases are determined by sampling problem instances and human subjects. In comparative evaluation studies, the sampling of test cases includes the sampling of competing solutions. To objectively estimate the performance of the solution, researchers need to define ground truth for tested problem instances, which can be either sampled or synthesized [133]. After sampling the test cases, researchers organize human subjects into groups (treatments) according to the study design. Two common designs include the within-subject (repeated measures) and between-subject (independent measures) designs. After organizing the study according to the selected design, researchers test human subjects with the sampled problems and collect quantitative measures of performance for each subject. These performance measurements can subsequently be analyzed per treatment using statistical tests (e.g. Analysis Of VAriance "ANOVA"). For assessment studies, statistics provide a confidence interval of the measured performance score for the solution. For comparative evaluation, the statistical tests ensure the significance of the difference between the performance of treatments. Some accuse such typical hypothesis testing methodology [134]. Nevertheless, Null-hypothesis significance testing (NHST) remains the most recognized methodology in quantitative scientific work.

The activities in the QUT category introduce risk to every criterion we analyze. A risk to the validity and generalizability criteria can be introduced as a result of sampling bias that excludes cases included in the study claim, sampling an insufficient number of cases to prove the claim, or failing to eliminate confounders when organizing treatments. The second risk can be reduced by applying a statistical test to show the potential of observing the findings for represented cases in general. The third risk is not a concern for assessment methods that do not generate evidence of usefulness as a result of comparing treatments.

The activities of QUT introduce risk to the feasibility criterion as well. Sampling representative cases can be infeasible because of the unavailability of representative human subjects or representative problem instances with known ground truth. Moreover, as in rational methods, it may not always be possible to identify a clear, objective metric that correctly distinguishes useful solutions from non-useful ones.

Quantitative User Opinion (QUO)

The activities in this category are quite similar to the previous category. However, the focus here is on assessing subjective aspects instead of the objective performance, and there is no need to establish ground truth for the test cases.

The difference between subjective and objective methods, in terms of risk can be illustrated as follows. The risk to the validity is higher in subjective methods, since besides potential sampling and assignment biases, subjective methods do not assess usefulness directly. As we have mentioned earlier, the evaluation of usefulness by definition is a way to assess solutions objectively. Subjective methods approach achieve this by assessing factors that are assumed to correlate with usefulness, such as user satisfaction. However, such correlation may not always be valid. According to Nelson [89], a system with limited utility could have high usability but would not be useful because of the missing functionalities. However, subjective methods are more feasible than objective methods. They do not require knowledge about ground truth nor quantifying objectives, and thus can be applied in more cases.

Quantitative Automation Testing (AUTO)

These methods apply the same activities as **THEO** methods. The only difference between the two categories is the method of measuring the objective metrics for abstract solutions. In **THEO**, extensive work is devoted to building the formal system used in deduction, which is challenging because it requires high abstraction skills and sufficient knowledge about the problem domain. An alternative approach, taken by methods in the **AUTO** category, is to prove usefulness empirically by relying on sampled cases and statistics. For example, most methods used to evaluate machine learning models rely on estimating the performance with a set of testing problem instances [41].

The risk to the validity and generalizability of the evidence generated by a method from the **AUTO** category is slightly less than the risk associated with the **QUT** category. The reason is the reduction in sampling bias in **AUTO** methods as the result of excluding the human dimension. On the other hand, the exclusion of the human dimension explicitly means less feasibility of **AUTO** methods, since they are only capable of evaluating abstract solutions described by a formal language.

Insight-based Evaluations (INST)

As an empirical category, sampling activities are typical in **INST**. A unique activity in this category is the qualitative data collection of insights. This is done by asking human subjects to self-report any insights they reach during the analysis by applying techniques such as diary [135] or think-aloud protocols [136]. Another unique activity in this category is the creation of measurable quantities out of collected qualitative data. The typical quantity to generate is insights count, which gives an indication of the usefulness of analytical support solutions.

Besides sampling bias, which can introduce risk to both validity and generalizability, **INST**'s unique activities may increase the risk to these criteria. For example, collecting insights as qualitative data introduces the possibility of misreporting some insights or misunderstanding reported ones. However, **INST** has a low feasibility risk since it does not require defining any objective metrics nor developing any tasks that ought to be evaluated quantitatively. **INST** also does not require prior knowledge about the ground truth of sampled problem instances.

Case Studies (CASE)

Instead of measuring the accomplishment of solutions with some predefined metric (which may not be feasible or known for concrete domain problems), **CASE** methods study realistic cases defined by actual real-world problem instances and intended users who are usually experts. To extract evidence of usefulness, evaluators pay extra attention to any data that can be captured during the examination. Collecting qualitative data is essential in case studies for creating a rich source of information, which helps in determining the usefulness of evaluated solutions. Many techniques can be implemented to generate qualitative data, including observation, semi-structured interviews, subject feedback, Think-aloud protocol, video/audio recordings, interaction logs, eye tracking and screen capturing [38]. During data collection, researchers may assist human subjects to overcome learnability issues. From the collected qualitative data, researchers can infer the value of evaluated solutions from the human subjects' perspective. This hypothesis of evaluated solutions' value can be used as evidence of usefulness, given that the human subjects are experts in the problem domain.

The risk to the validity and generalizability criteria for **CASE** methods can be explained as follows. Beside possible sampling bias, qualitative methods evaluate usefulness indirectly.

The risk resulting from this indirectness can stem from two issues. The first is the potential misunderstanding of the human subjects when hypothesizing the value of the evaluated solution, which is typically known as the credibility of study findings. The second risk is the credibility of the subjects themselves, whose opinions are considered evidence of usefulness. This validity is affected primarily by how knowledgeable the subjects are about the problem domain, and secondarily by how much they know about using the evaluated solution. Another possible source of risk to the validity of case studies comes from the evaluators. The data collection and analysis in case studies can be profoundly affected by evaluators' subjectivity. Inexperienced evaluators may miss relevant information during data collection or wrongly infer the value of the solution from collected data. The risk introduced by the evaluators can be minimized by experience and by following guidelines that reduce subjectivity. There are tremendous existing literature on the correct application of qualitative studies [137, 138].

The advantage of case studies lies in their feasibility. They do not require specifying and measuring objective metrics or abstracting the solution. They also do not require knowledge about ground truth for the problem instances included in the test cases, because their objective assessment is derived from expert opinion, who are assumed to be capable of assessing the usefulness while testing the solution. The only feasibility risk to this category is the availability of expert human subjects, and the sampling of representative realistic problem instances.

Inspection Methods (INSP)

The first activity of **INSP** is to identify the factors needed in useful systems through methods such as conducting a qualitative inquiry with stakeholders to identify requirements [113] or surveying the literature to identify known heuristics [83]. Once a set of requirements/heuristics is identified, researchers start inspecting the evaluated solution and judge whether it satisfies the identified requirements/heuristics.

INSP includes the most feasible methods, not requiring human subjects nor testing with any problem instances. However, these methods prove usefulness marginally and with many validity and generalizability concerns. The risk to the validity and generalizability of the findings of **INSP** include (a) the credibility of the information source, (b) the exhaustiveness of the elicited requirements/heuristics, and (c) the subjectivity of the inspectors. Inspection methods have been shown to have significantly less potential for identifying usability issues compared to formal testing [139]. This finding inherently means high risks to both the validity and generalizability of **INSP**'s evidence of usefulness.

3.4.3 A Ranking of the Summative Evaluation Categories

After identifying risk factors to the validity and generalizability, we combine both criteria into a single metric which we call **summative quality** (SQ). The term is inspired by applied medical research for categorizing and ranking the quality of research evidence [140]. We define SQ as the probability of *not* falling in any of the potential validity and generalizability risks introduced by a set of activities, i.e. the probability of an evidence to be valid and generalizable. Similarly, we consider feasibility as the probability of *not* falling in any of the risk factors that threaten feasibility.

SQ can be calculated by Equation 3.1. We assume that the risks introduced by different activities are independent. Thus, to measure the total quality from subsequent activities, we take the product of the complement of the probability of risk in each activity. Taking the product is typical in similar total probability calculations (e.g. [141]). It is worth mentioning that the granularity of describing evaluation methods should not affect the total risk calculation. A single activity in a coarse-grained description of a method should accumulate all risk probabilities of that method when described in a fine-grained manner.

$$SQ = \prod_{i=1}^{n} (1 - P(risk_i)) \tag{3.1}$$

Equation 3.1 measures the product of the probabilities of not falling in any of the n validity and generalizability risks. This model of risk assessment requires estimating the

Table 3.3.: The ranking of the seven categories of summative evaluation methods based on the potential risk to their validity, generalizability, and feasibility. We rank the categories according to their $\overline{SQ}$ and $\overline{feasibility}$.

Abb	Category	Summative Quality Rank	Feasibility Rank
THEO	Theoretical Methods	1	6
QUT	Quantitative User Testing	3	4
QUO	Quantitative User Opinion	5	2
AUTO	Quantitative Automation Testing	2	5
INST	Insight-based evaluation	4	2
CASE	Case studies	4	3
INSP	Inspection methods	4	1

probability of the captured risks, which is a challenging task. To overcome this issue and to be able to compare evaluation methods, we categorize the risk factors into three groups: high risk (HR), normal risk (NR) and risk reducers (RR) (Figure 3.2). High risk factors are introduced by any activities that infer usefulness indirectly (i.e., from evidence that do not measure objective metrics). Such activity would produce evidence of usefulness that have more uncertainty due to the high evaluators' potential subjectivity.

Using the categories of risk, we define $\overline{SQ}$ to compare evaluation methods In lieu of SQ. $\overline{SQ}$ can be defined as a triplet $\overline{SQ} = (\#HR, \#NR, \#RR)$, with each dimension representing the number of risk factors in each category. We calculate $\overline{SQ}$ for all categories then use the resultant triplets to observe any clear superiority of one category over another (e.g. (2,6,1) has less $\overline{SQ}$ given the two high risks compaerd to (0,8,1)). Based on this, we rank evaluation methods in terms of their $\overline{SQ}$ (Table 3.3). The table also ranks evaluation methods based on $\overline{feasibility}$, which can be defined as a tuple $(\#HR, \#NR)$ considering the feasibility risk factors. In case a clear superiority can not be decided (e.g. (0,10,1) Vs. (2,6,1)), we assign the same ranking to these methods. We stress that even though some categories rank low for $\overline{SQ}$, they may still be suitable for other purposes such as formative or exploratory.

3.5 Recommendations

Based on our taxonomy and analysis of summative evaluation methods, we provide the following recommendations:

1- Always select a feasible method with the highest summative quality. Prescribing an evaluation method for a given context can be done based on summative quality and feasibility. It is always encouraged to select the method with the highest summative quality. However, the feasibility of applying one of the methods in a given evaluation context may influence the selection. For example, the superiority of rational methods over empirical methods when testing usefulness; however, researchers may use an empirical method to evaluate a human-in-the-loop solution because of the infeasibility of abstracting human behavior using formal language as previously mentioned.

Our approach complements the nested model [46], which prescribes potential evaluation methods for each level. For instance, four different methods were prescribed to validate a solution in the encoding level. Complementing such prescriptions by following our approach can narrow down to a method from the Nested model prescribed methods.

2- Provide reasoning for evaluation method choice. We suggest providing solid reasoning when choosing an evaluation method for a summative evaluation. Our framework may help in this reasoning by considering the summative quality and feasibility as criteria. We note that it is always possible to use a weaker form of proving usefulness when it is feasible to generate stronger evidence with another method. For example, one can rely on subjective methods to assess the usefulness of a solution designed to tackle a problem that can be evaluated objectively. In such scenarios, evaluators should explain the limitation that prevents them from using the method that generates stronger evidence of usefulness.

An example from the literature for a study that could have provided such an explanation is [142]. The authors used the inspection method to evaluate the usefulness of DeepEyes, a VA system developed to enhance designing deep neural networks. DeepEyes could have been evaluated using a formal controlled experiment i.e. by measuring training time and the classification accuracy of the end architecture (when using DeepEyes vs. traditional trial and error). Inspection has less summative quality compared to controlled experiments; thus, choosing the former over the later requires justification.

3- Encouraging insight-based evaluation. A surprising finding from our survey is the limited application of insight-based evaluation to published work in VA. According to our analysis, insight-based evaluation is one of the few methods that do not suffer from high risk factors. It is capable of assessing human analytical processes with realistic problems while generating quantitative outcomes that can be replicated and generalized. According to our survey, researchers favor case studies over insight-based methods in evaluation contexts that are suitable for both. We encourage performing insight quantification and quantitative analysis instead of case studies to increase their precision and generalization potentials.

4- Apply multiple evaluation methods to minimize risk Our final recommendation encourages practitioners to apply multiple evaluation methods to prove the usefulness of their developed solutions. All of the evaluation methods include activities that could potentially invalidate the evidence they generate. An easy remedy is to compare the level of usefulness reached by different methods. This recommendation is strongly encouraged for subjective methods and inspection methods because of their relatively high validity and generalizability risks. Subjective methods are usually utilized to complement objective assessment, which is an excellent strategy for measuring usefulness from different angles.

3.6 Summary

We presented our survey of evaluation practices used with summative intentions in VA. We identified seven categories of evaluation, broke down the activities in each, and analyzed each category in terms of feasibility as well as the validity and generalizability of their findings. We proposed summative quality as the primary metric for selecting evaluation methods for the summative intention of proving usefulness. Based on the summative quality metric and the complementary feasibility metric, we proposed a ranking of the categories of evaluation.

One of the limitations in our analysis is the possible subjectivity in identifying risk factors. We attempted to minimize it by continuously consulting the literature and conducting a survey. Assigning risk factors to only two categoris could also be considered a limitation. However, we favor robustness over precision when analyzing evaluation methods.

Even though we based risk analysis on extensive literature and our survey, our proposed ranking of evaluation methods might be considered subjective. Regardless, we argue that it characterizes the risks involved in selecting methods for summative evaluation, and most importantly, our risk analysis paves the way for future research and community ranking, similar to many repeated fruitful efforts in medical research [140]. Categorizing risks associated with activities and even quantifying such risks based on expert-assigned scores or probabilities is an established practice in system engineering and risk assessment [143], and our work lays the foundation for such analysis of evaluation methods in VA.

By identifying risk factors and providing a methodology, our work also enables community-driven prescription of evaluation methods. According to [144], experts have high potential in judging risk factors and assigning probabilities. This approach can be used to assign probabilities to our identified risk factors using Equation 3.1. To reduce subjectivity in judgment, one can deploy a community-driven voting system to increase the accuracy of estimating the risk probabilities and to build standards to prescribe evaluation methods.

4. EVALUATING VA SOLUTIONS USING JUDGMENT ANALYSIS THEORY

Evaluating visual analytics (VA) solutions is a challenge that has existed since the beginning of VA research. A key challenge that faces VA evaluators is defining objective metrics to measure performance in solving analytics problems, especially in exploratory settings. By connecting the analytics and decision-making processes, we propose a new evaluation framework that utilizes judgment analysis (JA) theory to guide the design of evaluation experiments. We propose a new objective metric by combining JA theory and the insight-based evaluation framework [30] to evaluate the analytical support of VA solutions. Our framework also provides measures to assess the significance of the insights reached through the VA process. We demonstrate the utilization and superiority of our framework in a case study in the context of identifying social spambots in Chapter 5.

4.1 Introduction

The visual analytics approach to tackling data analysis problems continues to attract researchers from many domains. This approach allows making optimal decisions according to the information embedded in available data, combined with the knowledge gained through domain expertise. Including humans in the loop of decision-making complements automated algorithms and improves their effectiveness [145]. However, the existence of human in the VA approach to problem-solving increases the variability in its output, thus complicating their evaluation. Another complicating factor for the evaluation of VA solutions is the serendipitous nature of data exploration. Measuring the support provided by VA solutions to tackle analytical problems is, therefore, not straightforward and requires carefully defined objective metrics.

There are some promising approaches for evaluating the analytical support provided by VA solutions. Analytical support has been seen as the abstract objective of all VA tools as it measures the degree of support provided by the tool to confirm and discover facts in the data by the users [20]. One approach for evaluating analytical support is the insight-based evaluation framework [29], which has been used extensively in the past decade. However, one limitation of this framework appears in the validity of its quantitative objective metrics (insight-count), which does not consider differences in the significance of identified insights.

Another approach for evaluating visual analytics is to evaluate the quality of the decisions made in light of the produced analytical products. This *performance-based* approach has been used for the objective and holistic evaluation of VA systems, which is affected by both the VA tools and the individuals who operate them. To estimate the performance of the VA systems, the existing performance-based approaches test a sample of potential users and infer the performance of the VA systems for the whole population of potential users. However, two issues can arise in such approach. First, the population of potential users is assumed to be a homogeneous one that can be statistically described by analyzing them as a whole, whereas different users may actually belong to finer subgroups that behave uniquely from one another. The second issue appears in ignoring the potential variability in the decision tasks by testing the performance of the system with selected problem instances, which prevents generalizing the performance of the VA systems to the whole problem space which contains other untested instances.

To address these challenges, we build on Judgment Analysis (JA) theory [8] and propose a new evaluation framework capable of quantitatively measuring the analytical support provided by VA tools. The proposed framework evaluates the analytical support of VA tools by identifying insights reached by the users of the tools, then measuring their significance in the targeted decision problems using an information-theoretic metric [146]. An appealing advantage of the proposed framework is its capability for quantitatively validating its findings through statistical methods such as cross-validation [147]. Moreover, the proposed framework resolves the issues in existing performance-based evaluation methods by representing the performance of VA systems in the whole problem space. This is done by analyzing potential users individually rather than considering them as a single homogeneous group. This approach enables discovering clusters of users that need to be evaluated independently.

The main contributions of our work can be summarized as follows:

- An evaluation framework and new metrics based on JA theory to evaluate VA solutions in terms of insight-significance and general objective performance
- A case study demonstrating the successful use of our proposed framework for evaluating a visual analytics system for spambot labeling. The results prove the superiority of our framework in measuring the analytical support provided by the VA system.

We organize this chapter as follows: Section 4.2 defines the scope of our Judgment Analysis for VA (JAVA) framework and the requirements that must be satisfied for applying it. Section 4.3 explains how to apply our JAVA framework to evaluate VA. We conclude the chapter in Section 4.4.

4.2 Requirements for Applying JAVA Framework

VA systems consist of two sub systems: an information system (i.e., the VA tool) and a cognitive system (i.e., the analyst). When seeking to evaluate the usefulness of a VA approach, we either need to evaluate the holistic VA system or to evaluate the information system portion. Our JAVA framework can be used for both types of evaluation when the requirements explained in this section are satisfied.

To understand the applicability of the JAVA framework to evaluate VA systems holistically, it is important to view these systems as problem-solving approaches. The connection between visual analytics and decision making has been highlighted since the commencement of the field [11, 17, 148]). Building on this connection allows viewing visual analytics as a problem-solving approach, which combines human and machine to derive answers. From a theoretical point of view, evaluating any problem-solving approach can be done objectively by assessing the capability of the approach in solving all possible instances of the tackled problem (i.e., the question desired to be answered by the approach). We argue that this is the case for the evaluation of holistic VA systems as well. However, identifying the problem tackled by a VA system can be challenging, as it requires understanding the utility of the analytical products produced through the interaction between the information system and the cognitive system [149]. In some cases, the problems tackled by the VA systems are explicitly defined, which simplify the job of the evaluators. For example, the VA system proposed in [150] explicitly tackle the problem of optimizing neural network model selection in terms of efficiency and effectiveness. At the other extreme, some VA systems (e.g., systems with exploratory visual analytics tools [151]) tackle too many problems (i.e., help answering an undetermined number of questions), which makes evaluating these VA systems objectively impractical. However, it is possible for the evaluators to define a subset of the problems tackled by exploratory VA systems by narrowing the construct of their evaluation studies.

There are three main requirements to enable the use of JAVA framework to evaluate VA systems holistically. First, the framework can only be applied to evaluate VA systems that tackle well-defined problems. JAVA primarily relies on assessing the usefulness of VA systems by measuring their accomplishment in tackling the defined problems (i.e., the accuracy of the answers produced by the evaluated system to the question that defines the problem). Moreover, the framework requires testing individual human subjects with representative instances of the tackled problem to enable modeling the decision policy of these individuals. This modeling of individuals in JAVA is needed to generalize the behavior of the VA system to the whole problem space (i.e., generalize the behavior to the environment that includes all instances of the tackled problem). Finally, the JAVA framework also requires the ground-truth for the problem instances used to model decision policies so that the modeled behavior of the system can be objectively evaluated.

Evaluating the information system part (i.e., the VA tool) rather than the holistic VA system requires defining different objectives. The abstract objective of VA tools is to support the analyst in making sense of the analyzed data [17, 18]. Thus, analytical support can be defined as a metric to evaluate the usefulness of VA tools. A quantitative way of measuring analytical support is to estimate the number of insights that can be reached with the help of the VA tool as proposed in the insight-based framework [30]. JAVA framework follows a similar approach to estimate analytical support. However, instead of treating insights equally, JAVA framework measures the significance of the reached insights by linking them to the decisions made by the VA system as a whole. Insights are formulated in JAVA as *features* that parametrize the decision policies of individual analysts. This formulation allows calculating scores for insights importance using feature importance techniques such as information gain (i.e., the mutual information between the analysis decisions and the insights) as we explain in the next Section. Because of the reliance on the decisions made by the VA systems holistically, the aforementioned three requirements of evaluating VA systems using JAVA are also required when attempting to evaluate the information system part only.

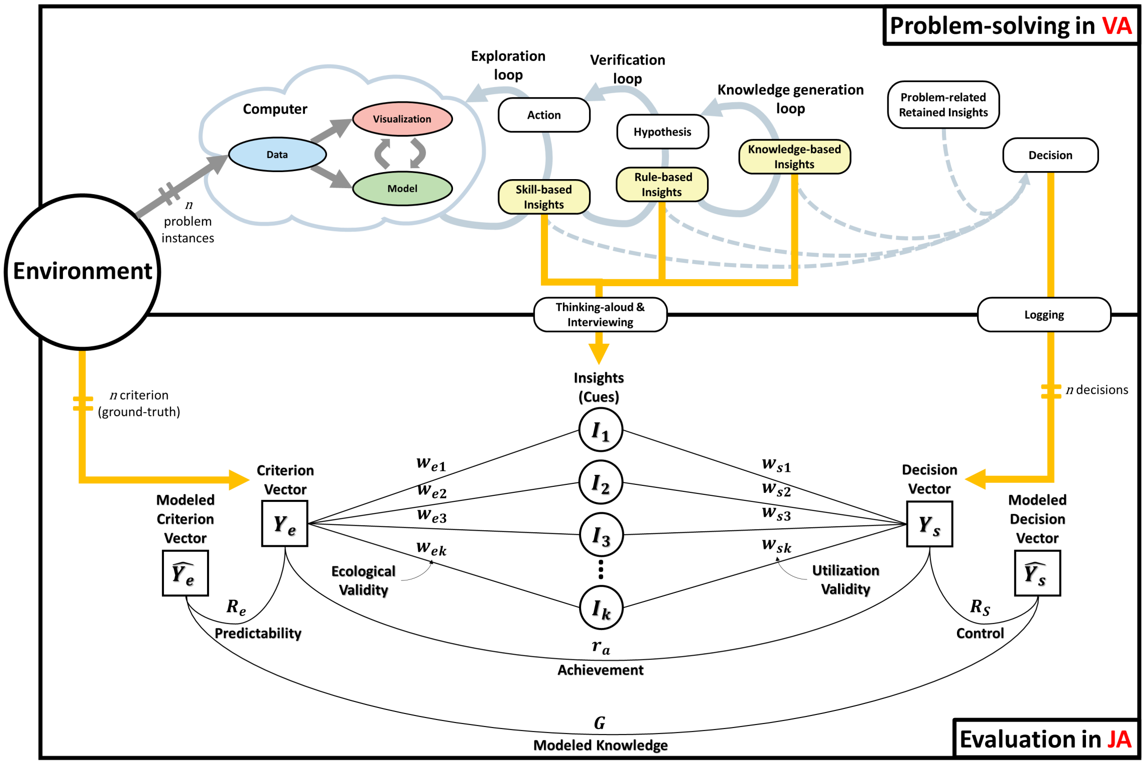


Fig. 4.1.: The relationship between the VA process of problem-solving (Top) and the components of JA theory illustrated through the Lens Model (Bottom). The yellow arrows represent the evaluation data collected by our JAVA framework during the testing phase.

4.3 The Procedure of JAVA Framework

After clarifying the requirements of JAVA framework, we describe the procedure of this framework to evaluate VA based on JA theory. The described mixed-methodology can be used to achieve both performance and analytical support assessment goals (i.e., to evaluate VA systems holistically or to evaluate the VA tools alone). During the discussion, we advise the reader to return to Fig. 4.1 to link JA components to VA components. This figure merges the knowledge generation model (KGM) of VA [18] with the lens model [8], illustrating the relationships between the different components in JA studies. We integrate KGM with Smuc's categorization of insights according to Rasmussen's ladder [33, 34] to reveal the cognitive processing demand during the VA process of problem-solving. This integration is needed to acknowledge the fact that in some problem-solving instances, a user could make decisions based on simple direct findings rather than complex, more cognitively demanding insights or knowledge. Smuc's categorization comes in handy since it considers these cognitive products as types of insights, which permit analyzing them similarly using our JAVA evaluation framework.

The procedure of JAVA consists of seven steps. It starts by **(1)** defining the problem tackled by the evaluated VA system, **(2)** sampling representative instances of it, and **(3)** identifying their ground truth. Then, the framework requires **(4)** collecting decisions about each of the sampled problem instances from each tested individual, along with a set of insights reached during analyzing the sampled problem instances. After completing the data collection phase for all tested individuals, the framework provides mechanism for **(5)** modeling the decision policies of the individuals as well as the ground truth. The models are estimated using the identified set of insights, the captured decisions, and truth for the sampled problem instances. Given these models, the framework **(6)** evaluates VA using our proposed *modeled knowledge* (G) and *Total Analytical Support* (TAS) metrics. Finally, the framework allows **(7)** clustering individuals with similar decision policies and assigns any study conclusion at the cluster level. More details about these seven steps are provided in the remainder of this Section.

1- Defining the Problem The first step in evaluating a VA system is to identify its purpose, which can be articulated as a solution to a problem. As we have explained in the previous Section, problems can be defined and tackled by a VA system explicitly, such as the *problem* of labeling Twitter accounts as genuine or spambot tackled by our previous work [152]. Some VA systems, on the other hand, tackle less precise problems such as exploratory VA systems [151]. Defining the problem tackled by these systems creates an extra task for the evaluators, which is learning about how analysts use the information gathered from the evaluated VA tools. Answering this question can be the goal of qualitative inquiries conducted with domain experts who explore similar data sets for decision making purposes. When evaluating a VA system, it is important to have a deep understanding of the tackled problem to be able to define it with a level of abstraction to identify the potential variations of the problem (i.e., the instances of the tackled problem).

2- Sampling Representative Instances It is usually not feasible to evaluate a VA system using all possible instances of a problem, which leads to the necessity of relying on a representative sample of problem instances. Representativeness is vital for inference, i.e., developing evaluation conclusions that generalize to the whole problem domain. JA literature suggests that the minimum number of required problem instances (n) be proportional to the number of parameters used to model individuals [8].

3- Identifying the Ground Truth In objective evaluation studies, the correct solutions for the tested problem instances (i.e., the ground truth) must be known for evaluators. There are several ways to identify the ground truth of problem instances. For some problems, it is possible to rely on direct observation, such as observing actual events occurring after weather forecasting [153]. Other problems may need identifying ground truth through qualitative inquiries of domain experts. An alternative practice to objectively evaluating a system tackling problems with unknown ground truth is to synthesize a realistic environment. However, this should be performed carefully to ensure an acceptable level of ecological validity. Whiting et al. [154] propose a system that assists in generating synthetic, yet realistic data with known criterion in the context of threat detection such as terrorist activities. Their contribution has been used in multiple VAST challenges [155], which are instances of synthetic, realistic environments with known ground truth. Once the ground truth for n problem instances are identified, evaluators can organize them in a criterion vector Y_e (Fig. 4.1).

4- Collecting Evaluation Data Our JAVA framework requires collecting two kinds of data from each tested human individual. These data are collected over the n sampled instances of the problem in a manner similar to longitudinal studies [156]. The first kind is the decisions for the tested n problem instances. These decisions can be viewed as n samples of a random variable, which can be organized in a n-dimensional decision vector Y_s (Fig. 4.1).

The second kind of data is the insights reached while solving the n instances of the problem. Similar to the decisions, insights are treated in our framework as random variables that are sampled while solving the instances. Evaluators need to complete two intertwined tasks to collect the needed information about insights. First, they need to identify the insights. Then, they need to measure them for different problem instances. Insights can be extracted from a cognitive system through various methods. In the original insight-based framework, evaluators ask human subjects to write down any insights they reach in a diary [29]. This method needs relatively less processing time, less cognitive resources, and less obtrusion, which makes it suitable for in-situ longitudinal studies that span multiple days. Another method is to extract insights from human subjects through self-reporting techniques such as the think-aloud protocol [30, 157]. This method asks the subjects to verbalize their thinking process, and then derives insights from the recorded data. The limitation of this method appears in its potential adverse effect on reasoning and problem-solving processes [158].

While identifying insights, evaluators need to capture variability in their measurements. The representative design of JA allows evaluators to expose different variations of the same insights. This variability in insights value, or at least the variability of their existence, allows measuring the insights with one of Stevens's scales [159]. A qualitative inquiry in the form of semi-structured interviews can be conducted to understand the identified insights in a way that enables measuring them.

By the end of step 4 of our JAVA framework, evaluators should have collected an n-dimensional decision vector Y_s, an n-dimensional criterion vector Y_e, and an $n \times k$ insight matrix I for each tested human individual. Elements of the two vectors represent the individual's decision and the ground truth of the n tested problem instances. Meanwhile, the columns of I represent k different insights identified while solving problem instances, with rows representing n measurements for those insights as observed in the corresponding instances.

5- Modeling Decision Policies and Truth After data collection, evaluators can statistically estimate individuals' decision policy model which leads to their captured decisions. Such modeling aims at understanding the decision behavior of individuals given the instances of the tackled problem. Moreover, the collected data permits evaluators to estimate a model that explains the variation of the problem and the ground truth associated with it. The estimated models provide evaluators with extra information on top of what they directly observe.

Identifying the parameters of the models is essential in JA theory. In traditional JA studies, researchers choose a set of parameters (cues) of interest and attempt to model individuals and truth using them. In the VA context, however, we are interested in studying insights generated during the knowledge-building process, which are not known in advance. By identifying the matrix I, evaluators can use the captured insights as parameters to model decision policies and ground truth. Using this formulation allows us to give a practical definition of insights as "features utilized by individuals' decision policies to solve problems".

The validity and the generalizability of the estimated models can be statistically measured using Cross-validation and Holdout methods [41]. The procedure of using these methods are identical to the common practices in machine learning context. Evaluators should start by splitting the data collected over the n tested instances into training and testing sets. The *training* set is used to estimate the models whilst the *testing* set is used to measure the *generalizability* of them. During the model estimation, cross-validation is used to *validate* the performance of the model. This method divides the training set into folds and use one of them to estimate the performance of the models trained with the rest of the folds. This is performed multiple times until each fold is used for validation. the performance estimate of each run can then be averaged to measure the overall validity of the model. During this validation, evaluators can test different modeling techniques and hyper-parameters' assignments, searching for the best model that fits the collected data. Once that model is validated, it is possible to test its generalizability using the testing set that is held out of the process of model selection and training.

The modeled decision policy of an individual and the modeled criterion can be used to predict the decisions and the ground truth of the tested n instances. The results of these predictions can be organized in the vectors $\hat{Y}_s$ and $\hat{Y}_e$, respectively. These vectors replace the raw vectors during performance analysis to measure the performance using the models instead of observations as we explain next.

6- Calculating Modeled Knowledge (G) & Total Analytical Support (TAS) Modeled knowledge (G in Fig. 4.1) is an objective metric that estimates an individual's performance. It is quantified by measuring the correlation or the distance between the vectors $\hat{Y}_s$ and $\hat{Y}_e$. in other words, modeled knowledge represents the part of an *individual's performance* which is captured by the estimated model of the decision policy of that individual and the estimated model of the truth. Unlike the *achievement* (i.e., the directly observed performance from comparing Y_s and Y_e), modeled knowledge assesses the knowledge of an individual about the solution of all possible variations of the tackled problem. This knowledge is manifested in the decision behavior of that individual which seeks to match the behavior of the truth with respect to the captured insights. We suggest using G instead of the achievement in performance estimation to benefit from the generalizability scores measured for the models used to calculate G (i.e., to associate the estimated performance with a measure of generalizability).

In many cases, it is desirable to evaluate VA tools in terms of the amount of support they provide for knowledge-building. The traditional insight-based method suggests using insight-count as a measure of generated knowledge [30]. As we have mentioned previously, relying on insight-count raises validity concerns because of the probable varying significance of insights. Our JAVA framework allows for measuring the significance of each insight. As we have described, insights in the proposed framework play the role of *parameters* of the estimated models of decisions and truth. Following this perspective maps the problem of measuring insight significance to the realm of *feature relevance* in statistics and machine learning [160]. One promising measure of feature relevance is *information gain* (also known as *mutual information*, equation 4.1), which measures the reduction in Shanon entropy for the distribution of decisions (Y_s or Y_e in our case) when the feature (any insight I_i in our case) is known [161]. We propose to use this measure to determine two scores of significance for each insight; **insight utilization validity** W_s, and **insight ecological validity** W_e (Fig. 4.1). Insight utilization validity is defined as the information gained from an insight when modeling the individual's decision policy, which quantifies the importance of the insight in that policy. On the other hand, ecological validity of an insight is defined as the information gained from including the insight when modeling the truth, which objectively validates the significance of the insight in the tested environment. These two scores of significance can replace the "Domain value" score, which is measured subjectively in traditional insight-based evaluation framework [30].

$$Information\_Gain(Y,I) = entropy(Y) - entropy(Y|I) \tag{4.1}$$

We propose the Total Analytical Support (TAS) metric (equation 4.2) to define a single metric that measures analytical reasoning support provided by a VA tool. This metric is defined as the sum of normalized insights' significance in an environment (i.e., the ecological validity W_e of each insight) minus any error in utilizing the insights. Utilization error is defined as the absolute difference between the ecological validity of the insights and their utilization validity, since both validity scores are identical in the perfect case (i.e., because the decision policy model and the truth model are desired to be identical according to the same insights used to model them).

$$TAS = \sum_{i=1}^{k} norm(W_{ei}) - |norm(W_{ei}) - norm(W_{si})| \tag{4.2}$$

$$norm(W_{ei}) = \frac{entropy(Y_e) - entropy(Y_e|I_i)}{min[entropy(Y_e), entropy(I_i)]} \tag{4.3}$$

$$norm(W_{si}) = \frac{entropy(Y_s) - entropy(Y_s|I_i)}{min[entropy(Y_s), entropy(I_i)]} \tag{4.4}$$

TAS fulfills two requirements. The metric favors tools that help in developing insights with high ecological validity W_es (i.e., high significance in the targeted problem). Moreover, the metric acknowledges the superiority of tools that convince users to utilize insights in a way similar to their ecological validity (i.e., low differences between W_e and corresponding W_s over all insights). For example, a tool that helps in developing one very relevant insight will have a higher TAS score than a tool that helps in producing many insignificant insights. Furthermore, a tool that convinces its users to utilize insights according to their environmental significance will have a higher score than a tool that does not guide the utilization of developed insights.

TAS is upper bound by insights-count, which is the case when a VA tool helps the individual in reaching insights that are significant in relevance to the environment and helps that individual to utilize generated insights according to their significance. The lower bound of TAS, on the other hand, is negative insight-count, which occurs when a VA tool leads an individual to insights that are not significant at all relevant to the environment and convince that individual to utilize these insignificant insights.

Measuring the validity of G and TAS for an individual depends on the validity of the estimated decision policy for that individual and the model of the truth. Step 5 explains how to measure the validity of the models as percentages, which can be represented as scores between 0 and 1. Calculating the validity of G and TAS requires accumulating these scores. This accumulation can be performed by multiplying the marginal validity scores (i.e., the validity scores of the decision model and the truth model calculated in step 5). Generalizability of G and TAS can be calculated in the same manner using the generalizability scores of the decision model and the truth model.

7- Clustering Individuals Up until this step, our framework has guided evaluators to generate idiographic knowledge about individuals, which enables observing behavioral differences among them. The followed representative design does not assume or claim generalizability in the user domain. However, the representative design can be followed by a clustering step to increase the nomothetic knowledge about sampled human individuals. The advantage of our framework appears in the precision of generalizing study findings to specific sections of the user domain (i.e., the potential users population). Instead of viewing the population of users as a whole and search for a single conclusion that fits the whole population, the fine-level collected data of our framework allows for controlling the resolution of the study by identifying subgroups and studying them separately. This leads to better overall generalizability, as it views the potential user population as multiple, and potentially independent populations.

Defining clusters within the m sampled individuals is essential in this paradigm. Many statistical clustering methods [162] can be applied to identify subgroups using the collected data. The input to these methods is a representation of studied human subjects, which reflect their decision policies (e.g., their calculated utilization validity vectors). A generalizable conclusion about insights significance (W_e and W_s of each insight), modeled knowledge (G), and Total Analytical Support (TAS) can then be measured and tested per cluster.

4.4 Summary

In this chapter, we bridge the gap between the field of judgment analysis and the evaluation of VA systems. First, we explain in an abstract way the logic of problem-solving in VA by proposing RPM model, which explains the formulated representations of problems during the process of solving them. Modeling the process of problem solving clarify the difference between the proposed JA evaluation framework and two other objective evaluation frameworks (performance-based and insight-based). Then, we explain how to evaluate VA using JA theory by combining the knowledge generation model [18] and the levels of insights [33], and link these two VA contributions to the JA components represented by the lens model [8]. To demonstrate our JA evaluation framework, we conducted a case study in which we evaluate a VA solution in the context of social spambot labeling. This case study is discussed in more details in the next chapter.

5. APPLICATION TO SOCIAL SPAMBOTS LABELING

Social media platforms are filled with social spambots. Detecting these malicious accounts is essential, yet challenging, as they continually evolve to evade detection techniques. In this chapter, we present VASSL, a visual analytics system that assists in the process of detecting and labeling spambots. Our tool enhances the performance and scalability of manual labeling by providing multiple connected views and utilizing dimensionality reduction, sentiment analysis and topic modeling, enabling insights for the identification of spambots. The system allows users to select and analyze groups of accounts in an interactive manner, which enables the detection of spambots that may not be identified when examined individually.

We use the problem of labeling social spambots as a case study for the proposed theoritical descussions in previous chapters. First, we explain which evaluation method is suitable to assess our VASSL by relying on the analysis proposed in Chapter 3. Our next research would be focused on using VASSL and the problem of labeling social spambots to apply our judgement analysis evaluation framework proposed in Chapter 4.

5.1 Introduction

A social spambot is a computer algorithm that automatically produces content and interacts with humans on social media, trying to emulate and possibly alter their behavior [9]. These types of bots have been used to propagate harmful content such as spreading radicalism [10]. To counter this threat, many automatic solutions have been designed to detect spambot accounts. However, the nature of the problem requires the continuous tracking of the performance of these models. As in many cybersecurity topics, malicious actors dynamically change and evade existing solutions. Researchers report a shift in the behavior of social spambots, which allows them to evade current solutions [163]. Detecting this new generation of spambots individually has become more challenging at the account-level since they tend to propagate spam through campaigns rather than using single accounts [164]. These accounts' behavior may be similar to genuine users in every aspect if monitored at the individual account-level. It has been shown that detecting this type of spambot with accepted accuracy is significantly more challenging for both human annotators and machine learning models [163]; as a result, there is a need for new detection methods that are scalable and capable of tackling the dynamically changing environment. To address this challenge, new solutions have emerged to provide analysis of spambots at a group-level [165,166]. Although it has been shown that these solutions significantly improve detection of the new generation of spambots in some instances, they are still unable to achieve the desired performance in general.

To tackle the problem of detecting this new type of spambot, we present VASSL, a visual analytics system that expedites and facilitates the process of spambot labeling. VASSL leverages multiple integrated computational and visual features to support human annotators in inspecting accounts from different angles and at different aggregation levels. Notably, it enables behaviour analysis of multiple accounts as groups, instead of analyzing accounts individually, enabling the detection of spammers using multiple accounts, as well as providing users with insights into the collective and dynamic behavior of spambots. VASSL also allows users to conduct analyses at a lower resolution, using views that reveal detailed information about a selected account.

VASSL is designed to work with Twitter accounts; however, the general concepts can be used with other social media platforms as well. VASSL provides five integrated interactive views that communicate different information about the accounts to support the process of labeling and is designed for use by analysts or expert users whose goal is to efficiently and effectively annotate spambots and/or gain insight about online spambot behavior.

After presenting the design, algorithms, and various components of VASSL, we provide a use case that demonstrates the utilization of the developed tool to perform a complete analysis and labeling tasks. The benchmark dataset used in the testing was crawled from Twitter and prepared by [163], who published the data for research purposes.

To validate the usefulness of VASSL, we present a formal user study that compares our tool's labeling performance against a manual labeling approach. The results indicate statistically significant improvement in the performance of human annotators when they use VASSL. VASSL improves the average accuracy of labeling by 14.6% while increasing the effectiveness of labeling by 0.84 account per minute.

VASSL is publicly available at https://vassl.new-dimension.co. Besides incorporating existing visualization techniques, we present and integrate two novel interactive visualizations to communicate patterns of groups in time series data and to communicate the distributions of groups in multi-dimensional feature space.

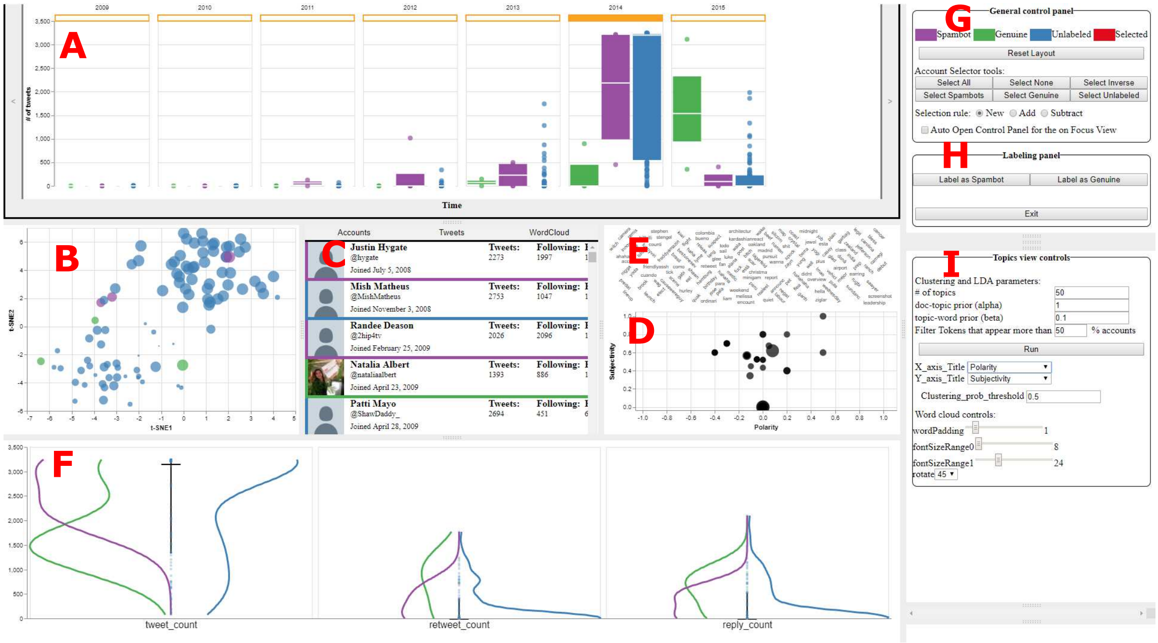
This chapter is organized as follows: In Section 5.2, we present an overview of the design of VASSL and the requirements we seek to satisfy. Section 5.3 describes the functionalities of the back-end and provides more details about the techniques we use to prepare the data for the front-end, which is discussed in Section 5.4. We present a use case of VASSL in Section 5.5. In Section 5.6, we provide two case studies to apply the theoretical proposals of previous chapters and conclude this chapter in Section 5.7 with directions for future work.

5.2 Design Requirements of VASSL

To derive the initial design of VASSL, we consulted the literature on both social spambot labeling and social media visual analytics, with specific attention to the representation of accounts, potential automatic and interactive clustering methods, and visualization of high dimensional and textual data. After reviewing related literature, we derived the following set of system requirements:

R1 **Show similarities among accounts.** This requirement is essential for enabling users to explore different characteristics to cluster the accounts based on [79, 163, 166].

R2 **Represent accounts at different aggregation levels.** Most of the features we found in the spambot detection literature can be considered time-series features. Examining these features at different temporal aggregation levels reveals different patterns which could help identify spambots [79].



Justin Hygate
@hygate
Joined July 5, 2008
Tweets: 2273
Following: 1997

Mish Matheus
@MishMatheus
Joined November 3, 2008
Tweets: 2753
Following: 1047

Randee Deason
@2hip4tv
Joined February 25, 2009
Tweets: 2026
Following: 2096

Natalia Albert
@nataliaalbert
Joined April 23, 2009
Tweets: 1393
Following: 886

Patti Mayo
@ShawDaddy_
Joined April 28, 2009
Tweets: 2694
Following: 451

Selection rule: ● New ○ Add ○ Subtract

□ Auto Open Control Panel for the on Focus View

Clustering and LDA parameters:
# of topics 50
doc-topic prior (alpha) 1
topic-word prior (beta) 0.1
Filter Tokens that appear more than 50 % accounts

X_axis_Title Polarity
Y_axis_Title Subjectivity
Clustering_prob_threshold 0.5

wordPadding 1
fontSizeRange0 8
fontSizeRange1 24
rotate 45

Fig. 5.1.: The default layout of the front-end of VASSL: A) the timeline view, B) the dimensionality reduction view, C) the user/tweet detail views, D) & E) the topics view (clustering / words), F) the feature explorer view, G) the general control panel, H) the labeling panel, and I) the control panels for all the views (the opened control panel in the figure is for topics clustering view).

- **R3 Summarize tweets' content and show details on demand.** According to Shniderman's visual information-seeking mantra [167], it is desirable to visualize the content summary of tweets to the users and show the details of the tweets as interactively requested.

- **R4 Allow the user to highlight and analyze groups of accounts.** The system should enable users to highlight and cluster accounts interactively. This interactive clustering is essential to reveal potential group-based spamming activities [163, 166].

- **R5 Enhance the efficiency and effectiveness of human workers.** The system should enable cost effective annotation and improve the accuracy of detecting spambots as well as reducing the time needed to label groups of accounts by human workers [9].

- **R6 Support different machines and screen sizes.** Most of the labeling tasks are performed on the web. A system that is intended to work in this environment should support workers with different operating systems and limited processing capabilities, and should adapt to various screen sizes [168].

With these requirements, we designed VASSL as a web-based application. The design assigns most of the computation intensive processes to the back-end of the system to reduce the load on the clients' machines. Moreover, the front-end is designed to have a flexible, responsive layout that allows the user to control the size of different views interactively. This feature is useful in focusing users' attention to particular parts of the information provided by VASSL. It is also helpful in generating a proper layout for different screen sizes for different clients. Both of these features were designed to satisfy (**R6**).

The most influential requirement that guides our system design is (**R1**). We built the entire system with the idea of showing similarities among accounts from different angles to support the interactive, human-guided clustering of accounts (**R4**). Four of the views in Figure 5.1 are capable of showing similarities among the accounts. The timeline view, in particular, is designed to show similarities among accounts using different temporal aggregation levels to satisfy (**R2**). Once a potential cluster of accounts is found, different interactive selection methods of VASSL can be used to highlight groups of accounts (**R4**), examine them in more detail (**R3**) and simultaneously label them as spambot or genuine. Such analysis saves human workers significant time (**R5**) because it allows the examination and labeling of accounts in parallel (Section 5.6.1), unlike the traditional manual labeling methods, which sequentially analyze and tag accounts one by one. Furthermore, the proposed parallel analysis enhances the effectiveness of human workers (**R5**) by revealing valuable insights needed to detect the new type of spambot that spreads spam in groups (Section 5.6.1).

To summarize the content of tweets (**R3**), we included two techniques in the system design; word cloud visualization [169] and topic modeling using Latent Dirichlet Allocation (LDA) [170]. The former is a well-known text visualization technique which reveals words that frequently appear in accounts tweets. Topic modeling, on the other hand, was chosen to show similarities among accounts in terms of their posting topics, which are used as a way to cluster accounts.

5.2.1 Targeted Users

Our system utilizes sophisticated techniques and visualizations that require training and expertise. Our goal is to provide new functionalities to more effectively and efficiently identify spambots, to ultimately reduce the time and cost of recruiting expert annotators and the annotation process (**R5**). The targeted users are human annotators whose terminal goal might be generating labelled datasets, or using the tool to understand and characterize spambot behaviour in a dataset. This guided us through many design choices with a focus on utility. However, for these functionalities to be useful, the users are required to have a certain level of expertise.

We made several assumptions about user expertise to operate our system effectively. The most important was users' knowledge of tuning machine learning models; specifically, dimensionality reduction and topic modeling. VASSL is designed to provide experts with interactive control of these techniques which rely heavily on parameters tuning. Familiarity with these tools and experience labeling social spambots will significantly improve users' experience with VASSL.

5.3 Data Processing and Analysis of VASSL

In this section, we describe the main functionalities of the back-end of VASSL, which can be summarized as: *a*) feature extraction, *b*) dimensionality reduction, *c*) topic modeling, and *d*) data communication.

The first functionality is the extraction of a set of features that represent Twitter accounts. We built on previous research to check the types of features that are known to be useful in spambot detection [70, 71, 74]. We identified a set of fifty features, e.g. total tweet count, average number of links, followers to following ratio, then generated four representations of these features by aggregating them temporally (**R2**).

Some of the defined features are derived by applying sentiment analysis to accounts' tweets, which is calculated offline. These features are useful for identifying spambot accounts [74] and can also provide overview information about the content of tweets (**R3**). We extracted several features that reflect the sentiment of accounts' tweets such as polarity and subjectivity scores. These scores are calculated by matching tweet words to corpus words that have been pre-labeled, e.g. movie review corpus [171]. We tokenize tweets' text, assign scores to the tokens by matching them with the labeled corpus words, then calculate the sentiment by taking a weighted sum of the assigned tokens' scores. Similarly to other features, the sentiment polarity and subjectivity is aggregated for each account. This aggregation generates time-independent features (aggregation over all tweets of the account) and three sets of time-dependent features (aggregation by year, month, or day). Because of the size of the data, this analysis is conducted offline as a preprocessing step.

The second functionality performed by the back-end is the generation of a two-dimensional representation of the accounts, appropriate for clustering tweets in a 2D display (**R1**). This reduction is useful to communicate similarities between accounts to the user. VASSL uses the extracted time-independent features for this purpose. We incorporate four different dimensionality reduction (DR) techniques: Kernel Principle component analysis (K-PCA) [172], Linear Discriminant Analysis (LDA) [173], Locally Linear Embedding (LLE) [174], and t-distributed Stochastic Neighbor Embedding (t-SNE) [175]. The four DR techniques are included in order to increase the effectiveness of the dimensionality reduction results in different contexts. For example, if the task is to label a set of unlabeled data without any information about labeled data, supervised DR methods such as LDA may not produce good results, unlike PCA or LLE, which are unsupervised solutions. However, if a user has already labeled parts of the data, she may utilize a supervised DR method for better class separation performance. Another factor that encourages our choice of DR techniques is the assumption of linearity. We give users multiple options for reducing the dimension of feature space using linear and non-linear mapping techniques.

VASSL supports two transformation methods which can be incorporated with the aforementioned dimensionality reduction methods: min-max normalization and standardization. Min-max normalization changes the range of the features and forces it to the range from 0 to 1. Standardization transforms the values to Z-scores. Such transformations are needed for some of the dimensionality reduction techniques. For example, PCA is known to be sensitive to differences in features variance and thus may performs badly if applied before normalization.

The third functionality of the back-end is topic modeling, which is performed by utilizing the Latent Dirichlet Allocation (LDA) model [170]. We use the generated topics as a way to cluster accounts (**R1**, **R4**), as explained in Section 5.4.4. VASSL employs multiple natural language processing techniques, such as lemmatization and stemming preprocessing, to transform the set of tweets for each account to tokens in a form suitable for LDA. The system then applies LDA to the set of tokens and generates a set of topics that best represent accounts' tweets. To increase the accuracy of the latent topics, the system applies LDA to each temporal aggregation level mentioned above.

One of the challenges we faced during VASSL development was the data scale. To discover anomalous botnet spammers, analyzing large numbers of accounts simultaneously is an ideal way to reveal patterns. However, the size of the data handled increases exponentially with the number of accounts, because we need to consider multiple representations of each account. To overcome this issue, the back-end keeps a communication channel open with connected clients, gradually feeding data. This channel is a querying mechanism between the front-end and the back-end, which only sends data that is visible in the views of the front-end and prepares the remaining in the back-end. This reduces the problem of information overload and creates a better analysis experience for the users.

The communication channel also facilitates modification of the behavior of automatic data analysis techniques according to users' input. For example, users are able to change the parameters of dimensionality reduction and topic modeling techniques from the front-end.

5.4 Visualization and Interaction Design of VASSL

The front-end of VASSL consists of five views along with their control panels (Figure 5.1).

5.4.1 Timeline View

The timeline view visualizes the distribution of time series features that represent Twitter accounts at three different aggregation levels. Influenced by [176], we choose to use the box plot for its simplicity yet capability of visualizing complex multivariate data, such as our time series. The design of the visualization combines a bubble chart and a box plot to enable the user to select individual accounts, while observing class statistics (**R1**). Accounts are visualized in this view as points in temporally sorted facets. Each account has a representation in each facet to communicate changes in time. The accounts are grouped in a facet according to assigned labels into genuine, spambot, or unlabeled, which are the three levels of the x-axis of the facet. These groups are color-coded as green, purple and blue respectively. The y-axis of the timeline view represents one or more time series such as the total tweet count and the average number of hashtags in tweets, depending on user selection. The orange boxes on top of each facet are temporal selectors which can be used in temporal zooming interaction as explained below.

VASSL supports three main user interactions with the timeline view. Hovering over the quartile boxes increases their transparency, which helps users examine the underlying distribution of accounts underneath the boxes. Users can also zoom in time by moving the mouse pointer inside a facet and scrolling up and down to zoom in and out. Zooming functionality changes the aggregation level of the time series to year, month, or day levels (**R2**, **R3**).

The last interaction supported in the timeline view is the account selection. Selected accounts are highlighted using red color. Users can click on the points representing the accounts to select and highlight the accounts. Users can also select accounts by brushing on any facet to select accounts that overlap with the brush. VASSL highlights selected accounts in every time facet as well as in all other views. Linking time series allows users to examine trends and anomalies for selected accounts over time. Moreover, linking the views allows users to examine different information about the selected accounts such as their tweets, their position in the feature space, etc (**R4**).

Users can view multiple time series, which divides the visual space of the timeline view among selected time series (Figure 5.2). This allows simultaneous exploration of multiple time series features using a small-multiple-like representation [177]. The control panel also enables users to change the temporal resolution of the view. The visualizations of time series automatically adjust to match the selected time resolution.

5.4.2 Dimensionality Reduction View

VASSL visualizes the results of the dimensionality reduction (DR) techniques, explained in Section 5.3, in a 2D scatterplot (Figure 5.1 (B)). The effectiveness of 2D scatterplots in visualizing DR results, while maintaining cluster separability has been shown in [178]. Our scatterplot allows exploring similarities among accounts (**R1**), which are color-coded according to their class. The bubble size represents the tweet count feature of each account. Similar to the timeline view, users can select accounts by clicking or brushing. Moreover, the view can be panned and zoomed as needed.

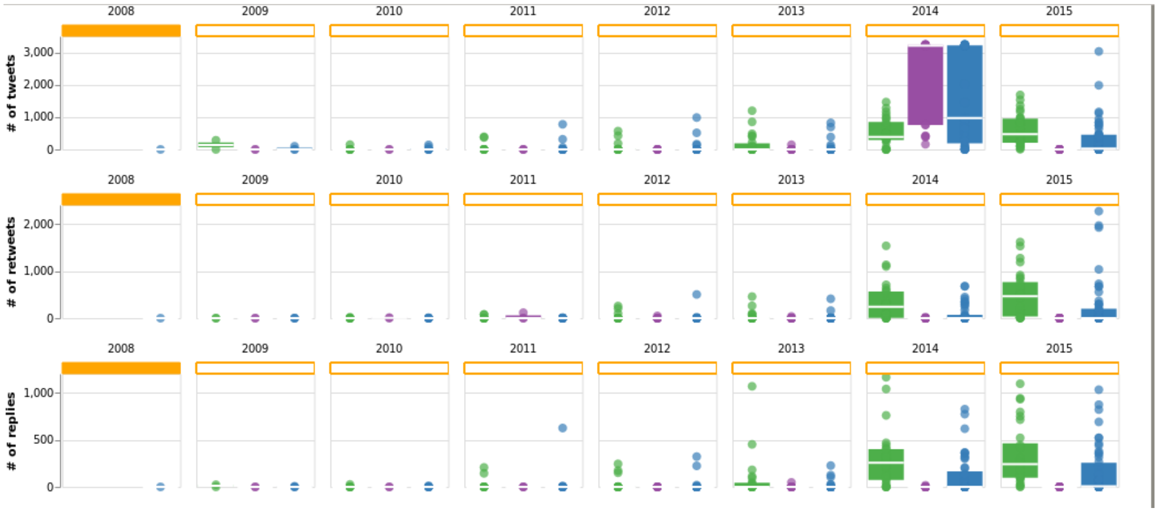


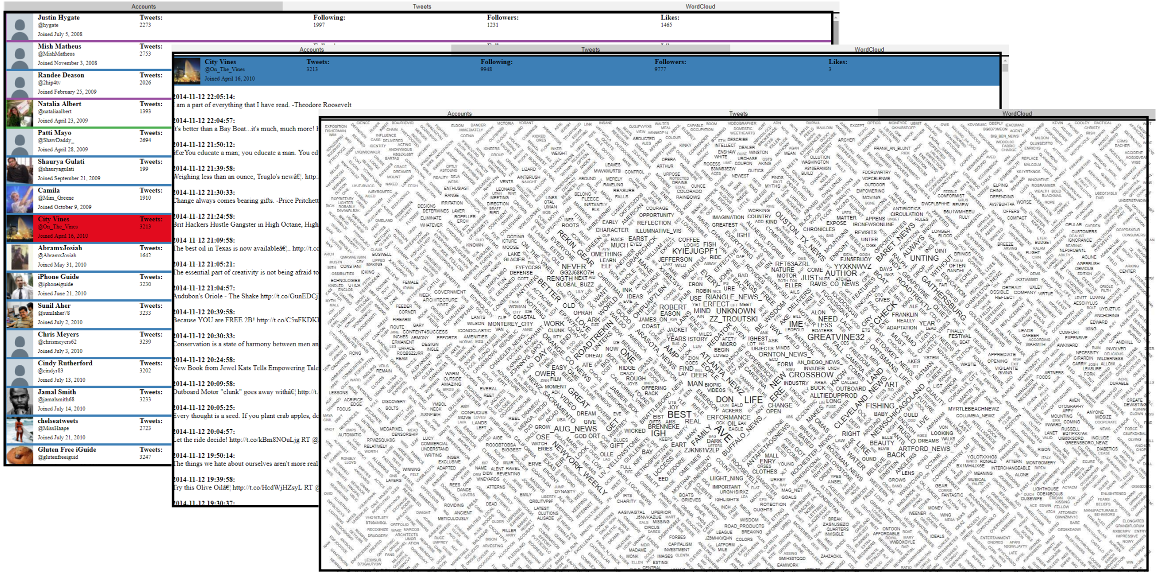
Fig. 5.2.: The timeline view visualizing three time series features at the year aggregation level. Blue shows unlabeled accounts, green for genuine accounts, and purple indicates spambots.

The users can select the specific DR technique to be used in the view from the four supported DR techniques. Users can also tune the hyperparameters of the techniques. For example, users can select the kernel function to be used with PCA, the number of neighbors to be considered in LLE, the perplexity of t-SNE, etc. The control panel also allows users to select a pre-reduction transformation including min/max normalization and score standardization, as explained in Section 5.3.

5.4.3 Users/Tweets Details Views

Three tabs are used in the details view: accounts' cards view, tweets view, and tweets' word cloud view. The accounts' cards view shows a list of accounts and other useful information, including accounts' names/screen names, profile images, joining date, total tweets, number of followers and followees, and the number of likes.

Users can select an account by clicking on its card (linked to other views). Once an account or a set of accounts are selected, users can access the tweets view, which shows the selected accounts' tweets in chronological order (**R3**). Including tweet text during the



Justin Hygate
@hygate
Joined July 5, 2008
Tweets: 2273
Following: 1997
Followers: 1231
Likes: 1465

Mish Matheus
@MishMatheus
Joined November 3, 2008
Tweets: 2753

Randee Deason
@2hip4tv
Joined February 25, 2009
Tweets: 2026

Natalia Albert
@nataliaalbert
Joined April 23, 2009
Tweets: 1393

Patti Mayo
@ShawDaddy_
Joined April 28, 2009
Tweets: 2694

Shaurya Gulati
@shauryagulati
Joined September 21, 2009
Tweets: 199

Camila
@Mim_Greene
Joined October 9, 2009
Tweets: 1910

City Vines
@On_The_Vines
Joined April 16, 2010
Tweets: 3213

AbramxJosiah
@AbramxJosiah
Joined May 31, 2010
Tweets: 1642

iPhone Guide
@iphoneiguide
Joined June 21, 2010
Tweets: 3230

Sunil Aher
@sunilaher78
Joined July 2, 2010
Tweets: 3233

Chris Meyers
@chrismeyers62
Joined July 3, 2010
Tweets: 3239

Cindy Rutherford
@cindyr83
Joined July 13, 2010
Tweets: 3202

Jamal Smith
@jamalsmith68
Joined July 14, 2010
Tweets: 3233

chelseatweets
@MissHaape
Joined July 21, 2010
Tweets: 2723

Gluten Free iGuide
@glutenfreeiguid
Tweets: 3247

City Vines
@On_The_Vines
Joined April 16, 2010
Tweets: 3213
Following: 9948
Followers: 9777
Likes: 3

2014-11-12 22:05:14:
I am a part of everything that I have read. -Theodore Roosevelt

2014-11-12 22:04:57:
It's better than a Bay Boat...it's much, much more! h

2014-11-12 21:50:12:
â€œYou educate a man; you educate a man. You ed

2014-11-12 21:39:58:
Weighing less than an ounce, Truglo's newâ€¦. http:

2014-11-12 21:30:33:
Change always comes bearing gifts. -Price Pritchett

2014-11-12 21:24:58:
Brit Hackers Hustle Gangster in High Octane, High

2014-11-12 21:09:58:
The best oil in Texas is now availableâ€¦. http://t.co

2014-11-12 21:05:21:
The essential part of creativity is not being afraid to

2014-11-12 21:04:57:
Audubon's Oriole - The Shake http://t.co/GunEDCj

2014-11-12 20:39:58:
Because YOU are FREE 2B! http://t.co/C5uFKDKI

2014-11-12 20:30:33:
Conservation is a state of harmony between men an

2014-11-12 20:24:58:
New Book from Jewel Kats Tells Empowering Tale

2014-11-12 20:09:58:
Outboard Motor "clunk" goes away withâ€¦ http://t.

2014-11-12 20:05:25:
Every thought is a seed. If you plant crab apples, do

2014-11-12 20:04:57:
Let the ride decide! http://t.co/kBm8NOuLjg RT @

2014-11-12 19:50:14:
The things we hate about ourselves aren't more real

2014-11-12 19:39:58:
Try this Olive Oilâ€¦ http://t.co/HcdWjHZsyL RT @

2014-11-12 19:30:37:

Fig. 5.3.: The three details views. Selecting one or more accounts from the cards view shows their tweets in the tweets view and a word cloud of all the tweets in the word cloud view.

analysis is essential to utilize the human ability to detect automated text generation (as explained in Section 5.5. Instead of accessing the entire tweet text, users can use the word cloud view to visualize selected tweets using the word cloud visualization technique [169]. The word cloud visualization is helpful for revealing repetitions of wording in selected tweets, which can guide the exploration of tweet text and labeling (**R3**).

5.4.4 Topic Clustering View

This topic clustering view uses two visualizations, which help analyze the results of applying Latent Dirichlet Allocation (LDA) to accounts' tweets. The first visualization is a bubble chart that represents the generated latent topics in a two-dimensional space (Figure 5.1 (D)). The axes of this visualization can be chosen from the control panel to be either unique IDs, topics polarity, or topics subjectivity, which are calculated by applying sentiment analysis to the topics' most probable words. The size of the bubble encodes a score for each topic which represents the sum of probabilities of posting in that topic by all the accounts (see Equation 5.1). The score of a topic T_i is calculated by summing up the probability of that topic in all j documents $D_n, \forall n \in \{1, 2, \ldots, j\}$. Documents in our analysis are accounts represented by the concatenation of all their tweets.

$$T_i Score = \sum_{n=1}^{j} P(T_i|D_n) \tag{5.1}$$

The second visualization in the topic clustering view is a word cloud visualization of the most frequent words in generated topics. LDA computes probabilities that show the distribution of these words in each topic. VASSL utilizes these scores to determine the size of the words in the word cloud. When a user hovers over a topic, the word cloud changes the size of the words according to their relevance scores, which help the user in exploring the semantics of the topics.

Users can interact with the topics bubble chart in multiple ways. Beside zooming and panning, the visualization supports selection and brushing of topics, which allows for exploration of each topic's words. When selecting multiple topics, VASSL aggregates words' probabilities to show their relevance to all selected topics (Figure 5.4).

Selecting a topic results in the selection of all accounts that have posted in that topic, with probability more than a threshold selected by the user (**R1**, **R4**). In other words, we can consider the topics as clusters where a single user can belong to more than one cluster (fuzzy clustering). Changing the threshold controls the sensitivity of cluster membership. Increasing the threshold narrows the results to accounts that post in a topic frequently, while reducing the threshold includes accounts that may rarely post in a topic.

The control panel for topics clustering view enables users to tune LDA hyperparameters. This includes the number of latent topics to generate, and the alpha and beta parameters which control the distribution of documents/topics and topics/words respectively. VASSL uses symmetric Dirichlet distributions. Assigning low alpha value results in modeling an account's tweets with a few numbers of topics and vice versa. Similarly, using low beta values results in modeling a topic using few words and vice versa. Tuning the three parameters enables users to control the generated topics to increase the effectiveness of the LDA.

Topic modeling of tweets is performed with respect to the selected temporal resolution. The topic clustering view is linked to the timeline view. When a user zooms into a particular period using the timeline view, the topics clustering view updates the topics to match the selected period (**R2**). This enables users to examine the change of tweets topic in time. The trade-off of this flexibility lies in the efficiency of topic modeling; because of the massive size of the documents, the topic modeling procedure is not performed at interaction speed, producing lag times every time the hyperparameters are changed. However, by keeping the hyperparameters constant, the topic views are generated and cached in advance of interaction.

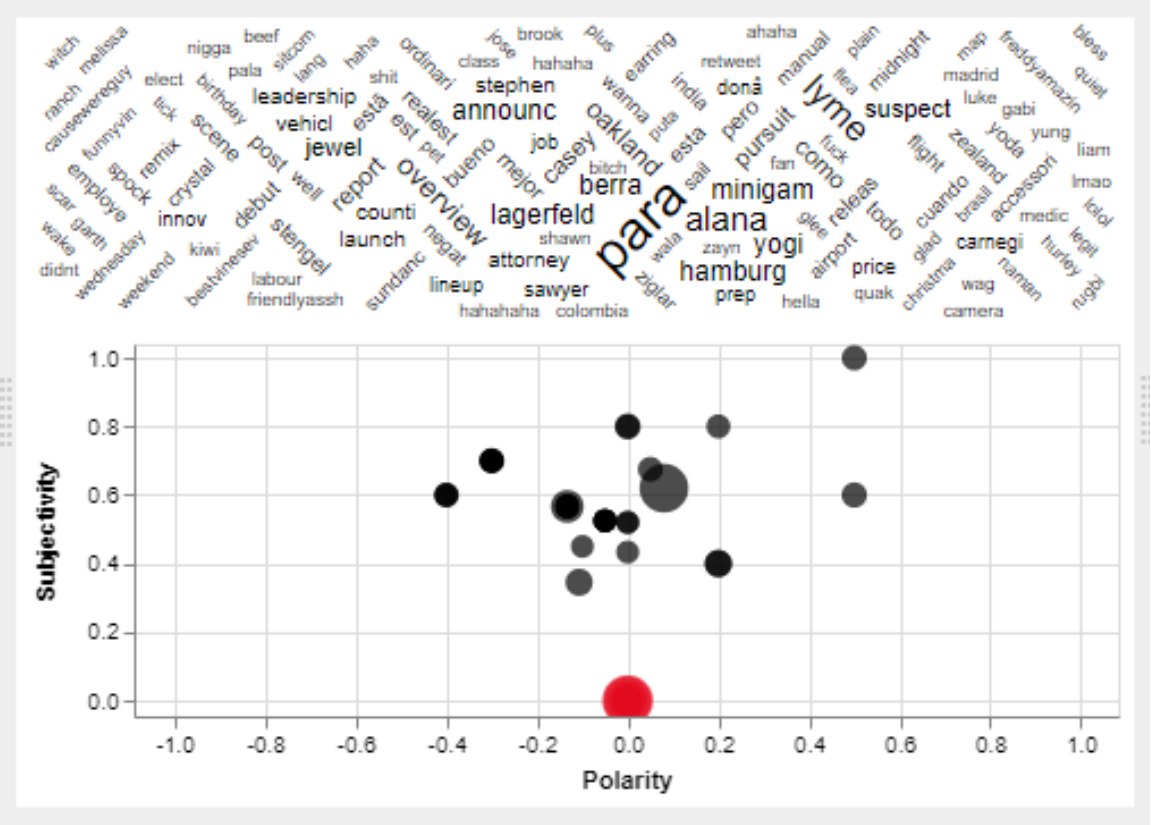


Fig. 5.4.: The topics clustering view with one topic selected. The bubble chart in the bottom shows the topics with bubble size communicating topics' scores. The word cloud on the top shows the topics' words with word sizes representing words-topics distribution.

We evaluated the time needed to perform topic modeling of 100 accounts at year aggregation level (the worst case scenario) on a cloud server with Intel Xeon CPU E5-2650L v3 @1.80GHz. It took an average of 69.09 seconds (with 1.33 SD) for 15 trials with different hyperparameters to complete the topic modeling and visualize the results. We observe a strong correlation between completion time and topic count as anticipated, and we report the completion time according to a randomly generated number of topics in the range from 1 to 100.

To overcome the time limitation of topic modeling, we implemented two solutions in VASSL. The first is to generate LDA topics for predefined profiles (pre-determined assignment of the hyperparameters) during the preprocessing stage. This allows VASSL to load the topics for these profiles at interaction speed. This method, however, would only be useful in cases where users do not consider tuning the hyperparameters online. The second solution is the usage of a spinner wheel that only disables interaction with the topics clustering view during LDA back-end calculations. This allows the user to interact with all other views of VASSL while computation is completed.

5.4.5 Feature Explorer View

The Feature Explorer view visualizes the distribution of accounts in selected features using a new design based on a violin plot (see Figure 5.5). We used a violin plot instead of a box plot to enable the user to examine multi-modality in any feature [179], which could indicate a potential cluster (**R1**, **R4**). Users can select as many features as needed in the feature explorer control panel, which contains a list of all features (see Figure 5.6). The maximum number of features that can be visualized simultaneously depends on screen size and human perception. Selecting features divides the available visual space among the features, and thus can reduce the capability of absorbing communicated information.

The feature explorer view shows statistical summaries for each feature independently, such as median and quartiles of the overall accounts as well as labeled groups. Features are represented as multiple horizontally adjacent facets, having the same Y-axis range in order to correlate the features. The accounts are represented as points in the horizontal center of each facet. The vertical locations of the accounts in a facet are determined by the value of the feature for these accounts. To reduce the visual clutter of the feature explorer view, the accounts point is transparent by default. Hovering over a class distribution increases the opacity of the accounts that belong to the hovered class. The black solid line in the horizontal center of each facet represents the 1st and 3rd quartile of all accounts regardless of their class while the black tick mark represents the median of all accounts. Besides its job of communicating quartile information of all accounts, the solid black line divides the facet into two areas. The area on the left of the solid line is used to indicate spambot and genuine class distributions (purple and green curves respectively) which are approximated by kernel density estimation (KDE) technique. The right area is used to communicate the KDE of the unlabeled accounts distribution as well as selected accounts distribution (blue and red curves respectively). These curves are visualized in each facet in a similar manner to communicate this statistical information in each feature (**R4**).

Users can interact with the feature explorer view to examine class statistics in each feature or to select a set of accounts. Hovering over a distribution curve reveals more information about the class it represents, including the median, the 1st and 3rd quartiles, and the accounts belonging to that class. The feature explorer view supports selecting and brushing, which is linked to all other views. When a user selects a set of accounts, the feature explorer draws the distribution curves of highlighted accounts in each opened feature facet. Figure 5.5 shows an example of five selected features as well as an illustration of the supported interactions. The control panel of the features explorer view allows the user to transform the features with similar transformations as described in Section 5.4.2, using min-max normalization or standardization to Z-scores, which is useful for comparison and outliers detection.

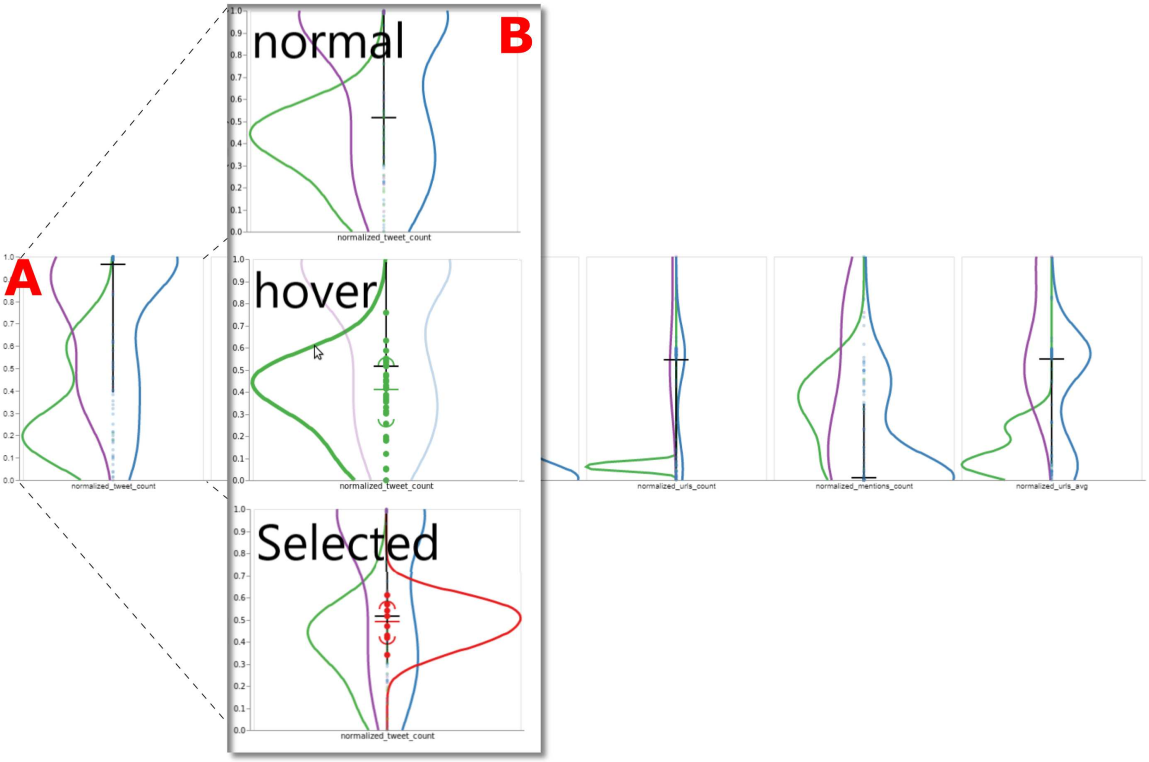
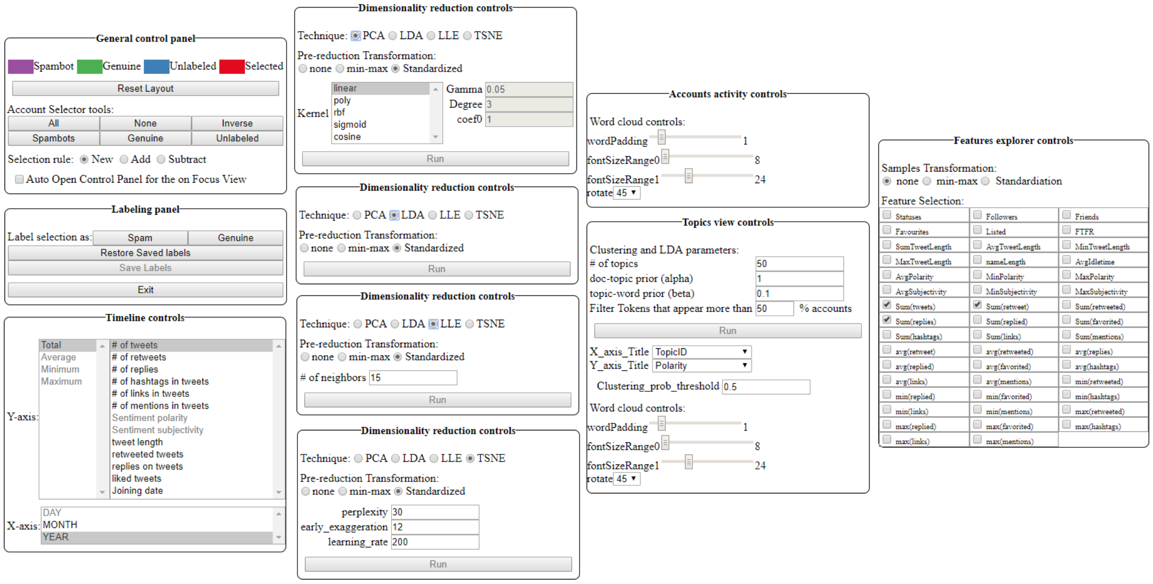


Fig. 5.5.: The Feature explorer view using a modified violin plot. The lines represent a kernel density estimation of classes PDFs. The accounts are distributed in the middle of each facet to facilitate selection. Blue represent unlabeled accounts, green for genuine and purple for spambots. A) Multiple facets representing a set of selected features. B) The effect of some user interaction on one of the selected features.

5.4.6 The Control Panels Area

The general control panel contains some functionalities that are applied to all views. It provides a legend that explains the color code we chose after consulting ColorBrewer [180]. The general control panel also facilitates multiple selection tools and rules that allow the user to perform complex selection queries by simple button clicks. This includes selecting all accounts, selecting inverse, selecting none and selecting all accounts of a particular class type. VASSL supports three selection rules: New, Add and Subtract. Choosing the



General control panel

🟪 Spambot 🟩 Genuine 🟦 Unlabeled 🟥 Selected

Reset Layout

Account Selector tools:

All | None | Inverse

Spambots | Genuine | Unlabeled

Selection rule: ● New ○ Add ○ Subtract

□ Auto Open Control Panel for the on Focus View

Labeling panel

Label selection as: Spam | Genuine

Restore Saved labels

Save Labels

Exit

Timeline controls

Y-axis:
Total
Average
Minimum
Maximum

# of tweets
# of retweets
# of replies
# of hashtags in tweets
# of links in tweets
# of mentions in tweets
Sentiment polarity
Sentiment subjectivity
tweet length
retweeted tweets
replies on tweets
liked tweets
Joining date

X-axis:
DAY
MONTH
YEAR

Dimensionality reduction controls

Technique: ● PCA ○ LDA ○ LLE ○ TSNE

Pre-reduction Transformation:
○ none ○ min-max ● Standardized

Kernel
linear
poly
rbf
sigmoid
cosine

Gamma 0.05

Degree 3

coef0 1

Run

Dimensionality reduction controls

Technique: ○ PCA ● LDA ○ LLE ○ TSNE

Pre-reduction Transformation:
○ none ○ min-max ● Standardized

Run

Dimensionality reduction controls

Technique: ○ PCA ○ LDA ● LLE ○ TSNE

Pre-reduction Transformation:
○ none ○ min-max ● Standardized

# of neighbors 15

Run

Dimensionality reduction controls

Technique: ○ PCA ○ LDA ○ LLE ● TSNE

Pre-reduction Transformation:
○ none ○ min-max ● Standardized

perplexity 30

early_exaggeration 12

learning_rate 200

Run

Accounts activity controls

Word cloud controls:

wordPadding 1

fontSizeRange0 8

fontSizeRange1 24

rotate 45 ▾

Topics view controls

Clustering and LDA parameters:

# of topics 50

doc-topic prior (alpha) 1

topic-word prior (beta) 0.1

Filter Tokens that appear more than 50 % accounts

Run

X_axis_Title TopicID ▾

Y_axis_Title Polarity ▾

Clustering_prob_threshold 0.5

Word cloud controls:

wordPadding 1

fontSizeRange0 8

fontSizeRange1 24

rotate 45 ▾

Features explorer controls

Samples Transformation:
● none ○ min-max ○ Standardiation

Feature Selection:

□ Statuses	□ Followers	□ Friends
□ Favourites	□ Listed	□ FTFR
□ SumTweetLength	□ AvgTweetLength	□ MinTweetLength
□ MaxTweetLength	□ nameLength	□ AvgIdletime
□ AvgPolarity	□ MinPolarity	□ MaxPolarity
□ AvgSubjectivity	□ MinSubjectivity	□ MaxSubjectivity
☑ Sum(tweets)	☑ Sum(retweet)	□ Sum(retweeted)
☑ Sum(replies)	□ Sum(replied)	□ Sum(favorited)
□ Sum(hashtags)	□ Sum(links)	□ Sum(mentions)
□ avg(retweet)	□ avg(retweeted)	□ avg(replies)
□ avg(replied)	□ avg(favorited)	□ avg(hashtags)
□ avg(links)	□ avg(mentions)	□ min(retweeted)
□ min(replied)	□ min(favorited)	□ min(hashtags)
□ min(links)	□ min(mentions)	□ max(retweeted)
□ max(replied)	□ max(favorited)	□ max(hashtags)
□ max(links)	□ max(mentions)	

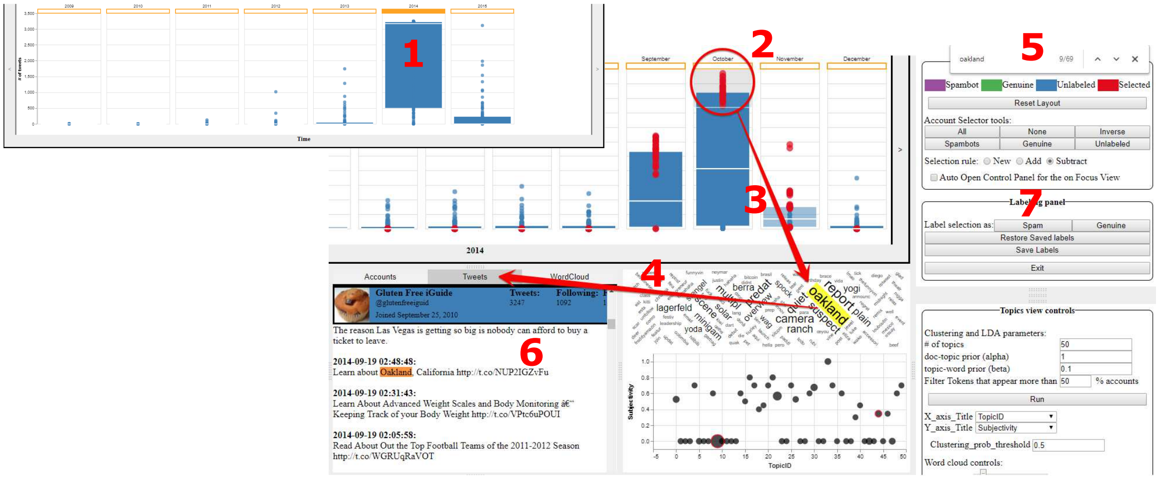
Fig. 5.6.: The control panels. Users can open a control panel of a view by pressing the “C” button on the keyboard while hovering over the view.

“New” rule unselects previously selected accounts, then selects whatever a user selects. Choosing the “Add” rule keeps existing selected accounts, and add whatever a user selects to the selection set. Choosing the “Subtract” rule removes whatever a user selects from any existing selection.

Beneath the general control panel is the labeling panel, which allows the user to label selected accounts as spambots or genuine. Labels are automatically saved in a database managed by the back-end. Figure 5.6 shows the five specific control panels for each view.

5.5 A Use Case Scenario of VASSL

In this section, we demonstrate how to utilize VASSL to support the process of detecting and labeling spambots. This scenario is supposed to represent labeling tasks that are traditionally performed manually by recruited human workers who inspect and annotate a set of Twitter accounts. We take the role of the human workers and present the insights reached with the help of VASSL while interacting with its views. We note that this analysis



Gluten Free iGuide
@glutenfreeiguid
Joined September 25, 2010
Tweets: 3247
Following: 1092

The reason Las Vegas is getting so big is nobody can afford to buy a ticket to leave.

2014-09-19 02:48:48:
Learn about Oakland, California http://t.co/NUP2IGZvFu

2014-09-19 02:31:43:
Learn About Advanced Weight Scales and Body Monitoring â€" Keeping Track of your Body Weight http://t.co/VPtc6uPOUI

2014-09-19 02:05:58:
Read About Out the Top Football Teams of the 2011-2012 Season http://t.co/WGRUqRaVOT

Account Selector tools:
Selection rule: ○ New ○ Add ● Subtract
□ Auto Open Control Panel for the on Focus View
Label selection as: Spam Genuine

Clustering and LDA parameters:
# of topics 50
doc-topic prior (alpha) 1
topic-word prior (beta) 0.1
Filter Tokens that appear more than 50 % accounts
X_axis_Title TopicID
Y_axis_Title Subjectivity
Clustering_prob_threshold 0.5

Fig. 5.7.: 1) Abnormality in tweet count on 2014. Scrolling changes the timeline view to month-aggregation-level for the 2014 period. 2) Brush on the month of October 2014 to select a group of accounts with abnormal tweeting frequency. 3) Identify frequent words in the most frequent topics the selected accounts post in. 4) Open tweet view to examine selected accounts tweets. 5) Search tweets for the captured most frequent word in outlined topics. 6) Examine the tweets contain that word which leads to a discovery of automated-like tweets. 7) Label selected accounts as spambots.

aims to showcase how to use VASSL and what possible insights could be derived with it; the next section also evaluates the toolkit formally through a user study.

A human worker, called Amy, started her analysis by loading 100 Twitter accounts and their tweets to VASSL. Her first glance on the timeline view (Figure 5.7 (1)) allowed her to spot an abnormal posting behavior in the year 2014. She zoomed into that period and examined the tweet count of the accounts in month-aggregation-level. She noticed that the unusual posting count happened in September and October where two groups could be separated based on tweeting frequency (Figure 5.7 (2)). Amy selected the group with a high posting count, which highlighted their representation in all the views. VASSL also outlined the topics posted by these accounts and the words that represent these topics. Amy found that the word with the highest score for the outlined topics was “Oakland” (Figure 5.7 (3)). She then examined the tweet view and searched for the word “Oakland” in selected accounts’ tweets (Figure 5.7 (4&5&6)). She found that all these accounts propagated the same tweet at different times in the months of September, October, and November of 2014. That tweet contained a link to an external website called “hub pages” that directed to a page with unfound contents. Examining other tweets posted by selected accounts showed a propagation of similar tweet content and links, but for other cities. Based on these findings, Amy to labeled selected accounts as spambots. The insights that led Amy to label these accounts as spambots were only reached as a result of analyzing multiple accounts as a group, which is the primary focus of VASSL.

After labeling the first group of accounts, Amy examined the dimensionality reduction view to find similar accounts. She found suspicious accounts that were not yet labeled and were very similar to the spambot accounts she had previously labeled. Amy examined the tweets of these accounts and found another account that propagated the same tweet about “Oakland”, so she included it with the spambots class.

Amy returned to the year-aggregation-level in timeline view and noticed that all the accounts that she labeled as spambots were inactive in all periods except in 2014 when they became abnormally active. All spambots posted more than 3000 tweets in 2014 with a very small variance among the accounts. By selecting all the accounts that tweeted more than 3000 times in 2014, Amy found one account that was unlabeled and had the same temporal pattern of posting as the group of spambots. She selected that account and compared it with the spambot accounts in the multidimensional feature space using the feature explorer view. She found that this account fitted well in the estimated distribution of spambots in most of the features, including tweet count, retweet count, reply count, URLs count, etc. This convinced Amy to label the account as a spambot even though the account tweeted on different topics than the pre-labeled spambots.

During her analysis, Amy gained a new insight into the sentiment of the topics. The group of spambots she had labeled usually tweeted about topics mostly constituted of words with very low subjectivity scores. This insight was derived from visualizing topics’ bubbles on the sentiment subjectivity axis (Figure 5.7). Moving from this finding, Amy selected all the topics that had low subjectivity scores, which highlighted the accounts that frequently tweeted about them. Removing spambots from the selected list showed that the remaining accounts tweeted on a variety of low subjective topics, unlike spambots, which commonly tweeted on the same low subjective topics. After examining the tweets of unlabeled accounts that tweeted on low subjective topics, Amy could not find any suspicious tweeting behavior. Moreover, these accounts did not follow the distributions of spambots in the feature explorer view; as a result, she did not label them as spambots.

In the topics clustering view, Amy found two topics with high tweeting scores, i.e. many accounts posting in those two topics. Both topics had a high subjectivity score. Examining topics' words showed that they consisted of curses and slang, which tend to appear naturally in humans' everyday expressions. Selecting those topics highlighted the accounts that posted in them frequently. Amy noticed from the dimensionality reduction view that the highlighted accounts were dissimilar to labeled spambots. She confirmed this dissimilarity by examining the distribution of these accounts in the feature explorer view and found that they could be separated from the distribution of spambots in many features. After reviewing the tweets from these accounts, Amy decided to label them as genuine.

Amy noticed that the separation between spambots and genuine accounts was significantly affected by the subjectivity of tweet topics. Thus, she decided to select all the topics that had a subjectivity score of more than 0.5 (subjectivity range = [0,1]) and examined their words. Most of these topics had the same word-topic distribution if the semantics of the words were considered. Selecting similar topics with high subjectivity scores highlighted four unlabeled accounts which clearly belonged to the distributions of genuine accounts.

With these insights, Amy was able to label 85% of the accounts. To label the remaining accounts, she examined them one by one and checked to see if she could observe similarities between them and any of the labeled accounts. At this stage, as it became harder to justify the labeling decision based on similarities only, Amy thoroughly examined each account's tweets. She started the exploration of tweets in the word cloud view to get an overview of the most frequent words that appeared in the accounts' tweets. A useful view in this stage was the dimensionality reduction view with linear discriminant analysis (LDA), to find the best 2-dimensions that separate the classes. Amy also opened ten features that had a clear distinction between the distribution of spambots and genuine accounts in the feature explorer view and kept track of the accounts' position in the feature space with respect to class distributions. Considering all the information that is communicated by VASSL, Amy was able to label all the accounts.

We compared the tags assigned by us (while assuming the role of Amy) to the available ground truth. We achieved an accuracy of 95% with one false positive and four false negatives for the spambot class. We labeled all 100 accounts within 15 minutes. This result was not meant to replace formal user study (the focus of the next section); we only provide the result to show how the steps taken in this section led to acceptable results. Identifying relationships among accounts and observing similarities in their behavior was critical to detecting the new type of social spambots.

5.6 Case Studies

In this section, We provide two case studies to apply the theoretical contributions we propose in Chapters 3 and 4.

5.6.1 Case Study 1: Choosing and Applying the Best Existing Method to Evaluate VASSL

To evaluate VASSL, we conducted a within-subjects user study with college students to evaluate the accuracy of labeling Twitter accounts using VASSL and to compare it with the typical manual labeling procedure. We also collected subjective opinion from the participants to capture the perceived usefulness and ease of use of our toolkit.

Participants We recruited 12 college students (11 male and 1 female). The major of most participants was Electrical and Computer Engineering. We limited the participation pool to individuals with basic knowledge about Twitter. Our participants were asked to work for up to 90 minutes and were compensated with \$10 for participation, plus \$5 as a motivation bonus when achieving $\geq$ 85% labeling accuracy.

Methodology Participants were asked to complete four sessions. In session (A), we asked the subject to manually tag 100 unlabeled Twitter accounts using a tool that provides typical information that can be found in Twitter, such as the tweets of the account, joining date, the total number of tweets, etc. The tool provided a list of accounts that need to be labeled and allowed the user to select an account from the list, examine its tweets and the account's information, and tag it (spambot/genuine). Participants were allowed not to tag an account at all. We gave the participants 30 minutes to complete that task but allowed them to finish before the time limit if they chose.

In session (B), we asked the participants to complete a 20-minute training session to learn about the different views of VASSL. The training started with a 10 minutes tutorial video, followed by up to 10 minutes in which we allowed the participants to explore VASSL's various functionalities.

After completing the training session, participants completed session (C) to test VASSL. Similarly to session (A), the participants were asked to label 100 unlabeled Twitter accounts in up to 30 minutes, this time using VASSL. Our study did not consider the learnability factor, so we allowed subjects to ask us questions during the session if they did not understand the information communicated by the views.

We controlled the order of taking the sessions to eliminate the confounders that could appear due to carryover effects across testing sessions. We randomly assigned participants to one of two groups, to identify who would use which labeling solution first. This element of the activity is essential for balancing out human factors and ensuring the validity of our results.

After completing the three sessions above, participants took a 10-minute exit questionnaire to communicate their personal opinions about the tools. The quantitative responses we collected from the survey follow the seven-point Likert scale proposed by Davis [181] to evaluate the perceived usefulness and ease of use. We also collected qualitative feedback from the participants to highlight issues in the current design of VASSL and suggest future research.

Hypothesis The null hypothesis in our objective experiment, which we aim to disprove, is: "The mean of labeling accuracy for manual labeling procedure is equal to the mean of labeling accuracy achieved by VASSL".

Data set During this experiment, we used the benchmark test set #2 by [163]. This data set contains 928 Twitter accounts (50% spambots and 50% genuine) and 2,628,181 tweets. The ground truth labels are available for this dataset. We took nine samples, each containing 100 accounts unique to that sample along with all its tweets. Each participant manually labeled one sample and tagged another sample with VASSL.

Results Figure 5.8 shows the performance of VASSL compared to manual labeling. We used four different metrics that measured the performance of labeling: precision, recall, accuracy, and F1 score. We bootstrapped the performance scores obtained from testing our participants and applied a single factor ANalysis Of VAriance (ANOVA) to test our null hypothesis for the four objective metrics. We found a significant improvement in the average accuracy [F(1,22)=9.7484, p=0.0049], average recall [F(1,22)=31.5232, p=0.00001], and average F1 [F(1,22)=13.6957, p=0.0012]. Adopting a significance level of 0.05, we reject the Null hypothesis for all these tests. However, we do not reject the Null hypothesis in the case of average precision [F(1,22)=0.8761, p=0.3594].

Another objective metric we collected to evaluate the effectiveness was the total number of tagged accounts in a test session. This metric replaced completion time, which could misrepresent participants who decided to end a session before tagging the entire test sample. Participants were able to label 76.36 accounts with VASSL on average, compared to 51.07 accounts for the manual labeling approach (with standard deviations of 7.9 and 8.92, respectively).

The bootstrapped results of the subjective scores we collected from the participants are presented in Figure 5.9. The figure reflects participants' opinions about the usefulness of VASSL, which increases effectiveness as observed in the first five factors. However, the figure shows significantly lower scores for VASSL in most of the ease of use factors compared to manual labeling.

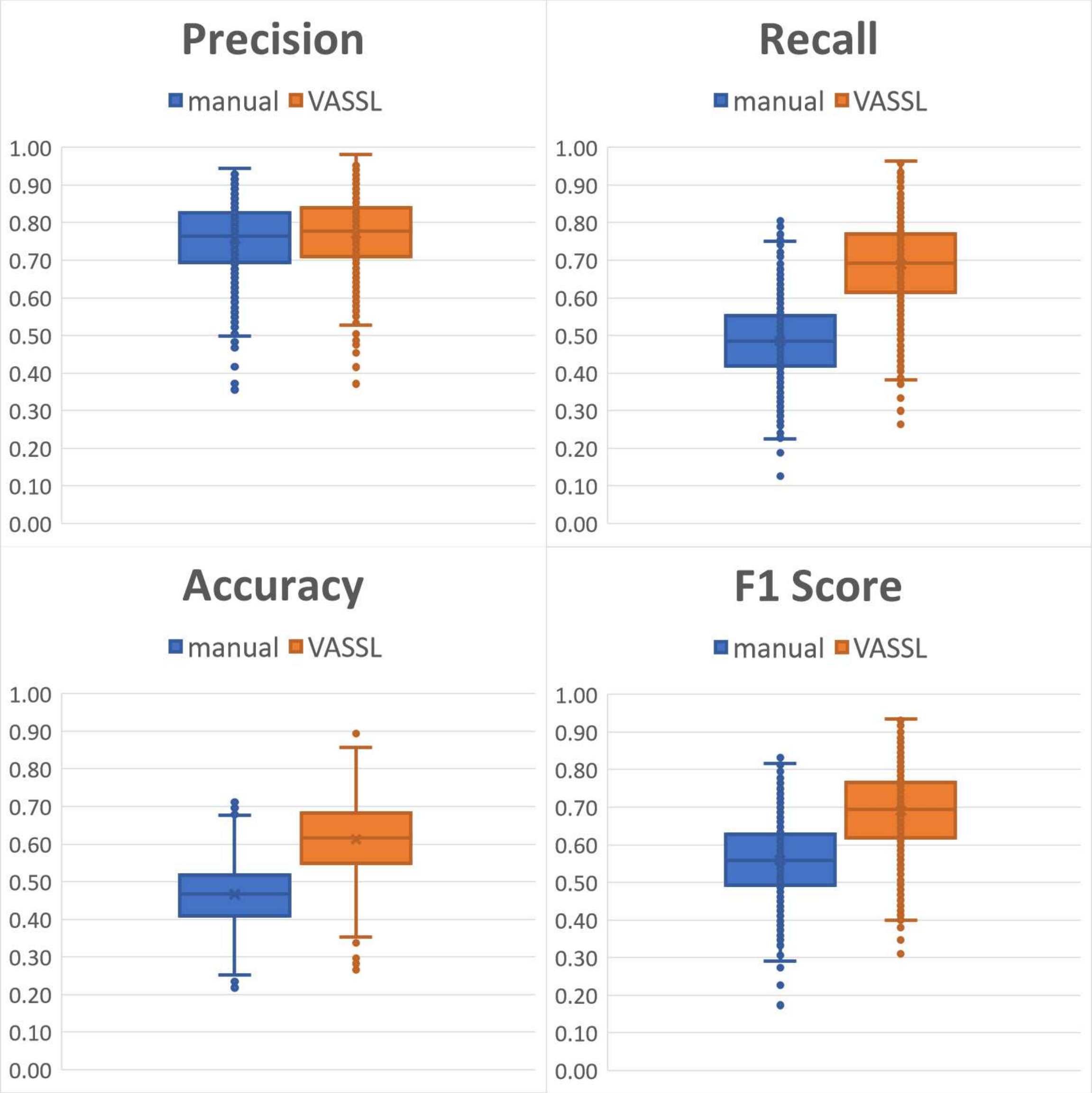


Fig. 5.8.: The objective performance of VASSL compared to manual labeling in terms of the precision, recall, and F1 score for the spambot class. We also include the overall accuracy of the labeling, which consider both spambot and genuine classes.

Findings and Implications

In this section, we provide more discussions on our experiment findings.

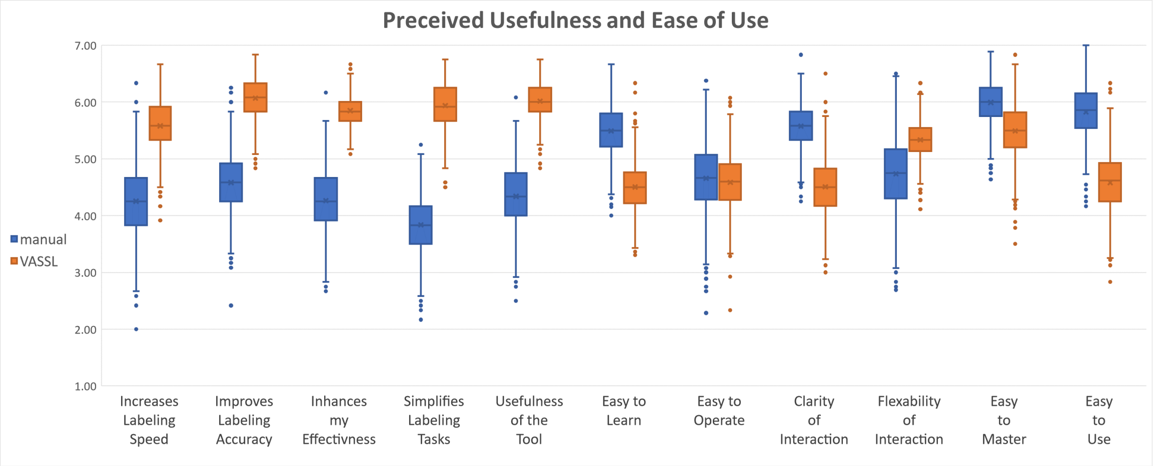


Fig. 5.9.: The subjective opinion about VASSL in terms of perceived usefulness and ease of use. The figure compares VASSL and manual labeling from user perspective.

Subjects are not experts Participants were college students instead of the targeted expert users due to availability. We acknowledge that the results may not represent an outcome with our intended users. However, we argue that better results would be observed if the system was tested by experts familiar with tuning clustering and dimensionality reduction methods as well as domain knowledge about spambots' behaviour. Many of the useful functionalities in VASSL were not utilized by our subjects because they were not familiar with them. For example, we noticed the common pattern of under-utilizing the tuning capabilities in the system. Subjects trusted the default values of the hyper-parameters and did not test any other alternatives that could have potentially improved their labeling performance. One can argue with confidence that an expert would not always accept the default values and would likely benefit from the flexibility of VASSL in enabling users to tune these hyper-parameters. Nevertheless, we acknowledge that testing the system with college students instead of expert users is a limitation we need to address in the future to gain more insight into other needed features as well as improvements to current features.

Objective results discussion The results of the conducted user study show an improvement in the performance of labeling spambots when compared to a manual labeling procedure. However, this improvement is not statistically significant for the precision criteria. We anticipated this result, because of the way our approach works. In some cases, clustering the accounts and analyzing them as groups could lead to false positives (genuine accounts labeled as spambots). For example, most of our participants reached an insight that enabled clustering many spambots using a single feature, i.e. zero number of likes for all the tweets, indicating a spambot. However, relying on this hypothesis could include genuine accounts that are less active on Twitter, or whose tweets are not liked as often. If the size of the cluster is big, the included genuine accounts may not undergo a detailed analysis, leading to incorrect labeling. To overcome this issue, we recommend first analyzing small clusters and carefully expanding them as needed.

Subjective results discussion The subjective scores collected from our participants highlighted a limitation of VASSL: although most of the participants found the tool useful, they thought it was difficult to use. Given the complexity of the system, this is mostly due to the steep learning curve required to work with the system. Participants worked with the system for only 30 minutes, after all. This finding can also be extracted from the qualitative data we have collected. The most common comment that appears in participant responses when asked about the most negative aspect in the system was “It is not too simple to use the tool.” Our focus during the design of the implementation was devoted to the utility aspects, targeting experienced analysts who regularly label Twitter spambots as our audience, so we included many functionalities that could help in various situations related to expert labeling. However, this focus on utility creates many usability limitations. One of the subjects stated, “I liked the multiple features which could have helped me accomplish my tasks, but it was challenging to learn to use them properly.”

Qualitative results discussion The exit questionnaire contained open-ended questions such as “What was your policy of labeling accounts as spambots?” and “What features/functionalities in VASSL helped you in implementing your policy?” We noticed from the collected answers that most of the subjects preferred to use one view as a primary view and confirm information from another view, but they rarely used more than two views. The DR view was the most utilized view and it was commonly complemented by examining selected accounts' details from the tweets view. Another common pattern was to select one feature in the feature explorer view, and select and analyze accounts according to pre-developed hypotheses such as "Low *like* counts." We can see from these patterns the variety of supported analyses in VASSL, which provides tools that supported different workflows.

5.6.2 Case Study 2: Comparing JA Evaluation Framework to Performance-based and Insight-based Evaluation Frameworks

The goal of this case study is to demonstrate the implementation of the proposed framework and to compare it to the traditional insight-based evaluation method.

The problem domain we chose to study concerns tagging spambot accounts in Twitter. Social media analysis is an active research topic in visual analytics [182, 183]. Identifying malicious accounts in these platforms, in particular, is an essential challenge [10]. Over 40% of all social media accounts are spambots, and their behavior keeps changing [184, 185]. Therefore, it is essential to generate labeled examples to train automated spam detection algorithms or to study the behavior of spambots.

The first step in conducting an evaluation study using JAVA framework is to define the environment. As we describe in Section 4.3, the problem instances perceived and analyzed are sampled from the set known as the environment, which represents all the variation of the defined problem. In this experiment, a problem instance is a Twitter account that needs to be labeled as a spambot or not. Thus, the tackled environment is supposed to include questions about the label of all accounts on Twitter. Because of the impracticality of studying this environment due to the unavailability of complete ground truth, we rely on a subset of the Twitter environment. This subset is defined using a benchmark dataset with known ground truth. We used the spambot benchmark dataset generated in [185], containing a collection of actual Twitter accounts manually labeled as spambots or genuine.

We report our evaluation experiment of two solutions used to tag Twitter accounts. The first solution, named Tool-A, is an interface that shows information commonly available on Twitter about an account, such as the number of followers and the texts of the tweets. Tool-A represents the labeling approach known as manual labeling, which works by examining the information available on Twitter to annotate an account. In this study, we compare Tool-A with Tool-B, a visual analytics tool featuring three interactive visualizations. Tool-B includes a timeline view, a feature explorer view, and an account details view to support human workers in reaching insights that could influence their labeling decisions. The timeline view provides interactive visualizations to explore the temporal change in time-dependent features representing the analyzed account (e.g., number of tweets and replies). The feature explorer view, on the other hand, visualizes features that represent the overall behavior of the analyzed account (e.g., followers to following ratio). Finally, the account details view provides annotators with the same information provided by Tool-A (e.g., the account's profile images and tweets.) Screenshots of Tool-A and Tool-B are shown in Fig. 1 and 2 in the supplementary material.

Our Judgment Analysis evaluation objective is to identify the better approach (i.e., Tool-A with human annotator Vs. Tool-B with human annotator) in terms of general performance and analytical support they provide. As we have mentioned, one of the three views of Tool-B is identical to Tool-A; therefore, we hypothesized that Tool-B should outperform Tool-A in terms of the provided analytical support given the additional capabilities. To test this hypothesis, we recruited human subjects to operate the two tools.

Setup We evaluated Tool-A and Tool-B in two independent sessions, following the representative design suggested by the JAVA framework (i.e., testing the evaluands with multiple instances representing the natural variation of the problem). Each session consisted of 16 instances that need to be solved by each subject. An instance asked to label a single Twitter account in 5 minutes given the information about that account. The first instance in each session was used as a tutorial to explain different components in the evaluated interfaces. Without the tutorial, each subject was tested with 15 instances of the labeling problem while using Tool-A in one session and another 15 instances while using Tool-B in the another session. The Twitter accounts used in the problem instances were randomly sampled from the benchmark dataset. After each labeling instance, we conducted a short semi-structured interview with the subjects to discuss the insights reached while solving the instance. Collecting these results allowed us to compare Tool-A and Tool-B using our proposed Modeled knowledge (G) and Total Analytical Support **TAS** metrics.

Participants We recruited ten graduate students (eight male and two female) majoring in Electrical and Computer Engineering. We limited the participation pool to individuals with basic knowledge about Twitter. Our participants were asked to work for 180 minutes (2 sessions, 10 minutes tutorials, up to 150 minutes task completion, and 20 minutes post-labeling interviews). We chose 3 participants to participate in a pilot study, which helped in deciding on the number of instances that can be tested in the fixed experiment time of 180 minutes. These 3 participants were excluded from participating in the actual experiment, and we discarded the data collected about them during the pilot study. We assign ids to the remaining seven participants who are the subjects of the reported experiment.

Data collection We collected two types of data from each human subject for each session, following our JAVA framework. The first type is the 15-dimensional decision vector (Y_s) which contains the labeling decisions made by the subject for every problem instance tested in the session (e.g., the first element in the vector represent the assigned label for the Twitter account tested in the first instance).

We also collected a matrix of insights (I) for each subject, which represents the insights reached while solving each task. Identifying insights and their variabilities was done through a think-aloud protocol during the tasks, complemented by the post-labeling semi-structured interview after each instance. We developed two interfaces that helped us in collecting reported insights and their variations promptly (Fig. 5 and 6 in the supplementary materials.) Moreover, we recorded the screen, utilized the recordings during the analysis phase to establish the insights matrix. At this stage, the collected matrix of insights for the subjects are not mutually exclusive since multiple subjects can reach the same subset of insights. A total of 173 insights were identified by all subjects. When a subject reported an insight, we discussed it with the subject in the interview time to identify a suitable measuring scale for that insight and capture it's measurement as seen by the subject. The default measurement for any insight is null, which represents the case when that insight is not reported in a tested instance. After a subject finished a session, we end up with 15 measurements for each insight reported by that subject during the session (i.e., a measurement for each solved problem instance in the completed session.)

Data processing We first cleaned the list of captured insights and unified common ones, resulting in a smaller list of unique insights. Some of the insights were clearly identical, whilst others needed an understanding of their semantics to group them. For example, we considered ”consistency of posting about a topic” and ”Randomness of tweet content” to be identical insights that determine the extent of posting in related topics. To ensure a valid grouping, we confirmed the insights against the recordings and the observation notes that we had collected to understand each insight properly. By merging identical insights, we derived a new list of 65 unique insights (I_1 through I_{65}) from processing the original 173 insights.

The next processing step was to define the measuring scale of the 65 unique insights and to measure them for each subject. The I_i had resulted from grouping and unifying the original insight matrix members of each subject. Since we had already defined and measured each one of the original insights in those matrices, we were able to derive the measurements of the I_is in a straightforward manner. We only paid extra attention when each I_i is created from merging insights with different measuring scales. In these cases, we define a new scale for the I_i, which includes all the levels of the original insights that constitute it. For example, multiple subjects reported the insight ”number of personal topics in tweets,” but they usually reported it with different scales like [none, exists] or [none, small, a lot]. We thus defined the new measuring levels for this insight as [none, low, normal, high] and treat the level ”exists” as ”normal.” Moreover, we added a special level to almost all of the insights scales, namely the ”not reported” level. This level represents an insight measure when a subject does not report that insight at all in a given instance. This led us to use a nominal scale to measure all of the insights defined in this experiment.

Analysis After measuring the insights with appropriate scales, we organized the collected data in a form suitable for analysis. The 65 unique insights and their measures over the

30 tasks (15 tasks per session for two sessions) were organized into *two* 15×65 matrices (I) for each subject (i.e., a total of 14 matrices with same column titles representing the insights but with different measurements for each subject and for each tool). The first matrix represents the insights' measures when using Tool-A, and the second one represents their measures when using Tool-B. Labels assigned by a subject were also organized into two 15-dimensional vectors (Y_s for Tool-A and Y_s for Tool-B). Similarly, ground truth was organized in two 15-dimensional vectors (Y_e for Tool-A and Y_e for Tool-B). The bottom part of Fig. 4.1 shows these components for a single subject after a single session. Thus, the analysis we propose in JAVA are performed in this experiment 14 times (i.e., two times per subject for all the seven subjects, with calculations in the Supplementary Material). We note that the elements of the collected data structures (the measurements of the decision, ground truth, and the insights per problem instance) are different for different subjects, including the vectors of ground truth. Subjects were tested independently with test samples that were independently drawn from the benchmark dataset, resulting in a high likelihood to test subjects with different problem instances.

Using the data structures described in the previous paragraph, we performed our proposed judgment analysis for each subject individually. We estimated the decision policies of the subjects with Tool-A by modeling their respective decision vector Y_s using their respective insight matrix I. To simplify the discussion, we explain the analysis conducted with one subject and emphasize that a similar analysis was performed on every subject individually.

Many different statistical techniques can be utilized to estimate the models of decision policy and truth (e.g., logistic regression or decision trees.) To select the best modeling technique, tuning the estimated model of an individual decision policy, and testing the model's internal and external validity, we followed these steps: we started by randomly choosing 10 instances from the 15 instances tested with Tool-A and use their respective assigned labels to build a decision model. We used the leave-one-out cross-validation method [186] to select and tune the model of individual decisions according to the validity scores obtained for different tested candidate models. As described in step 5 of Section 4.3, the validity score for a model can be viewed as a measure for its internal validity (i.e., the extent to which the estimated model correctly represents the individuals' decision policy in the tested 10 instances). The 5 assigned labels of the 15 tested instances, which are held out of the training process, are used as a test set for quantifying the external validity of the selected model of the individual's decision policy.

After selecting the decision policy model, we used it to generate the modeled decision vector $\widehat{Y_s}$ (the labels that would be assigned by users if they followed the estimated decision model exactly). This vector represents the labeling of the model to the same instances tested in the experiment (i.e., the model prediction of the accounts' label when representing the accounts by the rows of the insight matrix.) We subsequently use equation 4.1 to calculate the mutual information between the columns of I, i.e., the insights, and the resultant vector $\widehat{Y_s}$ to measure the utilization validity W_s for each insight (i.e., their significance in determining the label of Twitter account from the modeled individual's perspective). As we have mentioned in section 4.3, decisions and insights are formulated as random variables in JAVA, thus the collected data constitute sample distributions. To accurately estimating the utilization validity of insights, we applied bootstrap sampling to the data collected in the tested instances (sampling with replacement) for 1000 times and used the average of measured utilization as the best estimate for insights' utilization validity.

The previous analysis was conducted to estimate the behavior of the user part (the right side of the Lens Model in Fig. 4.1). The second part of JAVA is to analyze the environment (the variation of the problem) to enable estimating generalizable performance of the subject. We followed a similar approach to model the ground truth of the problem instances tackled by Tool-A, internally and externally validated it using cross-validation and hold-out sets, and predicted the bootstrapped ecological validity W_e of each insight, i.e., the importance of the insight in the studied environment. The only change was the use of the vector of truth Y_e instead of the vector of subject decisions Y_s when modeling, which led to the generation of the modeled criterion vector $\widehat{Y_e}$ used to calculate the ecological validity of the insights.

Finally, using the bootstrapped utilization validity W_s and the bootstrapped ecological validity W_e, we apply formula 4.2 to calculate the Total Analytical Support **TAS** for the subject when he/she solve the tasks with Tool-A. This score quantifies the level of analytical support Tool-A provided to the individual subject when she\he attempts to label a Twitter account. We also computed the Modeled Knowledge **G** by calculating the accuracy of the individual's decision model in matching the truth model (i.e., comparing $\widehat{Y_s}$ and $\widehat{Y_e}$.)

To compare Tool-A with Tool-B for the analyzed individual subject, we repeated the evaluation process for the 15 cases tackled by Tool-B. This includes estimating the decision model and truth model, measuring insights' utilization validity and ecological validity, and calculating **G** and **TAS** when the subject used Tool-B. As a reminder, the previous paragraphs describe the analysis of one individual subject. We repeated the same analysis for each of our seven subjects independently, following the design of JAVA framework.

After applying the same analysis for all the subjects, we tested if an aggregation of subjects can be obtained through cluster analysis. As we have described in Section 4.3, clustering is important to discover potential populations inside the population of potential users. According to JA theory [8], these subsets of the whole population must be studied independently to generate finer conclusions compared to treating the population as a homogeneous. For example, our experiment seeks to find superiority of one of the tools over the other (Tool-A vs Tool-B). This superiority must be analyzed for each cluster independently.

The JAVA framework provides multiple ways to define similarities and differences between subjects to cluster them. Selecting one of these ways is determined based on the study objective. For example, one can cluster according to the similarity in utilizing insights, as we choose in this experiment. Another approach could be to cluster based on the degree of change in policy utilization when using Tool-A and Tool-B, which is useful when studying the superiority of one tool over the other for the group of users who have persistent decision policies and those who are flexible to adapting new policies. Once defining the approach and a suitable representation of subjects, we can apply a clustering technique to identify the clusters. We prefer using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [187] because it does not require pre-determining of the number of clusters, and for its capability to detect outliers. We were able to identify three clusters (Cluster A, Cluster B, and Cluster C), two of which have only one subject that are different than all the other subjects and a one cluster grouping four subjects. We analyzed these clusters in the

Table 5.1.: The results of the case study. The table compares evaluating performance using raw data ($\mathbf{r_a}$ in Fig. 4.1) vs. using the proposed modeled knowledge (**G** in Fig. 4.1). It also compares the analytical support metrics of insight count vs. the proposed Total Analytics Support (**TAS**). Internal and external validity of **G** and **TAS** are also included. The results are shown for the seven individual subjects (**S1** through **S7**) and for the identified clusters (Cluster A, B, and C).

Subject (cluster)	**Tool-A**						**Tool-B**					
	$\mathrm{r_a}$ (accuracy)	**G**	**Insight Count**	**TAS**	**Internal Validity**	**External Validity**	$\mathrm{r_a}$ (accuracy)	**G**	**Insight Count**	**TAS**	**Internal Validity**	**External Validity**
S1 (Cluster A)	100.00%	80.00%	13	4.82	72.00%	100.00%	93.33%	86.67%	11	3.09	64.00%	80.00%
S2	93.33%	86.67%	19	4.35	90.00%	100.00%	80.00%	80.00%	20	2.87	90.00%	80.00%
S3	66.67%	93.33%	14	-0.76	54.00%	80.00%	86.67%	93.33%	8	2.98	80.00%	80.00%
S4	53.33%	53.33%	23	2.42	81.00%	64.00%	53.33%	100.00%	21	0	63.00%	48.00%
S5 (Cluster B)	80.00%	93.33%	19	4.78	72.00%	80.00%	86.67%	100.00%	20	6.54	100.00%	64.00%
S6	46.67%	73.33%	19	3.60	20.00%	24.00%	73.33%	60.00%	15	1.40	80.00%	60.00%
S7	86.67%	80.00%	21	2.72	100.00%	80.00%	73.33%	80.00%	12	1.21	72.00%	60.00%
S2, S3, S4, S7 (Cluster C)	**average:** 75.00% **SD:** 18.36%	**average:** 78.33% **SD:** 17.53%	**average:** 20 **SD:** 4	**average:** 2.18 **SD:** 2.14	**average:** 81.25% **SD:** 19.75%	**average:** 81.00% **SD:** 14.74%	**average:** 73.33% **SD:** 14.40%	**average:** 88.33% **SD:** 10.00%	**average:** 16 **SD:** 7	**average:** 1.77 **SD:** 1.43	**average:** 76.25% **SD:** 11.50%	**average:** 67.00% **SD:** 15.79%

same manner as individual subjects. Next, we discuss the experiment results of individuals and the clusters.

Findings and Implications Table 5.1 summarizes the results of the experiment. The table compares the analytical support provided by Tool-A and Tool-B in terms of the traditional insight count metric as well as our proposed Total Analytical Support (**TAS**) metric. The table also shows our proposed modeled knowledge (**G**) metric that evaluates the performance of an individual in the tested environment. For **TAS** and **G**, the table also lists the internal and external validity of the measure using the internal and external validity of the developed models of subject policy and ground truth. Results are reported at both the individual level and the cluster level.

We found a marginal difference between Tool-A and Tool-B in terms of the analytical support they provide to our subjects. Clusters A and B only include one subject each, i.e., these users are different than all other subjects. Such limited number of subjects in these clusters prevented generalizing superiority of one tool over the other for the populations represented by these two clusters (i.e., we cannot apply any statistical testing to reject the null hypothesis of equal support for both tools). As for cluster C, the four subjects belonging to this cluster allowed us to statistically test the superiority of either tool using ANOVA. The result shows no statistically significant difference between the average **TAS** of Tool-A and Tool-B for cluster C [$F_{(1,2)} = 0.105$, $p = 0.757$]. This finding aligns with one of our observations about the utility of Tool-B. Most of our subjects relied primarily on tweets' text to discover spambots, which explains the similarity in both tools' analytical support. One can interpret this finding as the unwillingness of our subjects to adopt new policies that utilize the extra pieces of information provided by the extra visualizations of Tool-B, which can in part be caused by the limited training time we provided for them.

There is a divergence in the analytical support metrics we tested. More insights do not indicate more TAS scores in general. For example, S1 reported a total of 13 unique insights when he used Tool-A. However, according to the collected evaluation data, Tool-A provided 4.82 on the TAS scale in analytical support. Meanwhile, S4 reported 23 unique insights when using Tool-A, but his usage yielded a TAS score of 2.42.

One advantage of applying our framework is to increase the trustworthiness of the reported conclusions. Measuring the internal and external validity enables evaluators to decide whether to trust and accept the results of their analysis or not. For example, the evaluation results of subject S6 can be omitted when compiling an evaluation report, because S6' decision model suffered from low internal and external validity. The insights he was reporting were not affecting his observed decision behavior, thus were not sufficient to model him. Another advantage of reporting the internal and external validity is to be able to compare and rank the evidence generated from different replicated studies.

The primary objective of this case study is to compare the two ways of measuring analytical support (i.e., the traditional insight count vs. the proposed TAS). To uncover the superiority of one over the other, we correlate subjects' scores in both metrics with their achievement (i.e., their overall accuracy in labeling all tested instances). More analytical support should have more positive effect on the performance of the overall system, thus a good assessment of the analytical support should reflect this positive correlation. We found that the Pearson correlation coefficient between TAS and achievement is strong ($r = 0.68$, $n = 12$, $p = 0.01$). On the other hand, insight count had barely moderately **negative** correlation with the achievement ($r = -0.42$), which in the worst interpretation could mean that more insights could mislead users and adversely affect the performance. The reported results, however, cannot proof this claim because we fail to reject the null hypothesis of $r = 0$ for insight count ($n = 12$, $p = 0.17$). Despite this, the results we report still indicate the superiority of our proposed TAS metric over the traditional insight count when evaluating the analytical support of a visual analytics tools.

The reported results validate the benefits of applying our JAVA framework over the traditional insight-based method. However, it is important to note the feasibility cost of our framework mainly due to the time and resources required to collect and analyze the data. With the implementation of custom software libraries for calculating TAS and related scores, part of this cost can be alleviated, which we leave for future work.

5.7 Summary

In this chapter, we presented VASSL, a visual analytics toolkit that supports the analysis and labeling of social media accounts, for the specific use of identifying spambots. The datasets created using VASSL can be used in improved automated approaches for detecting spambots, whose nature and behavior changes dynamically to escape the current algorithms. With an effective tool such as VASSL, it is possible to quickly generate large annotated datasets that reflect the behavior of social spambots. We presented a detailed use case of the toolkit to perform a complete analysis and labeling task. Next, we evaluated VASSL and demonstrated the significant improvement in labeling performance. Finally, we conduct a case study with our JA evaluation framework to show case supperiority of it over traditional evaluation techniques in terms of its validity. In this case study, we use a short version of VASSL and compare it to manual labeling in the context of labeling a single twitter account at a time.

6. CONCLUSION

This thesis aims at improving the validity of current practices of evaluating VA solutions. After providing the necessary background about visual analytics domain and the concept of evaluation in scientific fields, we tackle two problems related to evaluation. The first problem is concerned with analyzing the validity, generalizability, and feasibility of current summative evaluation practices to highlight their limitations and propose a prescription for practitioners. We tackle this problem by building a taxonomy for the methods used in recent evaluation studies. Then, we recommend using summative quality and feasibility metrics, which we define to analyze the methods and reason about their applicability in different evaluation contexts.

The second problem we tackle in this thesis is to minimize the validity risk for the existing evaluation methods. To solve this problem, we propose a judgment analysis framework to improve the validity of performance-based and insight-based evaluation methods. We explain, in an abstract manner, the logic of problem-solving in VA and illustrate why it is important to rely on JA theory to evaluate VA systems. Then, we explain how to apply our proposed JA framework to evaluate VA by following seven steps. We propose to use two new metrics (the Modeled Knowledge and the Total Analytical Support metrics) to evaluate the performance and the analytical support of VA solutions.

To confirm the applicability of our theoretical evaluation framework, we select social spambot labeling as a case study. We implement VASSL, a VA solution that tackles the labeling problem. By utilizing the metrics proposed in Chapter 3, we show how to reason about selecting an evaluation method to prove the usefulness of VASSL. We then conduct an evaluation study using the prescribed method and confirm the usefulness of our VA solution. In the same context of social spambot labeling, we conduct another evaluation study to apply our JA framework proposed in Chapter 4. We compare the validity of the findings of our framework with the findings of traditional performance-based and insight-based methods.

The results of this study show superiority of our framework compared to the traditional ones in terms of validity at the cost of feasibility.

There is a set of work that can be explored in the future to expand the research done in this dissertation. The first is analyzing groupware VA systems designed for problem solving and decision making with elements of collaboration. In these solutions, multiple cognitive systems interact with a VA solution, which links their semantic representations of the data to derive a joint decision policy. Thanks to the conceptual linkage we have provided in Chapter 4 between JA and VA, we can reuse many of JA study designs that can help in assessing the role of collaboration in VA.

Another promising direction to pursue in the future is to study the activity performed by VA users to gain insights. Many studies target this problem by linking insight generation to users interaction [188–190]. Based on this idea, we can examine if an automatic suggestion system can guide the process of identifying insights. This could reduce the validity risks associated with the qualitative method used in identifying insights in our JA framework and systemize the process of identification.

The survey and the analysis we have provided in this dissertation include only the evaluation methods used in VA literature. An expansion and an enhancement to the proposed analyses is yet another future direction. We aim to examine methods that have not been utilized in VA literature and their applicability in the field. An example of these methods is the formal specification and verification method [191], which can evaluate the effectiveness of algorithms instead of measuring their efficiency.

As for the problem of social spambot labeling, one futurt work to investigate is extending VASSL by incorporating more machine learning models to provide suggestions for users. Similarly, user annotations should progressively improve automatic model suggestions. Future research can focus on testing the benefit of adding active learning and machine suggestion to spambot labeling. Furthermore, collaboration modules need to be created that can enable multiple users to collaborate to generate labels with minimal effort, with the help of machine-learning models as well as the insight generated by other collaborators.

REFERENCES

REFERENCES

[1] N. Sultanum, D. Singh, M. Brudno, and F. Chevalier, “Doccurate: A curation-based approach for clinical text visualization,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 142–151, 2019.

[2] R. A. Leite, T. Gschwandtner, S. Miksch, S. Kriglstein, M. Pohl, E. Gstrein, and J. Kuntner, “Eva: Visual analytics to identify fraudulent events,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 24, no. 1, pp. 330–339, 2018.

[3] A. Kumpf, B. Tost, M. Baumgart, M. Riemer, R. Westermann, and M. Rautenhaus, “Visualizing confidence in cluster-based ensemble weather forecast analyses,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 109–119, 2018.

[4] M. Angelini, G. Blasilli, T. Catarci, S. Lenti, and G. Santucci, “Vulnus: Visual vulnerability analysis for network security,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 183–192, 2019.

[5] N. Cao, C. Lin, Q. Zhu, Y.-R. Lin, X. Teng, and X. Wen, “Voila: Visual anomaly detection and monitoring with streaming spatiotemporal data,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 23–33, 2018.

[6] S. Guo, Z. Jin, D. Gotz, F. Du, H. Zha, and N. Cao, “Visual progression analysis of event sequence data,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 417–426, 2019.

[7] M. Cavallo and Ç. Demiralp, “Clustrophile 2: Guided visual clustering analysis,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 267–276, 2019.

[8] R. W. Cooksey, *Judgment analysis: Theory, methods, and applications.* Academic Press, 1996.

[9] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, “The rise of social bots,” *Communications of the ACM*, vol. 59, no. 7, pp. 96–104, 2016.

[10] V. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, F. Menczer *et al.*, “The darpa twitter bot challenge,” *arXiv preprint arXiv:1601.05140*, 2016.

[11] K. A. Cook and J. J. Thomas, “Illuminating the path: The research and development agenda for visual analytics,” Pacific Northwest National Laboratory (PNNL), Richland, WA (US), Tech. Rep., 2005.

[12] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim, “Knowledge generation model for visual analytics,” *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1604–1613, 2014.

[13] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler, "Visual analytics: Scope and challenges," in *Visual data mining*. Springer, 2008, pp. 76–90.

[14] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale, "Empirical studies in information visualization: Seven scenarios," *IEEE transactions on visualization and computer graphics*, vol. 18, no. 9, pp. 1520–1536, 2012.

[15] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller, "A systematic review on the practice of evaluating visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2818–2827, 2013.

[16] E. Bertini and D. Lalanne, "Surveying the complementary role of automatic data analysis and visualization in knowledge discovery," in *Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery: Integrating Automated Analysis with Interactive Exploration*, ser. VAKD '09. New York, NY, USA: ACM, 2009, pp. 12–20. [Online]. Available: http://doi.acm.org/10.1145/1562849.1562851

[17] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, *Visual Analytics: Definition, Process, and Challenges*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 154–175.

[18] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim, "Knowledge generation model for visual analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1604–1613, Dec 2014.

[19] M. Chen, D. Ebert, H. Hagen, R. S. Laramee, R. van Liere, K. Ma, W. Ribarsky, G. Scheuermann, and D. Silver, "Data, information, and knowledge in visualization," *IEEE Computer Graphics and Applications*, vol. 29, no. 1, pp. 12–19, Jan 2009.

[20] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale, "Empirical studies in information visualization: Seven scenarios," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 9, pp. 1520–1536, Sep. 2012.

[21] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller, "A systematic review on the practice of evaluating visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2818–2827, Dec 2013.

[22] J. J. Van Wijk, "The value of visualization," in *VIS 05. IEEE Visualization, 2005.*, Oct 2005, pp. 79–86.

[23] J. Stasko, "Value-driven evaluation of visualizations," in *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*, ser. BELIV '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 46–53. [Online]. Available: https://doi.org/10.1145/2669557.2669579

[24] B. Shneiderman and C. Plaisant, "Strategies for evaluating information visualization tools: Multi-dimensional in-depth long-term case studies," in *Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*, ser. BELIV '06. New York, NY, USA: ACM, 2006, pp. 1–7. [Online]. Available: http://doi.acm.org/10.1145/1168149.1168158

[25] R. Arias-Hernandez, L. T. Kaastra, T. M. Green, and B. Fisher, "Pair analytics: Capturing reasoning processes in collaborative visual analytics," in *2011 44th Hawaii International Conference on System Sciences*, Jan 2011, pp. 1–10.

[26] R. Amar and J. Stasko, “Best paper: A knowledge task-based framework for design and evaluation of information visualizations,” in *IEEE Symposium on Information Visualization*, Oct 2004, pp. 143–150.

[27] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI ’90. New York, NY, USA: ACM, 1990, pp. 249–256. [Online]. Available: http://doi.acm.org/10.1145/97243.97281

[28] J. Scholtz, “Developing guidelines for assessing visual analytics environments,” *Information Visualization*, vol. 10, no. 3, pp. 212–231, 2011.

[29] P. Saraiya, C. North, Vy Lam, and K. A. Duca, “An insight-based longitudinal study of visual analytics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1511–1522, Nov 2006.

[30] P. Saraiya, C. North, and K. Duca, “An evaluation of microarray visualization tools for biological insight,” in *IEEE Symposium on Information Visualization*, Oct 2004, pp. 1–8.

[31] R. Chang, C. Ziemkiewicz, T. M. Green, and W. Ribarsky, “Defining insight for visual analytics,” *IEEE Computer Graphics and Applications*, vol. 29, no. 2, pp. 14–17, March 2009.

[32] M. Khayat, M. Karimzadeh, D. S. Ebert, and A. Ghafoor, “The validity, generalizability and feasibility of summative evaluation methods in visual analytics,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019.

[33] M. Smuc, “Just the other side of the coin? from error to insight analysis,” *Information Visualization*, vol. 15, no. 4, pp. 312–324, 2016.

[34] J. Rasmussen, “Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-13, no. 3, pp. 257–266, 1983.

[35] S.-Y. Tan and T. Chan, “Defining and conceptualizing actionable insight: a conceptual framework for decision-centric analytics,” *arXiv preprint arXiv:1606.03510*, 2016.

[36] M. R. Endsley, “Toward a theory of situation awareness in dynamic systems,” in *Situational Awareness*. Routledge, 2017, pp. 9–42.

[37] J. E. McGrath, “Methodology matters: Doing research in the behavioral and social sciences,” in *Readings in Human–Computer Interaction*. Elsevier, 1995, pp. 152–169.

[38] S. Carpendale, “Evaluating information visualizations,” in *Information visualization*. Springer, 2008, pp. 19–45.

[39] A. Crisan and M. Elliott, “How to evaluate an evaluation study? comparing and contrasting practices in vis with those of other disciplines: Position paper,” in *2018 IEEE Evaluation and Beyond-Methodological Approaches for Visualization (BELIV)*. IEEE, 2018, pp. 28–36.

[40] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, 2009.

[41] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.

[42] M. Chen and H. Jaenicke, "An information-theoretic framework for visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1206–1215, 2010.

[43] M. Chen and A. Golan, "What may visualization processes optimize?" *IEEE transactions on visualization and computer graphics*, vol. 22, no. 12, pp. 2619–2632, 2016.

[44] M. Chen and D. S. Ebert, "An ontological framework for supporting the design and evaluation of visual analytics systems," in *Computer Graphics Forum, to appear*, vol. 38, no. 3, 2019.

[45] K. Andrews, "Evaluation comes in many guises," in *AVI Workshop on BEyond time and errors (BELIV) Position Paper*, 2008, pp. 7–8.

[46] T. Munzner, "A nested process model for visualization design and validation," *IEEE Transactions on Visualization & Computer Graphics*, no. 6, pp. 921–928, 2009.

[47] M. Meyer, M. Sedlmair, and T. Munzner, "The four-level nested model revisited: blocks and guidelines," in *Proceedings of the 2012 BELIV Workshop: Beyond Time and Errors-Novel Evaluation Methods for Visualization*. ACM, 2012, p. 11.

[48] S. McKenna, D. Mazur, J. Agutter, and M. Meyer, "Design activity framework for visualization design," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 2191–2200, 2014.

[49] M. Sedlmair, M. Meyer, and T. Munzner, "Design study methodology: Reflections from the trenches and the stacks," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 18, no. 12, pp. 2431–2440, 2012.

[50] J. C. Scholtz, *User-centered evaluation of visual analytics*, ser. Synthesis digital library of engineering and computer science, 2018.

[51] N. Mahyar, S.-H. Kim, and B. C. Kwon, "Towards a taxonomy for evaluating user engagement in information visualization," in *Workshop on Personal Visualization: Exploring Everyday Life*, 2015.

[52] Y.-a. Kang, C. Görg, and J. Stasko, "Evaluating visual analytics systems for investigative analysis: Deriving design principles from a case study," in *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*. IEEE, 2009, pp. 139–146.

[53] J. Scholtz, "Beyond usability: Evaluation aspects of visual analytic environments," in *Visual Analytics Science and Technology, 2006 IEEE Symposium On*. IEEE, 2006, pp. 145–150.

[54] E. Brunswik, *Perception and the representative design of psychological experiments*. Univ of California Press, 1956.

[55] L. Petrinovich, "Probabilistic functionalism: A conception of research method." *American Psychologist*, vol. 34, no. 5, p. 373, 1979.

[56] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[57] A. H. Murphy, "Skill scores based on the mean square error and their relationships to the correlation coefficient," *Monthly weather review*, vol. 116, no. 12, pp. 2417–2424, 1988.

[58] S. Pajer, M. Streit, T. Torsney-Weir, F. Spechtenhauser, T. Möller, and H. Piringer, "Weightlifter: Visual weight space exploration for multi-criteria decision making," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 611–620, Jan 2017.

[59] C. J. Hursch, K. R. Hammond, and J. L. Hursch, "Some methodological considerations in multiple-cue probability studies." *Psychological review*, vol. 71, no. 1, p. 42, 1964.

[60] L. R. Tucker, "A suggested alternative formulation in the developments by hursch, hammond, and hursch, and by hammond, hursch, and todd." *Psychological review*, vol. 71, no. 6, p. 528, 1964.

[61] T. R. Stewart, "A decomposition of the correlation coefficient and its use in analyzing forecasting skill," *Weather and forecasting*, vol. 5, no. 4, pp. 661–666, 1990.

[62] C. Ziemkiewicz, A. Ottley, R. J. Crouser, K. Chauncey, S. L. Su, and R. Chang, "Understanding visualization by understanding individual users," *IEEE Computer Graphics and Applications*, vol. 32, no. 6, pp. 88–94, 2012.

[63] E. M. Peck, B. F. Yuksel, L. Harrison, A. Ottley, and R. Chang, "Towards a 3-dimensional model of individual cognitive differences: Position paper," in *Proceedings of the 2012 BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, ser. BELIV '12. New York, NY, USA: ACM, 2012, pp. 6:1–6:6. [Online]. Available: http://doi.acm.org/10.1145/2442576.2442582

[64] J. Scholtz, "Beyond usability: Evaluation aspects of visual analytic environments," in *2006 IEEE Symposium On Visual Analytics Science And Technology*, Oct 2006, pp. 145–150.

[65] A. Kirlik and R. Strauss, "Situation awareness as judgment i: Statistical modeling and quantitative measurement," *International Journal of Industrial Ergonomics*, vol. 36, no. 5, pp. 463–474, 2006.

[66] R. Strauss and A. Kirlik, "Situation awareness as judgment ii: Experimental demonstration," *International Journal of Industrial Ergonomics*, vol. 36, no. 5, pp. 475–484, 2006.

[67] S. Miller, A. Kirlik, A. Kosorukoff, and J. Tsai, "Supporting joint human-computer judgment under uncertainty," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 52, no. 4, pp. 408–412, 2008.

[68] L. Jin, Y. Chen, T. Wang, P. Hui, and A. V. Vasilakos, "Understanding user behavior in online social networks: A survey," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 144–150, 2013.

[69] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, “Who is tweeting on twitter: human, bot, or cyborg?” in *Proceedings of the 26th annual computer security applications conference*. ACM, 2010, pp. 21–30.

[70] J. P. Dickerson, V. Kagan, and V. Subrahmanian, “Using sentiment to detect bots on twitter: Are humans more opinionated than bots?” in *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 2014, pp. 620–627.

[71] F. Morstatter, L. Wu, T. H. Nazer, K. M. Carley, and H. Liu, “A new approach to bot detection: striking the balance between precision and recall,” in *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 2016, pp. 533–540.

[72] K. Lee, B. D. Eoff, and J. Caverlee, “Seven months with the devils: A long-term study of content polluters on twitter.” in *ICWSM*, 2011, pp. 185–192.

[73] G. Stringhini, C. Kruegel, and G. Vigna, “Detecting spammers on social networks,” in *Proceedings of the 26th annual computer security applications conference*. ACM, 2010, pp. 1–9.

[74] O. Varol, E. Ferrara, C. A. Davis, F. Menczer, and A. Flammini, “Online human-bot interactions: Detection, estimation, and characterization,” *arXiv preprint arXiv:1703.03107*, 2017.

[75] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer, “Botornot: A system to evaluate social bots,” in *Proceedings of the 25th International Conference Companion on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 273–274.

[76] C. Yang, R. Harkreader, and G. Gu, “Empirical evaluation and new design for fighting evolving twitter spammers,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1280–1293, 2013.

[77] Z. Miller, B. Dickinson, W. Deitrick, W. Hu, and A. H. Wang, “Twitter spammer detection using data stream clustering,” *Information Sciences*, vol. 260, pp. 64–73, 2014.

[78] F. Ahmed and M. Abulaish, “A generic statistical approach for spam detection in online social networks,” *Computer Communications*, vol. 36, no. 10-11, pp. 1120–1129, 2013.

[79] N. Cao, C. Shi, S. Lin, J. Lu, Y.-R. Lin, and C.-Y. Lin, “Targetvue: Visual analysis of anomalous user behaviors in online communication systems,” *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 280–289, 2016.

[80] H. Lin, S. Gao, D. Gotz, F. Du, J. He, and N. Cao, “Rclens: Interactive rare category exploration and identification,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 7, pp. 2223–2237, 2018.

[81] P. Isenberg, T. Zuk, C. Collins, and S. Carpendale, “Grounded evaluation of information visualizations,” in *Proceedings of the 2008 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*. ACM, 2008, p. 6.

[82] M. Taras, "Assessment–summative and formative–some theoretical reflections," *British journal of educational studies*, vol. 53, no. 4, pp. 466–478, 2005.

[83] J. Nielsen, "Heuristic evaluation," *Usability inspection methods*, vol. 17, no. 1, pp. 25–62, 1994.

[84] G. Ellis and A. Dix, "An explorative analysis of user evaluation studies in information visualisation," in *Proceedings of the 2006 AVI workshop on beyond time and errors*, ser. BELIV '06. ACM, 2006, pp. 1–7.

[85] M. Sedlmair, "Design study contributions come in different guises: Seven guiding scenarios," in *Proceedings of the 2016 Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*, 2016, pp. 152–161.

[86] J. J. Van Wijk, "The value of visualization," in *Visualization, 2005. VIS 05. IEEE*. IEEE, 2005, pp. 79–86.

[87] J. Stasko, "Value-driven evaluation of visualizations," in *Proceedings of the 2014 Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*, 2014, pp. 46–53.

[88] P. Saraiya, C. North, and K. Duca, "An insight-based methodology for evaluating bioinformatics visualizations," *IEEE transactions on visualization and computer graphics*, vol. 11, no. 4, pp. 443–456, 2005.

[89] J. Nielsen, *Usability engineering*. Elsevier, 1994.

[90] G. Grinstein, A. Kobsa, C. Plaisant, and J. T. Stasko, "Which comes first, usability or utility?" in *IEEE Visualization, 2003. VIS 2003.*, 2003, pp. 605–606.

[91] N. Andrienko, G. Andrienko, J. M. C. Garcia, and D. Scarlatti, "Analysis of flight variability: a systematic approach," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 54–64, 2019.

[92] G. Y.-Y. Chan, P. Xu, Z. Dai, and L. Ren, "V i b r: Visualizing bipartite relations at scale with the minimum description length principle," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 321–330, 2019.

[93] M. Chen, K. Gaither, N. W. John, and B. McCann, "An information-theoretic approach to the cost-benefit analysis of visualization in virtual environments," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 32–42, 2019.

[94] Y. Zhao, F. Luo, M. Chen, Y. Wang, J. Xia, F. Zhou, Y. Wang, Y. Chen, and W. Chen, "Evaluating multi-dimensional visualizations for understanding fuzzy clusters," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 12–21, 2019.

[95] S. Liu, C. Chen, Y. Lu, F. Ouyang, and B. Wang, "An interactive method to improve crowdsourced annotations," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 235–245, 2019.

[96] M. C. Kaptein, C. Nass, and P. Markopoulos, "Powerful and consistent analysis of likert-type ratingscales," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010, pp. 2391–2394.

[97] R. Pienta, F. Hohman, A. Endert, A. Tamersoy, K. Roundy, C. Gates, S. Navathe, and D. H. Chau, “Vigor: interactive visual exploration of graph query results,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 215–225, 2018.

[98] P.-M. Law, R. C. Basole, and Y. Wu, “Duet: Helping data analysis novices conduct pairwise comparisons by minimal specification,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 427–437, 2019.

[99] L. Wilkinson, “Visualizing big data outliers through distributed aggregation,” *IEEE Transactions on Visualization & Computer Graphics*, no. 1, pp. 1–1, 2018.

[100] C. Xie, W. Xu, and K. Mueller, “A visual analytics framework for the detection of anomalous call stack trees in high performance computing applications,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 215–224, 2019.

[101] J. Wang, L. Gou, H.-W. Shen, and H. Yang, “Dqnviz: A visual analytics approach to understand deep q-networks,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 288–298, 2019.

[102] J. Zhao, M. Glueck, P. Isenberg, F. Chevalier, and A. Khan, “Supporting handoff in asynchronous collaborative sensemaking using knowledge-transfer graphs,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 340–350, 2018.

[103] H. Chung, S. P. Dasari, S. Nandhakumar, and C. Andrews, “Cricto: Supporting sensemaking through crowdsourced information schematization,” in *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 2017, pp. 139–150.

[104] R. Arias-Hernandez, L. Kaastra, T. M. Green, and B. D. Fisher, “Pair analytics: Capturing reasoning processes in collaborative visual analytics,” in *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum fÃ1/4r Informatik, 2011.

[105] B. Shneiderman and C. Plaisant, “Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies,” in *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*. ACM, 2006, pp. 1–7.

[106] M. Stein, H. Janetzko, A. Lamprecht, T. Breitkreutz, P. Zimmermann, B. Goldlücke, T. Schreck, G. Andrienko, M. Grossniklaus, and D. A. Keim, “Bring it to the pitch: Combining video and movement data to enhance team sport analysis,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 13–22, 2018.

[107] H. Wang, Y. Lu, S. T. Shutters, M. Steptoe, F. Wang, S. Landis, and R. Maciejewski, “A visual analytics framework for spatiotemporal trade network analysis,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 331–341, 2019.

[108] D. Orban, D. F. Keefe, A. Biswas, J. Ahrens, and D. Rogers, “Drag and track: A direct manipulation interface for contextualizing data instances within a continuous parameter space,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 256–266, 2019.

[109] J. Scholtz, “Developing guidelines for assessing visual analytics environments,” *Information Visualization*, vol. 10, no. 3, pp. 212–231, 2011.

[110] M. Tory and T. Moller, “Evaluating visualizations: do expert reviews work?” *IEEE computer graphics and applications*, vol. 25, no. 5, pp. 8–11, 2005.

[111] K. Allendoerfer, S. Aluker, G. Panjwani, J. Proctor, D. Sturtz, M. Vukovic, and C. Chen, “Adapting the cognitive walkthrough method to assess the usability of a knowledge domain visualization,” in *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on.* IEEE, 2005, pp. 195–202.

[112] S. Jänicke and D. J. Wrisley, “Interactive visual alignment of medieval text versions,” in *2017 IEEE Conference on Visual Analytics Science and Technology (VAST).* IEEE, 2017, pp. 127–138.

[113] Y. Wu, X. Xie, J. Wang, D. Deng, H. Liang, H. Zhang, S. Cheng, and W. Chen, “Forvizor: Visualizing spatio-temporal team formations in soccer,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 65–75, 2019.

[114] N. Pezzotti, T. Höllt, J. Van Gemert, B. P. Lelieveldt, E. Eisemann, and A. Vilanova, “Deepeyes: Progressive visual analytics for designing deep neural networks,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 98–108, 2018.

[115] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner, and M. Sedlmair, “Comparing visual-interactive labeling with active learning: An experimental study,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 298–308, 2018.

[116] J. W. Creswell and V. L. P. Clark, *Designing and conducting mixed methods research.* Sage publications, 2017.

[117] D. Park, S. Kim, J. Lee, J. Choo, N. Diakopoulos, and N. Elmqvist, “Conceptvector: Text visual analytics via interactive lexicon building using word embedding,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 24, no. 1, pp. 361–370, 2018.

[118] Y. Chen, P. Xu, and L. Ren, “Sequence synopsis: Optimize visual summary of temporal event data,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 45–55, 2018.

[119] P. K. Muthumanickam, K. Vrotsou, A. Nordman, J. Johansson, and M. Cooper, “Identification of temporally varying areas of interest in long-duration eye-tracking data sets,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 87–97, 2019.

[120] H. Strobelt, S. Gehrmann, M. Behrisch, A. Perer, H. Pfister, and A. M. Rush, “S eq 2s eq-v is: A visual debugging tool for sequence-to-sequence models,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 353–363, 2019.

[121] Y. Ming, H. Qu, and E. Bertini, “Rulematrix: Visualizing and understanding classifiers with rules,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 342–352, 2019.

[122] J. Zhang, Y. Wang, P. Molino, L. Li, and D. S. Ebert, “Manifold: A model-agnostic framework for interpretation and diagnosis of machine learning models,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 364–373, 2019.

[123] M. Gelderman, “The relation between user satisfaction, usage of information systems and performance,” *Information & management*, vol. 34, no. 1, pp. 11–18, 1998.

[124] S. Fu, H. Dong, W. Cui, J. Zhao, and H. Qu, “How do ancestral traits shape family trees over generations?” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 205–214, 2018.

[125] X. Zhao, Y. Wu, W. Cui, X. Du, Y. Chen, Y. Wang, D. L. Lee, and H. Qu, “Skylens: Visual analysis of skyline on multi-dimensional data,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 24, no. 1, pp. 246–255, 2018.

[126] P.-M. Law, Z. Liu, S. Malik, and R. C. Basole, “Maqui: Interweaving queries and pattern mining for recursive event sequence exploration,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 396–406, 2019.

[127] D. T. Campbell and J. C. Stanley, “Experimental and quasi-experimental designs for research,” *Handbook of research on teaching*, pp. 171–246, 1963.

[128] Y. S. Lincoln and E. G. Guba, *Naturalistic inquiry*. Sage, 1985, vol. 75.

[129] P. L. Bannerman, “Risk and risk management in software projects: A reassessment,” *Journal of Systems and Software*, vol. 81, no. 12, pp. 2118–2133, 2008.

[130] G. Marczyk, D. DeMatteo, and D. Festinger, *Essentials of research design and methodology.* John Wiley & Sons Inc, 2005.

[131] B. J. Copeland, *The essential turing*. Clarendon Press, 2004.

[132] M. Chen, M. Feixas, I. Viola, A. Bardera, H.-W. Shen, and M. Sbert, *Information theory tools for visualization*. AK Peters/CRC Press, 2016.

[133] M. A. Whiting, J. Haack, and C. Varley, “Creating realistic, scenario-based synthetic data for test and evaluation of information analytics software,” in *Proceedings of the 2008 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*. ACM, 2008, p. 8.

[134] G. Cumming, “The new statistics: Why and how,” *Psychological Science*, vol. 25, no. 1, pp. 7–29, 2014.

[135] P. Saraiya, C. North, V. Lam, and K. A. Duca, “An insight-based longitudinal study of visual analytics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1511–1522, 2006.

[136] M. Smuc, E. Mayr, T. Lammarsch, W. Aigner, S. Miksch, and J. Gärtner, “To score or not to score? tripling insights for participatory design,” *IEEE Computer Graphics and Applications*, vol. 29, no. 3, pp. 29–38, 2009.

[137] A. L. Strauss and J. Corbin, *Basics of qualitative research: grounded theory procedures and techniques*. Sage publications, 1998.

[138] J. W. Creswell, *Qualitative inquiry et research design: choosing among five approaches*. SAGE, 2018.

[139] H. W. Desurvire, "Faster, cheaper!! are usability inspection methods as effective as empirical testing?" 1994.

[140] G. H. Guyatt, A. D. Oxman, G. E. Vist, R. Kunz, Y. Falck-Ytter, P. Alonso-Coello, and H. J. Schünemann, "Grade: an emerging consensus on rating quality of evidence and strength of recommendations," *The British Medical Journal (BMJ)*, vol. 336, no. 7650, pp. 924–926, 2008. [Online]. Available: https://www.bmj.com/content/336/7650/924

[141] H. Rohani and A. K. Roosta, "Calculating total system availability," https://bit.ly/30ZJPXk, 2014, accessed: 2019-07-30.

[142] N. Pezzotti, T. Höllt, J. Van Gemert, B. P. Lelieveldt, E. Eisemann, and A. Vilanova, "Deepeyes: Progressive visual analytics for designing deep neural networks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 98–108, 2017.

[143] Y. Y. Haimes, *Risk modeling, assessment, and management*. Hoboken, NJ: John Wiley & Sons, 2015.

[144] D. Hubbard and D. Evans, "Problems with scoring methods and ordinal scales in risk assessment," *IBM Journal of Research and Development*, vol. 54, no. 3, pp. 2:1–2:10, 2010.

[145] G. K. L. Tam, V. Kothari, and M. Chen, "An analysis of machine- and human-analytics in classification," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 71–80, 2017.

[146] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. New York, NY, USA: Wiley-Interscience, 2006.

[147] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI'95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143. [Online]. Available: http://dl.acm.org/citation.cfm?id=1643031.1643047

[148] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler, *Visual Analytics: Scope and Challenges*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 76–90.

[149] P. Pirolli and S. Card, "The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis," in *Proceedings of international conference on intelligence analysis*, vol. 5. McLean, VA, USA, 2005, pp. 2–4.

[150] D. Cashman, A. Perer, R. Chang, and H. Strobelt, "Ablate, variate, and contemplate: Visual analytics for discovering neural architectures," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 863–873, 2020.

[151] L. Battle and J. Heer, "Characterizing exploratory visual analysis: A literature review and evaluation of analytic provenance in tableau," *Computer Graphics Forum*, vol. 38, no. 3, pp. 145–159, 2019.

[152] M. Khayat, M. Karimzadeh, J. Zhao, and D. S. Ebert, “Vassl: A visual analytics toolkit for social spambot labeling,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019.

[153] T. R. Stewart, K. F. Heideman, W. R. Moninger, and P. Reagan-Cirincione, “Effects of improved information on the components of skill in weather forecasting,” *Organizational behavior and human decision processes*, vol. 53, no. 2, pp. 107–134, 1992.

[154] M. A. Whiting, J. Haack, and C. Varley, “Creating realistic, scenario-based synthetic data for test and evaluation of information analytics software,” in *Proceedings of the 2008 Workshop on BEyond Time and Errors: Novel evaLuation Methods for Information Visualization*, ser. BELIV ’08. ACM, 2008, pp. 8:1–8:9.

[155] K. Cook, G. Grinstein, and M. Whiting, “The vast challenge: history, scope, and outcomes: An introduction to the special issue,” *Information Visualization*, vol. 13, no. 4, pp. 301–312, 2014.

[156] J. Gerken, P. Bak, and H. Reiterer, “Longitudinal evaluation methods in human-computer studies and visual analytics,” in *InfoVis 2007 : Workshop on Metrics for the Evaluation of Visual Analytics, Sacramento, CA, 2007*, 2007.

[157] E. Zgraggen, A. Galakatos, A. Crotty, J. Fekete, and T. Kraska, “How progressive visualizations affect exploratory analysis,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 8, pp. 1977–1987, 2017.

[158] J. W. Schooler, S. Ohlsson, and K. Brooks, “Thoughts beyond words: When language overshadows insight.” *Journal of experimental psychology: General*, vol. 122, no. 2, p. 166, 1993.

[159] S. S. Stevens *et al.*, “On the theory of scales of measurement,” 1946.

[160] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[161] L. Yu and H. Liu, “Efficient feature selection via analysis of relevance and redundancy,” *Journal of machine learning research*, vol. 5, no. Oct, pp. 1205–1224, 2004.

[162] Rui Xu and D. Wunsch, “Survey of clustering algorithms,” *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

[163] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, “The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race,” in *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 2017, pp. 963–972.

[164] J. Zhang, R. Zhang, Y. Zhang, and G. Yan, “The rise of social botnets: Attacks and countermeasures,” *IEEE Transactions on Dependable and Secure Computing*, 2016.

[165] B. Viswanath, M. A. Bashir, M. B. Zafar, S. Bouget, S. Guha, K. P. Gummadi, A. Kate, and A. Mislove, “Strength in numbers: Robust tamper detection in crowd computations,” in *Proceedings of the 2015 ACM on Conference on Online Social Networks*. ACM, 2015, pp. 113–124.

[166] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Social fingerprinting: detection of spambot groups through dna-inspired behavioral modeling," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 561–576, 2018.

[167] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *The Craft of Information Visualization*. Elsevier, 2003, pp. 364–371.

[168] U.S. Department of Health and Human Services, *The research-based web design & usability guidelines*. Washington: U.S. Government Printing Office, 2006.

[169] C. Seifert, B. Kump, W. Kienreich, G. Granitzer, and M. Granitzer, "On the beauty and usability of tag clouds," in *2008 12th International Conference Information Visualisation*. IEEE, 2008, pp. 17–25.

[170] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[171] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the ACL*, 2004.

[172] B. Schölkopf, A. J. Smola, and K.-R. Müller, "Kernel principal component analysis, advances in kernel methods: support vector learning," 1999.

[173] G. McLachlan, *Discriminant analysis and statistical pattern recognition*. John Wiley & Sons, 2004, vol. 544.

[174] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[175] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[176] D. F. Williamson, R. A. Parker, and J. S. Kendrick, "The box plot: a simple visual method to interpret data," *Annals of Internal Medicine*, vol. 110, no. 11, pp. 916–921, 1989.

[177] S. van den Elzen and J. J. van Wijk, "Small multiples, large singles: A new approach for visual data exploration," in *Computer Graphics Forum*, vol. 32, no. 3pt2. Wiley Online Library, 2013, pp. 191–200.

[178] M. Sedlmair, T. Munzner, and M. Tory, "Empirical guidance on scatterplot and dimension reduction technique choices," *IEEE transactions on visualization and computer graphics*, vol. 19, no. 12, pp. 2634–2643, 2013.

[179] J. L. Hintze and R. D. Nelson, "Violin plots: A box plot-density trace synergism," *The American Statistician*, vol. 52, no. 2, pp. 181–184, 1998.

[180] M. Harrower and C. A. Brewer, "Colorbrewer.org: An online tool for selecting colour schemes for maps," *The Cartographic Journal*, vol. 40, no. 1, pp. 27–37, 2003.

[181] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS quarterly*, pp. 319–340, 1989.

[182] Y. Wu, N. Cao, D. Gotz, Y.-P. Tan, and D. A. Keim, "A survey on visual analytics of social media data," *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2135–2148, 2016.

[183] S. Chen, L. Lin, and X. Yuan, "Social media visual analytics," in *Computer Graphics Forum*, vol. 36, no. 3. Wiley Online Library, 2017, pp. 563–587.

[184] J. Zhang, R. Zhang, Y. Zhang, and G. Yan, "The rise of social botnets: Attacks and countermeasures," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 6, pp. 1068–1082, 2018.

[185] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race," in *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 2017, p. 963–972.

[186] G. C. Cawley and N. L. Talbot, "Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers," *Pattern Recognition*, vol. 36, no. 11, pp. 2585 – 2592, 2003.

[187] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[188] J. S. Yi, Y.-a. Kang, J. T. Stasko, and J. A. Jacko, "Understanding and characterizing insights: How do people gain insights using information visualization?" in *Proceedings of the 2008 Workshop on BEyond Time and Errors: Novel evaLuation Methods for Information Visualization*, ser. BELIV '08, 2008, pp. 4:1–4:6.

[189] H. Guo, S. R. Gomez, C. Ziemkiewicz, and D. H. Laidlaw, "A case study using visualization interaction logs and insight metrics to understand how analysts arrive at insights," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 51–60, 2016.

[190] K. Reda, A. E. Johnson, J. Leigh, and M. E. Papka, "Evaluating user behavior and strategy during visual exploration," in *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*, ser. BELIV '14, 2014, pp. 41–45.

[191] G. Bernot, M.-C. Gaudel, and B. Marre, "Software testing based on formal specifications: a theory and a tool," *Software Engineering Journal*, vol. 6, no. 6, pp. 387–405, 1991.

VITA

VITA

Mosab Khayat received his Bachelor degree in Computer Engineering from the Umm Al-Qura University, Makkah, Saudi Arabia in 2011. He received his Master's in Computer Engineering from the School of Electrical and Computer Engineering at Purdue University, West Lafayette, IN, USA in 2015. His research interests include modeling and evaluation of visual analytics systems, visualization, and machine learning. Contact him at: mkhayat@purdue.edu or maakhayat@uqu.edu.sa.