FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA

Domain Knowledge-based Visualization Recommendation System

Dissertation
zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)

vorgelegt dem Rat der Fakultät für Mathematik und Informatik
der Friedrich-Schiller-Universität Jena

von Pawandeep Kaur
geboren am 22.08.1986 in Jalandhar, Indien

Gutachter

1. Prof. Dr. Birgitta König-Ries, Friedrich-Schiller-Universität Jena, 07743 Jena, Germany
2. Prof. Dr. Haim Levkowitz, University of Massachusetts Lowell, Lowell, MA 01854, United States
3. Dr. Stefan Jänicke, University of Southern Denmark, 5230 Odense, Denmark

Tag der öffentlichen Verteidigung: 14. July 2021

Ehrenwörtliche Erklärung

Hiermit erkläre ich,

- dass mir die Promotionsordnung der Fakultät bekannt ist,
- dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte oder Ergebnisse eines Dritten oder eigenen Prüfungsarbeiten ohne Kennzeichnung übernommen und alle von mir benutzten Hilfsmittel, persönliche Mitteilungen und Quellen in meiner Arbeit angegeben habe,
- dass ich die Hilfe eines Promotionsberaters nicht in Anspruch genommen habe und daß Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe.

Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts haben mich folgende Personen unterstützt:

- Prof. Dr. Birgitta König-Ries

Ich habe die gleiche, eine in wesentlichen Teilen ähnliche bzw. eine andere Abhandlung bereits bei einer anderen Hochschule als Dissertation eingereicht: Ja / [✓ Nein].

Jena, den ...14.07.2021....

Unterschrift

[Pawandeep Kaur]

To my parents and my beloved husband

Acknowledgement

Teachers in India are described as Guru. So, my first and foremost gratitude goes to my Ph.D. Guru, my Professor, Prof. Dr. Birgitta König Ries. She offered me this position, provided an excellent research environment, was always there for any support, and showed tremendous faith in my work abilities. I am highly thankful for her guidance and encouragement. I want to thank my external reviewers Prof. Dr. Haim Levkowitz and Dr. Stefan Jänicke, for their valuable comments on this thesis. Then a big thanks to all the internal reviewers of this thesis. I owe a special thanks to Dora Kiesel from the Bauhaus University, who collaborated to develop the biodiversity visualization text classifiers. A great thanks to Dr. Alsayed Algergawy for his guidance in improving the context-aware variable selection algorithm. Without the support of the thesis evaluators and study participants, I would not have accomplished this work. I want to pay my special thanks to them and other biodiversity members I met at the conferences and workshops.

A heartfelt thanks to the Biodiversity Exploratories project for their funding. From the day I arrived in Jena, my colleagues from the BE BExIS team had helped me with different tasks to make my work more convenient. I am highly thankful to them, especially Andreas Ostrowski, Eleonora Petzold, and ex-team member Michael Owonibi. Immense praise and thanks to my multi-cultural fusion team, which had helped me with my work by providing feedback and ideas and had made my work-life more pleasant. A special thanks to the contribution of all my student helpers in this thesis. I feel grateful to all the experts with whom I shared ideas at different conferences and business travels.

I want to thank my wonderful and supportive family and friends: my father, who gave me wings to fly to choose my life directions and has always encouraged me to do my best. My brother Amritpal Singh, who taught me to use our first computer back in 2001 and from whom I get inspired to choose informatics in my bachelor studies. My late mother, who wanted me to be a doctor. I know, though, she wanted it to be a medical doctor but hey mamma! I am now a Ph.D. doctor. I hope somewhere from the sky-high you are smiling at me with pride. Thanks and praise to my dearest husband Boris, who has loved me and supported me in all my decisions ever since we met in 2015. My supportive parents-in-law for their love and extensive child care to my kids so that I can complete my thesis. Thanks to all my kids' nannies for their valuable assistance. Thanks to all my friends for just being there for me.

My deepest gratitude to my spiritual Guru for providing me wisdom and courage to do my work honestly and without caring about any obstacles.

I know I might have missed acknowledging some names at this point, but I am heartily thankful to them for all their big or small contributions to this work. Last but not least, I pay my thanks to this Jena city for: a wonderful husband, two beautiful kids, a loving family, great colleagues and friends, a doctorate title, and I hope this list continues...

Abstract

Studies have long advocated the inclusion of domain knowledge for producing an effective visualization system. The insights and reasoning artifacts gained from these systems are closer to the knowledge of the domain users and the data context. However, most existing knowledge-based visualization applications focus on integrating domain knowledge tailored only for the specific analytic task. Visualization recommendation systems are those systems that provide different insights into the dataset by automatically selecting different views or visualizations of the dataset. Previous work relating to the development of visualization recommendation systems suggest visualizations based on different parameters: visual mapping of data attributes, preselection of user tasks and mapping accordingly, deviation based theory, machine trained visuals to data encoding schemes, ontology mapping, etc. However, there are limited studies that have tried to include domain knowledge as the visualization selection criteria. In developing a visualization recommendation system where the ultimate goal is not to answer any specific question but to explore the dataset's multidimensional insights, the inclusion of domain knowledge is not common. Thus, though we know that domain knowledge could be a pivotal ingredient to increase the visualization interpretation, how such knowledge can be included in a visualization recommendation system has not yet been sufficiently explored.

In this thesis, we have explored how domain knowledge can be integrated into various stages of visualization recommendation systems. As a result of that work, we have developed a novel domain knowledge-based visualization recommendation system. We have used biodiversity research as our application domain. The contributions of this thesis are: 1) The domain knowledge-based visualization recommendation model. 2) A system for automatic runtime generation of visual goals. We have developed the first visualization text classifier that suggests visualizations by processing a domain-specific text. Using our visualization taxonomy, this classifier then functions on the provided data's metadata text to generate visual goals (Distribution, Network, Composition, Trend, Comparison, Overview). This classifier further contributes to a novel machine learning-based technique of gathering domain knowledge from visualization image captions in the literature. Moreover, we developed the very first visualization or chart type classifier based on textual data. 3) Finally, in this work, we designed a context-aware variable selection algorithm that automatically selects the most relevant variable set to visualize a high-dimensional dataset.

Zusammenfassung

Studien befürworten seit langem die Einbeziehung von Domänenwissen bei der Erstellung eines effektiven Visualisierungssystems. Die aus solchen Systemen gewonnenen Erkenntnisse und Argumentationsartefakte sind näher am Wissens- und Datenkontext der fachspezifischen Nutzer. Die meisten vorhandenen wissensbasierten Visualisierungsanwendungen konzentrieren sich jedoch auf die Integration von Domänenwissen, das lediglich auf die spezielle Analyseaufgabe zugeschnitten ist. Bei der Entwicklung eines Systems zur Empfehlung von Visualisierungen, bei dem das ultimative Ziel nicht darin besteht, eine bestimmte Frage zu beantworten, sondern die mehrdimensionalen Erkenntnisse des Datensatzes zu untersuchen, ist die Einbeziehung von Domänenwissen eher unbekannt. Obwohl wir wissen, dass Domänenwissen ein entscheidender Faktor zur verbesserten Interpretation einer Visualisierung sein könnte, ist nicht bekannt, wie dieses Wissen in das Visualisierungsempfehlungssystem aufgenommen werden kann. Visualisierungsempfehlungssysteme sind Systeme die unterschiedliche Einsichten in den Datensatz bieten, indem sie automatisch verschiedene Ansichten oder Visualisierungen des Datensatzes auswählen. In früheren Arbeiten hinsichtlich Systemen zur Empfehlung von Visualisierungen basieren die Vorschlägen für Visualisierungen auf verschiedenen Parametern: visuelle Zuordnung von Datenattributen, Vorauswahl der Benutzeraufgaben und entsprechende Zuordnung, abweichungsbasierte Theorie, maschinell trainierte Visualisierungen zu Datencodierungsschemata, Ontologie basierte Zuordnung usw. Es gibt jedoch nur wenige Studien, die versucht haben Domänenwissen als Auswahlkriterium für Visualisierungen einzubeziehen.

In dieser Arbeit haben wir untersucht, wie Domänenwissen in verschiedene Phasen der Visualisierungsempfehlungssysteme integriert werden kann. Hieraus resultierend haben wir das allererste Visualisierungsempfehlungssystem das auf Domänenwissen basiert bereitgestellt. Als Anwendungsbereich haben wir das Forschungsfeld der Biodiversität verwendet. Die Beiträge dieser Arbeit sind: 1) Das allererste Visualisierungsempfehlungsmodell das auf Domänenwissen basiert. 2) Ein System zur automatischen Erzeugung des Visualisierungsziels während der Laufzeit. Wir haben die ersten auf Domänenwissen basierenden Visualisierungstextklassifizierer entwickelt, die Visualisierungen aus dem Text vorhersagen. Mithilfe unserer Visualisierungstaxonomie arbeiten diese Klassifizierer dann auf dem Metadatentext der bereitgestellten Daten um visuelle Ziele zu generieren. Diese Klassifikatoren tragen ferner dazu bei, eine neuartige, auf maschinellem Lernen basierende Technik zum Sammeln von Domänenwissen aus Bildunterschriften von Visualisierungen in der Literatur zu sein. Darüber hinaus allererste Visualisierungs- oder Diagrammtypklassifizierer basierend auf den Textdaten. 3) Der kontextsensitive Algorithmus zur Variablenselektion, der automatisch einen prominenten Variablensatz auswählt um den hochdimensionalen Datensatz zu visualisieren.

Contents

List of Figures

List of Tables

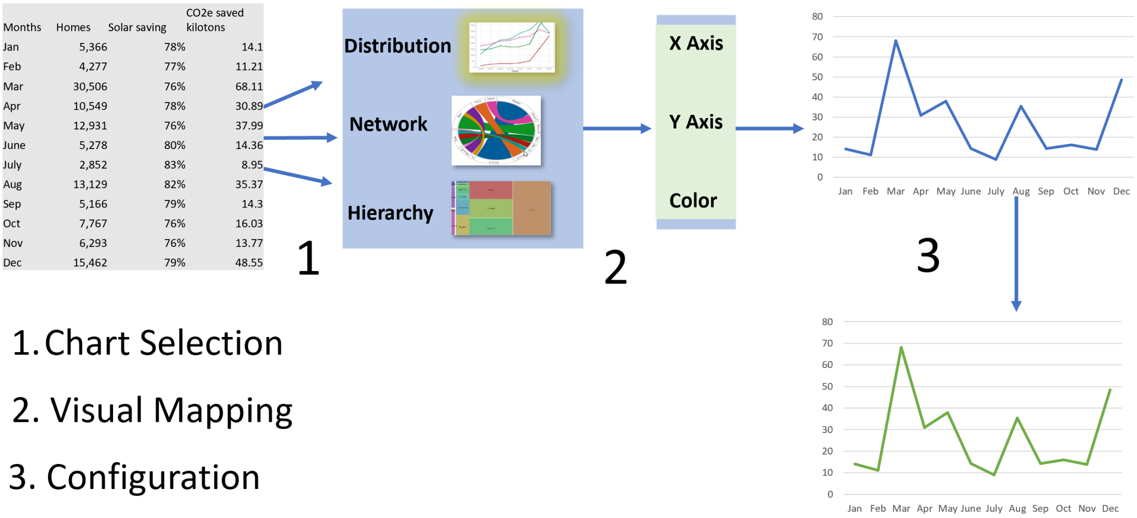
Chapter 1

Introduction

The human brain can comprehend images a lot easier than words or numbers. This aptness is due to our cognition's ability to detect patterns, anomalies, textures or distances in graphics. Data graphics or visualizations summarize data and present the most relevant information in an easy-to-understand form. This makes data visualization an essential tool in exploring, analyzing, and presenting both the obvious and less obvious data features. The increasing awareness of the importance of visualization and the vast diversity in types of data visualized have led to the generation of a plethora of visualization classes. Given these many visualization classes or types, and the various ways each class shows a particular aspect of the data, and ever-increasing visualization applications (for a data science domain, it is result presentation, data quality or trend analysis), individuals are increasingly faced with the difficulty of deciding which visualization is most appropriate for their task.

1.1 Visualization Selection Problem

As mentioned by [Bertin, 1983], *"A hundred different graphics for the same information"*. If we say that this number has increased to thousands after nearly four decades, it would not be an overstatement. Due to the unlimited choice of graphics or visualizations available, a visualization process is considered as a search process [Chen et al., 2008]. In the construction of visualization, one needs to make various nuanced judgments, e.g., selecting an appropriate visualization tool, the type of chart, type of variables and color scheme. This process is iterative, which ends when one produces satisfactory results. In Figure 1, we show a workflow of a manual visualization creation process on a tabular dataset.



Months	Homes	Solar saving	CO2e saved kilotons
Jan	5,366	78%	14.1
Feb	4,277	77%	11.21
Mar	30,506	76%	68.11
Apr	10,549	78%	30.89
May	12,931	76%	37.99
June	5,278	80%	14.36
July	2,852	83%	8.95
Aug	13,129	82%	35.37
Sep	5,166	79%	14.3
Oct	7,767	76%	16.03
Nov	6,293	76%	13.77
Dec	15,462	79%	48.55

Figure 1: Manual visualization creation process.

To understand this visualization creation process, we first need to know the terminologies related to it, which are described in the next section.

1.1.1 Basic concepts and terminologies

- **Data visualization:** Colin Ware [Ware, 2012] defines visualization as, *"a graphical representation of data or concepts,"* which is either an *"internal construct of the mind"* or an *"external artifact supporting decision making."* In other words, visualizations assist humans with data analysis by representing information visually. Traditionally visualization is grouped into two major areas: *Scientific Visualization*, which involves scientific data with an inherent physical component, and *Information Visualization*, which involves abstract and non-spatial data [Tory and Moller, 2004]. Both of them create graphical models and visual representations from data that support direct user interaction to explore and acquire insight into useful information embedded in the underlying data [Ferreira de Oliveira and Levkowitz, 2003].

  According to [Mennis et al., 2000], data is observational measurements that have been recorded in some way, whereas information is generalized data, ordered, and contextualized in a meaningful way. The information thus is selective towards data, and it separates the important from the relatively unimportant. Therefore, data visualizations are the ones which provide an insight into the observational data that is not yet information or not yet contextualized. In this work, we propose a solution for one specific domain's observational data. Therefore, throughout this thesis, a reference to a keyword visualization is meant for data visualization. Reference to the data is for observational data. The terms visualizations, charts, graphs, plots and graphics are used synonymously in many places.

- **Visual goals:** Visualizations can be classified by their representational goals or tasks. Different authors have provided different views on the organization of these tasks. A useful reference can be found in the InfoVis wiki[1]. In this thesis, we have considered two broad categories of these tasks, i.e., high-level tasks

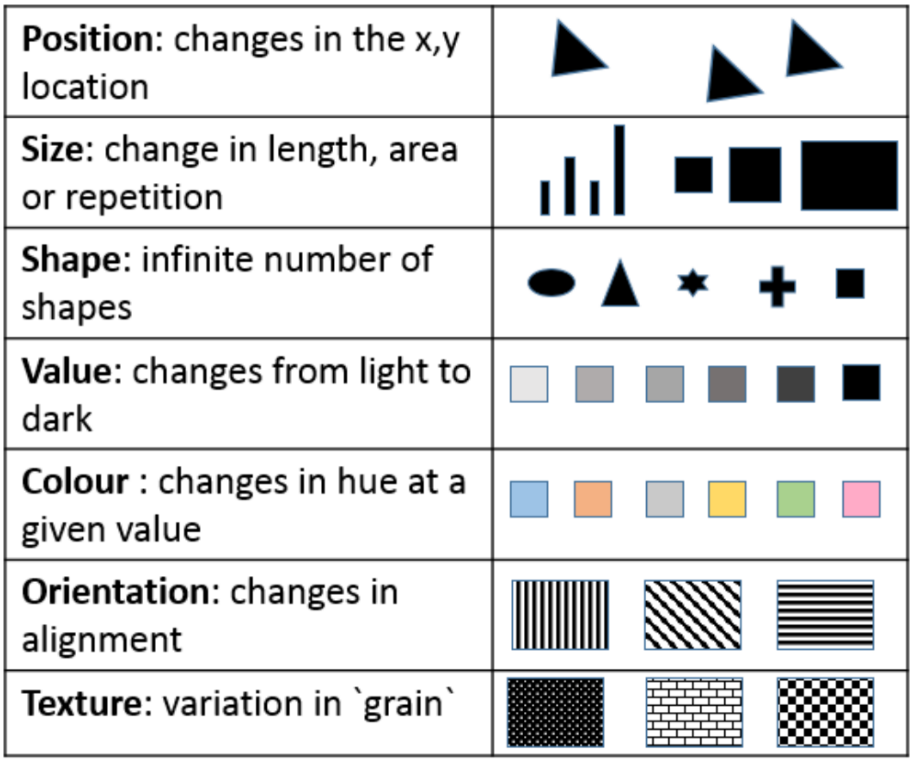
[1] www.infovis-wiki.net/wiki/Task

and low-level tasks. High-level tasks [Schulz et al., 2013] define a relationship between the chart variables. For example, scatterplots are relevant for representing ‘correlation’ and ‘distribution’. As coined by [Amar and Stasko, 2004], low-level tasks manipulate the chart and the data and thus aid in representing high-level tasks. For example, data transformation or data filtering.

In this thesis, both goals and tasks have the same meaning. A reference to visual goals or tasks is meant for high-level tasks.

- **Visual Marks:** Jaques Bertin [Bertin, 1983] has argued that visual marks are the basic visualization units that visually differentiate one graphical object from another. He developed methods through which these units can be modified, including position, size, shape, or color. These predefined modifications are called visual variables. He defined seven visual variables as shown in Figure 2. Visual variables are also called visual attributes, visual marks, visual components, and visual elements. We have used these terms interchangeably throughout the thesis.

- **Visualization types:** We have used visualization types, visualization techniques, visualization classes, and chart types synonymically in this thesis. All these terms differentiate one type of visualization from another. For example, scatterplots and line charts are two different visualization types.



Position: changes in the x,y location	▲ ▲ ▲
Size: change in length, area or repetition	▮ ■ ■ ■
Shape: infinite number of shapes	⬬ ▲ ✶ ✚ ■
Value: changes from light to dark	■ ■ ■ ■ ■ ■
Colour : changes in hue at a given value	🟦 🟧 ⬜ 🟨 🟩 🟪
Orientation: changes in alignment	▥ ▧ ▤
Texture: variation in `grain`	▩ ▦ ▩

Figure 2: Bertin’s Visual Variables. Adapted from Bertin’s Original Visual Variables [Bertin, 1983].

Once we know these basic visualization terminologies, we can understand the manual visualization creation process depicted in Figure 1.

1. **Chart selection:** Chart type or visualization type selection is crucial in a visualization creation process. It is directly based on the user's goal of the analysis. What does a user want to show or want to see from the visualization? Each visualization has its own set of representative goals [Harris, 2000]. For example, if a user is interested in data distribution, then the appropriate visualizations are line charts, scatterplots or other distribution based charts. In case the user wants to show the network or connection between variables, then appropriate visualizations: node-link diagrams, chord diagrams.

2. **Visual mapping:** InfoVis wiki[2] defines visual mapping as a mapping between data aspects and visual variables, i.e., assigning specific data attributes to visual characteristics to facilitate visual sense-making. Once the type of visualization is determined, the second step is to map the data variables to the chart-specific visual marks. The mapping of the data to the visual variables is done based on some existing classification schemes. For example, the one used by Tableau [Mackinlay, 1986] is presented in Table 1.1. Here, based on its data attribute, the dataset variables are mapped to the visual marks. Here, 'C' is categorical, 'Qi' is quantitative interval, 'Qd' is quantitative discrete, and 'Cdate' is categorical date data attributes.

Table 1.1: Tableau Visual Mapping Rules [Mackinlay, 1986]

Pane Type (Field 1)	**Pane Type (Field 2)**	**Mark Type**	**View Type**
C	C	Text	Cross-tab
Qd	C	Bar	Bar view
Qd	Cdate	Line	Line view
Qd	Qd	Shape	Scatterplot
Qi	C	Gantt	Gantt view
Qi	Qd	Line	Line view
Qi	Qi	Shape	Scatter plot

3. **Chart configuration:** Once the chart is developed, it is important to configure it in a presentable form. A chart needs to be configured with proper scales, color scheme, chart size and legends to make it more understandable and interpretable.

As we can see, even creating a single visualization involves complex decision-making steps. The mismatch of any of these elements leads to misinterpreted charts. If these elements are ignored, people might interpret the data unintendedly or not understand the underlying information [Kulyk et al., 2007]. Studies have criticized the visualizations in scientific articles due to many of the following quality issues: inadequate, missing, or contradictory explanation or labeling, visual clutter and distortion, extraneous and unnecessary decoration, non-standard graphic conventions, inappropriate selection of representations (e.g., simple univariate displays when multivariate displays were needed) [Cooper et al., 2002, Schriger et al., 2006, Dasgupta et al., 2017]. Previous usability studies [Dasgupta et al., 2017], investigated the reasons behind these visualization usage inadequacies and found that users (especially scientists) lack trust in cutting-edge tools as opposed to conventional analysis mediums. They often use their own analysis and visualization techniques thus

[2] `www.infovis-wiki.net/wiki/Visual_Mapping`

do not consider the spectrum of available chart types. Due to the lack of automated assistance for visualization creation and tremendous efforts, time and resources need to learn a new tool; a non-visualization expert would avoid using new technologies. To decrease these usability barriers, studies [Wongsuphasawat et al., 2015, Parameswaran et al., 2013, Vartak et al., 2014] have advocated to use automation at different levels of the visualization creation process. This further opens the doorway to the research in visualization recommendation systems.

1.2 Visualization Recommendation Systems

A Visualization Recommendation System is described by [Vartak et al., 2017], as one *"that automatically recommend visualizations that highlight patterns or trends of interest, thus enabling fast visual analysis"*. According to [Hu et al., 2019], *"Visualization recommender systems aim to lower the barrier to exploring basic visualizations by automatically generating results for analysts to search and select, rather than manually specify"*.

Visualization recommendation systems automatically construct visualizations and show various data insights visually. These visualization recommendations are based on the different data aspects and users' analytical goals. These aspects were thoroughly investigated and are presented in Chapter 3.

1.3 Domain of Application: Biodiversity Research

Biodiversity research understands the enormous diversity of life on earth and identifies the factors and interactions that generate and maintain this diversity. Biodiversity data is the data accumulated from research done by biologists and ecologists on different taxa and levels, land use, and ecosystem processes. For proper preservation, reusability, and sharing of such data, metadata is provided along with the data. This metadata contains vital contextual information related to the datasets like the purpose of the research work, data collection method, and other important keywords. In order to answer the most relevant questions of biodiversity research, synthesis of data stemming from the integration of datasets from different experiments or observation series is frequently needed. Collaborative projects thus tend to enforce centralized data management. This is true, e.g., for the Biodiversity Exploratories (BE) [Fischer et al., 2010], a large-scale, long-term project funded by Deutsche Forschungsgemeinschaft (DFG). The Exploratories use the BExIS platform (Biodiversity Exploratories Information System) [Lotz et al., 2012] for central data management. The instance of BExIS[3] used within the Biodiversity Exploratories serves as one of the primary sources for collecting this study's requirements. The large collection of data available in the BE instance of BExIS results from research activities by many disciplines involved in biodiversity science for many years.

[3] www.bexis.uni-jena.de

1.4 Motivation

Most of the BE data is observational data, which is in a raw/unprocessed form. This data is highly complex, heterogeneous, and often not easy to understand. To explore, interpret, present, and reuse such data, a system is required to visualize these datasets effectively. Providing a workflow to explore such data is essential for scientists to decide if such data fit their hypothesis and is relevant for their work. As stated by [Boyle et al., 1993], the benefits of such visual exploration tools at the data management level support the continual search of the data without transferring it from one tool to another.

Visualization recommendation tools at the database level assist in getting data insights and help with a visualization selection dilemma. The visualization selection dilemma happens when a user cannot find the relevant visualization techniques or types to represent their data. Nowadays, with ample visualizations available, an appropriate visualization selection can become challenging for a visualization layman [Kaur et al., 2018].

Furthermore, matters related to visualization are made even more complicated by human perception subjectivity [Rui et al., 1998], which means people perceive the same thing differently under different circumstances. For better understanding, readers primarily need to relate the visualizations to the realm of their existing knowledge domain [Amar and Stasko, 2004]. To ensure that the chosen visualization does indeed convey the intended message to the target readers, a visualization model should integrate the domain knowledge and the context of the data at the different levels of the visualization design process. Studies have long advocated the inclusion of domain knowledge for producing an effective visualization system. The insights and reasoning artifacts gained from these systems are closer to the knowledge of the domain user and the data context. However, most existing knowledge-based visualization applications focus on integrating domain knowledge tailored exclusively for the specific analytic task [Federico et al., 2017, Wagner et al., 2017, Wagner et al., 2018].

1.5 Problem Specification

In developing a visualization recommendation system where the ultimate goal is not to answer any specific question but to explore the dataset's multidimensional insights, the inclusion of domain knowledge is not common. Though we know that domain knowledge could be a pivotal ingredient to increase the visualization interpretation, how such knowledge can be included in the visualization recommendation system has not yet been sufficiently explored. This thesis explores and answers this question by using biodiversity as our application domain. In the earlier days of our research, we could not find studies understanding the visualization requirements for this community. Consequently, we conducted several surveys, meetings and interviews to know their requirements. Therefore, this thesis also contributes in understanding the biodiversity community's visualization requirements and wishes. Our intensive visualization usability study is reported in Chapter 2. Based on their feedback, we investigated the scientific literature to understand the current state-of-the-art (Chapter 3). Analyzing both these aspects, we observed that the visualization science still lacks studies related to domain-based recommendation systems. Due to insufficient research on this subject, there are no clear directions on how the domain knowledge for such visualization systems can be gathered and integrated. In the coming chapters, we have provided a detailed investigation into this problem and our solution.

1.6 Thesis Structure

This thesis is structured into the following chapters:

- *Chapter 1* provides a brief introduction to the visualization and biodiversity domain and the motivation for this research.
- *Chapter 2* presents details about the visualization requirements survey which we had conducted to gather our community's visualization requirements.
- *Chapter 3* presents the current state-of-the-art in the context of visualization tools for the biodiversity community and visualization recommendation techniques.
- Based on the identified issues and requirements from the previous chapters, *Chapter 4* lays down the core requirements fulfilled by our research and the contributions of this thesis.
- *Chapter 5* introduces the first contribution of our research work. It presents our Domain Knowledge-based Visualization Recommendation Model. We describe in detail the role of different components of this model.
- *Chapter 6* describes the construction of the Biodiversity Visualization Text Classifier. It details the necessary procedures we had adopted to obtain the required data, the classification process, the enhancements, results, and the comparison with other studies.
- *Chapter 7* presents our visualization taxonomy. It also explains the workflow to generate the visual goals based on the predicted visualization list from our visualization classifier.
- *Chapter 8* presents our Context-aware Variable Selection Algorithm. It presents the motivation of this work and the workflow we have adopted to construct this algorithm.
- *Chapter 9* presents the visualization tool that we have developed as a result of our research. It provides the technical details on the construction of our visualization system.
- *Chapter 10* presents the results of the quantitative and qualitative evaluations conducted to evaluate the Biodiversity Visualization Text Classifier, Context-aware Variable Selection Algorithm, and the overall system.
- In *Chapter 11*, we summarize the results of this thesis and conclude by presenting some open issues and future research directions.

1.7 Publications

- Kaur, P., Owonibi, M., & Koenig-Ries, B. (2015, May). Towards Visualization Recommendation-A Semi-Automated Domain-Specific Learning Approach. *In Proceedings of the 27th GI-Workshop Grundlagen von Datenbanken.* 1366. 30–35. Magdeburg, Germany.

  It is one of our first publications related to this work. It highlights our motivation and the vision of creating a visualization recommendation system that is tightly integrated with the data domain. The details of which have been partially discussed in this chapter.

- Kaur, P., & Owonibi, M. (2017, February). A review on visualization recommendation strategies. *In Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications.* 3. 266-273. Porto, Portugal.

  This paper is a result of our study of the state-of-the-art on visualization recommendation techniques. It distinguished these techniques into several groups based on different aspects as detailed in Chapter 3.

- Kaur, P., Gaikwad, J., & König-Ries, B. (2016). Towards recommending visualizations for biodiversity data. *Biodiversity and conservation.* 25(9). 1801-1803.

  This paper discusses the usefulness of our solution to the biodiversity community. Through this paper, we appealed to the community to participate in our online visualization requirement survey. Based on this, we have created our user-centric visualization system. The results from this survey are also discussed in Chapter 2.

- Kaur, P., Klan, F., & König-Ries, B. (2018, June). Issues and Suggestions for the Development of a Biodiversity Data Visualization Support Tool. *In Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers.* 73-77. Brno, Czech Republic.

  This paper presents results from our visualization requirement survey, which was distributed online and offline among biodiversity researchers. The problems identified from this survey and the gathered feedback have served as an integral part of creating our visualization solution. This paper has also contributed to the writing of Chapter 2.

- Kaur, P., & König-Ries, B. (2017, June). Visualization Taxonomy based on the Specification of User's Goal and Data Dimensions. *In Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Posters.* 29-31. Barcelona.

  This paper introduces our visualization taxonomy based on the user goals and the data dimensions. For the construction of this taxonomy, we gathered some high-level goals and assigned different visualizations to them. The detail of this work is provided in Chapter 7.

- Kaur, P., & Kiesel, D. (2020, February). Combining Image and Caption Analysis for Classifying Charts in Biodiversity Texts. *In Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications.* 3. 157-168. Valletta, Malta.

  This paper provides the methods we have adopted to create the first text-based visualization classifier. This classifier was produced by processing a large set of visualization captions from the biodiversity publications. This paper has also contributed to the writing of Chapter 6.

Chapter 2

Requirement Analysis

**Part of this chapter is based on work published in the Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: EuroVis'18, 73–77, Brno, Czech Republic*

In the previous chapter, we have briefly introduced the problem of visualization selection and other complexities involved in creating a visualization. Due to which there is limited usability of novel visualization technologies among the non-visualization community. To support such a community, we have provided a vision to create a domain knowledge integrated visualization recommendation solution. However, before developing software for a specific community, it is important to perceive the usefulness of the new system for them [Marangunić and Granić, 2015]. Such a software design process where all stakeholders (e.g., partners, customers, end-users) are actively involved is known as participatory design process (sometimes also referred to as co-operative design process) [Hinrichs et al., 2017]. The participatory design ensures that the resulting technical solutions meet all needs, that the final systems are usable, and that it can be easily integrated into existing workflows of the end-users [Jänicke et al., 2020]. The involvement of the end-users in the process has always proven to be very rewarding and has lead to the creation of successful products [Lindsay et al., 2012].

For our research, we have also followed a similar participatory design approach. In the earlier stages of this work, we gathered the requirements, needs, and aspirations from our community on a visualization software product through a user requirement survey. In the coming sections, we present detailed information about this survey.

2.1 Method

We performed a survey (Appendix A) to get direct feedback from our domain users about the domain specific operations they perform with different visualizations, challenges they face in visualizing their data and the technological assistance that can support them. This survey was done via the medium of a paper questionnaire and an online form at various conferences organized by German and international biodiversity organizations. These organizations are: GfÖ (The Ecological Society of Germany, Austria and Switzerland)[1], iDiv (German Centre for Integrative Biod-

[1] `www.gfoe.org`

iversity Research)[2], GFBio (German Federation for Biological Data)[3], CRC 1076 Aquadiva[4], BE (Biodiversity Exploratories)[5] and TDWG (Biodiversity Information Standards)[6].

The online survey was active from August 2015 until December 2017. Besides, a commentary paper [Kaur et al., 2016] with a survey link was also published in an international journal to reach a large audience. We have received 100 responses in total. Considering the outreach of participants through all these venues, this number is low. This is symptomatic of the limited willingness to share knowledge across interdisciplinary borders. Within the survey, some questions were multiple choice and others were single choice. For some questions, a commentary section was provided to allow the participants to provide additional information and viewpoints on different inquiries. For the convenience of the participants in completing the survey, no mandatory fields were added. This resulted in many questions remaining unanswered. Therefore, the scores calculated and presented in the next section are based on the number of answers for each question received rather than the total number of survey responses.

2.1.1 Results and discussion

2.1.1.1 Issues with visualization selection

Figure 3 shows that the majority of biodiversity researchers feel comfortable with their visualization skills and indicate not to face problems when selecting and creating visualizations. On the other side, the study participants have expressed (Figure 4) the need for a visualization support tool to assist them in these processes by recommending suitable visualizations. Through comments, they have directed their concerns on various issues they face when choosing a proper visualization. In the following, we have analyzed these comments and have categorized them into distinct visualization selection challenges:

- **Visualization selection dilemma:** The participants face difficulties in finding the best visualization solution to represent their data. Nowadays, with ample visualizations available, an appropriate visualization selection can become challenging as for a visualization layman; every other visualization looks the same.

- **Dependency on the visualization publication medium:** The participants find it more complicated to publish visualizations in journal articles, as it is costly to use colors. Whereas for online presentations, users have a wide selection and choice of visualizations which they can easily configure to make them more appealing to their audience.

- **Lack of knowledge:** The participants feel that they are unaware of alternative types of visualization techniques. Their visualization selection options are limited to what they have developed earlier or what they have seen in

[2] www.idiv.de

[3] www.gfbio.org

[4] www.aquadiva.uni-jena.de

[5] www.biodiversity-exploratories.de

[6] www.tdwg.org

previously published work. Due to this, they use similar visualization types repetitively.

- **Visualizing large and complex datasets:** The participants find it difficult to choose suitable visualizations to represent large and complex datasets. It is problematic to convey a message within multi-dimensional datasets clearly and precisely using a single figure.

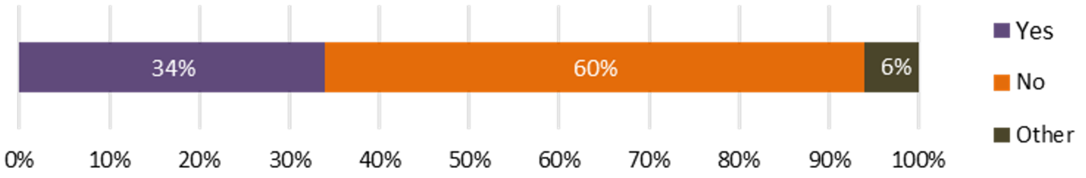


Figure 3: Do users find it difficult to select a visualization for presenting their data? The total number of responses received was 100.

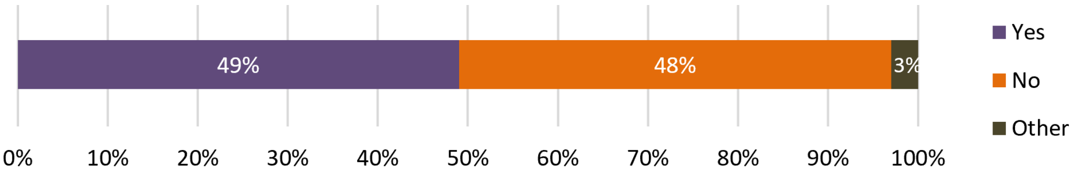
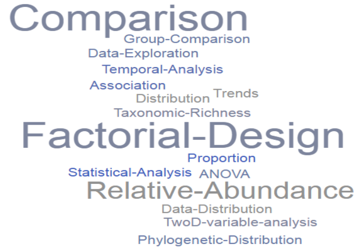
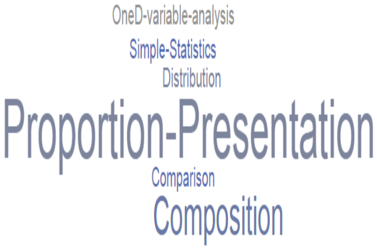
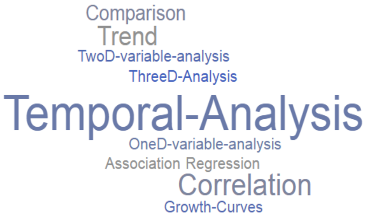
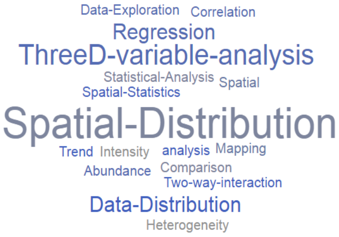
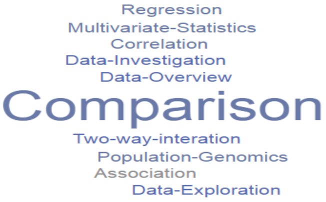
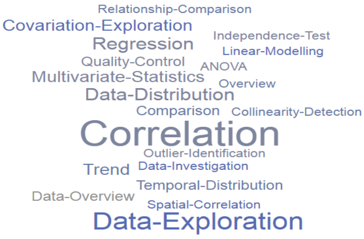
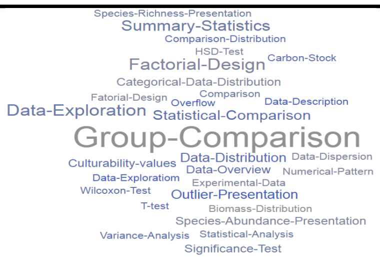
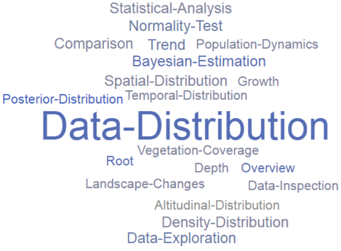
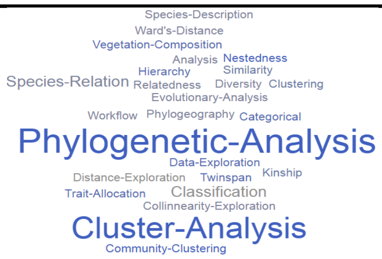
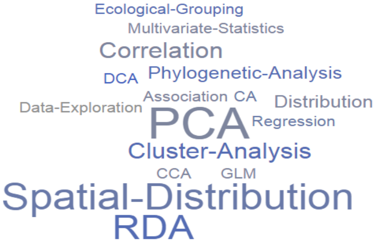


Figure 4: Are users interested in having a software tool that can guide them in the selection of suitable visualizations? The total number of responses received was 100.

2.1.2 Visualizations and their usage in the biodiversity domain

To this end, participants were shown a list of different visualizations and were asked to indicate the different purposes for using these visualizations in their daily work. This list was produced after knowing the types of common visualizations available in the biodiversity publications. In order to get a varied result, participants were asked to provide the answer to this question in a form of free-text. Table 2.1 shows the most frequently used visualizations and its usage. It's raw data is available in Appendix B. The word cloud associated with each visualization shows the usage or purposes indicated by the study participants. The larger the size of the word is, the more frequently it was mentioned by the participants. It is evident that biodiversity scholars use a spectrum of different visualizations for similar tasks; for example, the representation of data grouping and its comparison is done by scatterplot, boxplot, and bar chart. However, there are typically one or two tasks that are prominent to each visualization. Scatterplot for example, is used to illustrate the result of a principal component analysis (PCA) or to visualize the spatial distribution of objects, e.g., species. Dendrograms are frequently used for facilitating phylogenetic or cluster analysis. In Appendix E, we have taken examples from biodiversity publications regarding different visualizations along with their domain specific usage. The study participants were also asked to provide the reasons for not using some of

Table 2.1: Visualization types and the purposes they are used for in the biodiversity domain.

BarChart	PieChart
	
LineChart	**Heatmap**
	
Coplot	**ScatterplotMatrix**
	
Boxplot	**DensityPlot**
	
Dendrogram	**Scatterplot**
	

the listed visualizations. We have categorized these reasons into two groups: Never Needed and Don' Know (not aware of the visualization). Figure 5 indicates that Parallel Coordinates, Treemap, Venn Diagram and Coplot (conditioning scatterplot) are much less used than the other visualizations, although at least half of the respondents were aware of those types of visualizations. This raises question why those visualizations were not considered although most of them are more advanced and suited to multidimensional data. As it turned out, participants consider Parallel Coordinates as difficult to interpret and hard to comprehend. One participant said that instead of it he will prefer to represent different dimensions via different 3d plots. The study participants also noted that one of the reasons for rarely using Treemaps is that it is often dynamic and is thus hard to include in a paper. Some participants show more preference to Lattice Graphs than Coplot. The reason for the rare use of Venn Diagrams is that they are mostly known to represent concepts or ideas rather than numerical data. One participant said that he would like it when its area and colors were also meaningful.

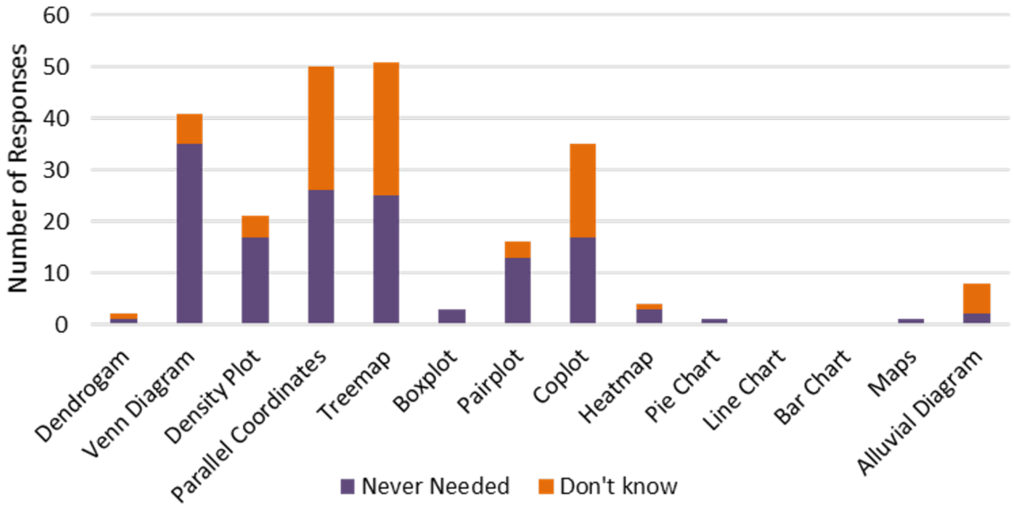


Figure 5: Stacked column chart showing the number of participants that never needed a given visualization or were not aware of it yet.

2.1.3 Visualization tool requirements

In the following, we have analyzed and have categorized the comments from the participants about their expectations on a software tool that supports visualization:

- **Visualization support for data management tasks:** We found out that visualization usage is not limited to the purpose of presenting results. There is a high necessity of visualization support for other data management related tasks. Our results (Figure 6) reveal that data analysis, result presentation, and data exploration are the three prominent tasks that can be effectively supported by visualization. Moreover, Figure 6 also conveys that researchers have started realizing the usefulness of visualization for data quality assurance and data search.

- **Factors for visualization selection:** Visualization tools offer visualization selection based on certain factors. In our study (Figure 7), we have found

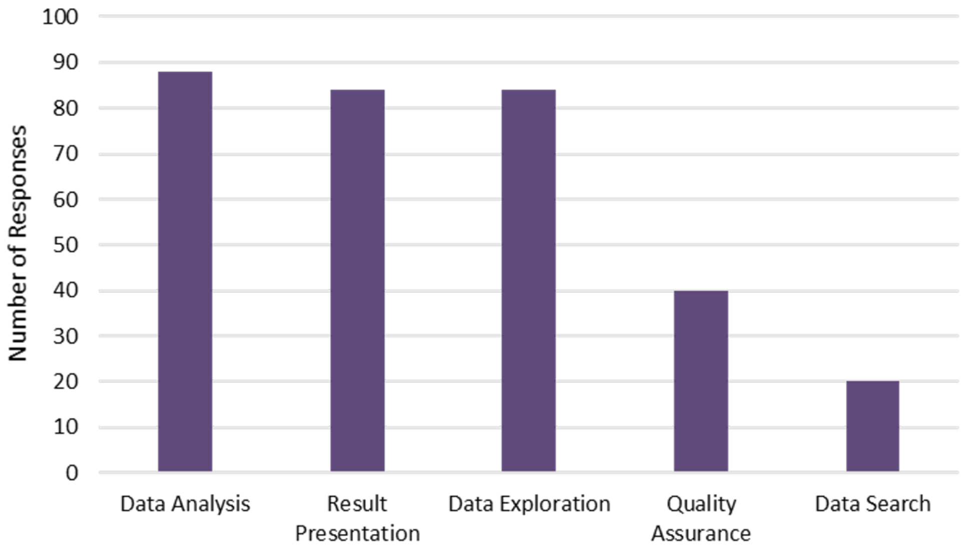


Figure 6: For what purposes do you use visualizations? Total responses received 99. Multiple answers per participant possible.

that scientists consider these three factors as most prominent for visualization selection: data type, aesthetics, and data size. Here aesthetics refers to the clarity and comprehensibility of a visualization. Comparing these factors to the ones presented in [Rougier et al., 2014], they are quite similar with an addition to the factor 'Ease of use' which our participants have indicated equally crucial as 'Data size'.

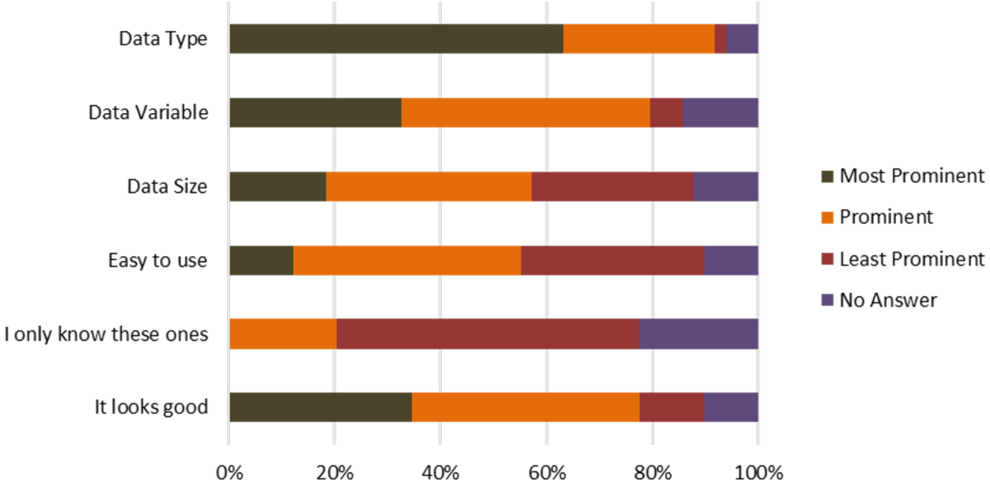


Figure 7: What factors do users consider while selecting a particular visualization for their task? Total responses received 99. Multiple answers per participant possible.

- **User centric:** The participants feel that a visualization software tool should not be too prescriptive or conditional. It means that it should provide a range of options (solutions) to the users to select from instead of selecting one for them. This also means that the solutions (or visualization recommendations) should not be fixed to and based on some preset conditions within the software. The software should be adaptive to integrate user responses or preferences in a real-time and then provide a personalized solution.

- **Easy to use:** The participants expressed their needs for a visualization software that is easy to access, use and understand. Instead of making a user guess on what procedures to follow to create a visualization, the software should guide the user at each step.

- **Showcasing:** The participants believe that showcasing what visualizations are present in the system will make them aware of the different options available within the tool. Such a showcase can be implemented in the form of a visualization knowledgebase, website or a guidebook. This can assist them in efficiently exploring, interpreting and developing graphical representations of their data.

- **Interactivity:** The participants consider interactivity as an important feature of a visualization. A visualization software tool should offer support for interactive visualizations. Interaction within a visualization helps explore the different data dimensions, gives a better overview of visualization and its elements, provides visualization customization, and enables the audience to engage with the visualization.

- **Multi-platform support:** The participants indicated that the visualizations produced by software tools should be flexible and platform-independent. This means that visualizations should be easy to export or import and should not depend on any one graphical tool. Other graphical platforms or tools should able to easily alter them.

- **Color-deficient friendly:** The participants want visualization tools to produce color-blind friendly visualizations so that the color-blinded community can effectively use them. A color-blind person has trouble seeing shades of red and green or yellow or blue[7]. So, the visualizations produced for them either avoid such color combinations, include both textures and patterns instead of only colors, use colors with high contrast, leverage symbols wherever possible, or use special color-blind friendly color palettes.

- **Visualization audience:** The participants also consider the visualization audience as one of the important factors in the visualization selection process that needs to be considered within the tool. The visualization selection will be different if the visualization is going to be presented to graduate students, experienced scientists, layman or stakeholders.

2.1.4 Data exploration workflow

One of the other important data management tasks for which visualization is used is data exploration, as shown in Figure 6. Data exploration provides a sneak peek into the data at hand and thus helps make an initial decision about the relevance of a dataset for answering a certain research question. What steps need to be followed to get an initial exploration and understanding of the data? We have asked our survey participants to provide their experience about how they explore a dataset. We have summarized their answers into the following four major steps:

[7] https://www.aao.org/eye-health/diseases/what-is-color-blindness

- **Data Pre-processing:** In this step, if the dataset is not clean, then one would investigate data for various quality issues and perform necessary cleaning. Then one would further examine the different features of the data (for example, data dimensions, data size, data types).
- **Data Overview:** In this step, one might perform the following actions:
  - getting an overview of the dataset via different multi-dimensional visualizations
  - examining the distribution of the data to understand if it is skewed or symmetric
  - detecting outliers
  - summarizing the data for further statistical analysis or refinements
- **Data Refinement:** In this step, one might perform the following actions:
  - filter or subset the data based on the individual analysis goal
  - transform the data (for example at different scales to remove skewness)
  - create the derived or compound variables as per the analysis requirements
  - remove outliers if those were spotted in the previous step
- **Data Analysis:** In this step, one might perform the actual analytical tasks like hypothesis formulation, understanding relationships existing within a dataset or doing comparisons.

These are the preliminary steps that researchers follow to explore the data wherein different visualizations are needed to facilitate each step. Data investigation is the foremost step that users perform. Then depending on the individual goals, some of the remaining steps follow in non-particular order. For example, if a user has some information about the data then the user will go for data refinement to explore the variable of its interest. Whereas, if a user has no prior information about the data, then the user might be interested in seeing a multi-dimensional view to get an overview of the complete dataset and then choose the variables of interest. After the refinement step, the user might be interested in summarizing the variables of interest and then would want to do further analysis. Again, after analysis, the user might perform further data refinement or get an overview of the altered dataset as per its requirements.

Our last three steps are also somewhat similar to Shneiderman's Mantra of Visual Information Seeking, i.e., "Overview first, zoom and filter and details-on-demand" [Shneiderman, 1996]. Once the data is pre-processed or cleaned, the user is typically interested in getting the overview of the data first, which is similar to our second step. Then, she refines the data using a filter or other techniques, similar to our data refinement step. Finally, if the user is further interested, she performs data analysis which is the same as details-on-demand. Apparently, these tasks are complex and abstract tasks which can further breakdown sequentially by using typologies like the one presented by [Brehmer and Munzner, 2013].

2.2 Summary

Our study revealed that although biodiversity researchers feel comfortable with their current visualization practices, they wish to have a software support to choose appropriate visualizations to represent their data. Major challenges arise from many visualizations available today and from the increased size and complexity of the data to visualize. Due to which, they are not able to keep pace with the current visualization developments. Therefore, a tool developed for them must not be too complicated and at the same time should not be too prescriptive. It should help users follow a simple and intuitive workflow to concentrate on the data insights rather than only understanding the tool's functionalities.

We have also observed that apart from using visualization for data presentation and analysis, users now realize the usefulness of visualization for other data management tasks like data exploration, data search, and quality assurance. Thus opening up a research dimension for the visualization community to provide visualization as a service to the data management process at its different stages.

The resultant data for this survey is available online[8]. The results from this requirement analysis survey strengthened our vision to create a visualization assistance tool for them. From these results, we were certain to work towards the visual data exploratory tool for the BExIS data management system. This tool should provide data-driven visualization assistance to explore the dataset's multi-dimensionality. In the next chapter, we explore the state-of-the-art on such visualization recommendation tools for data exploration.

[8] `https://github.com/PawanKaur/SurveyDataset`

Chapter 3

State-of-the-Art

**Part of this chapter is based on work published in the Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: IVAPP'17, 266-273, Porto, Portugal*

In the previous chapter, we have presented the results from our visualization tool requirement survey. We have observed that our community needs data-driven visualization assistance software to explore the multi-dimensionality of the biodiversity datasets. In this chapter, we will explore various such available visualization technologies to fulfill the needs of the community. We will investigate this by presenting a study on the currently available visualization technologies specifically for the biodiversity community and the current state-of-the-art studies in the related visualization science.

To do so, we have divided this chapter into different sections, wherein we will review the literature based on the following topics:

1. Visualization tools for the biodiversity domain
2. Visualization recommendation systems
3. Variable selection algorithms

3.1 Visualization Tools for the Biodiversity Domain

While reviewing the available literature and tools related to the biodiversity domain, we have observed that there are different dimensions in which the community use these visualization tools. In the following, we have enlisted some of these observed dimensions in four different groups:

- **Question specific visualization tools:** We have put those visualization tools in this group, which are developed specifically to answer particular questions about the biodiversity domain. For example, Keanu [Thrash et al., 2019] is a visualization tool that shows the presence and abundance of organisms in a sample, by analyzing sequence content alignments against a database that contains taxonomical data. TaxonTree [Lee et al., 2004] is an interface to visualize the Linnaean Classification for taxonomic names in the kingdom Animalia. TaxonTree allows users to browse and search a tree of about 200,000 animal names constructed by integrating data from several public and private sources.

- **Visualization as an aid to data search:** Software tools in this group use visualization assistance to search datasets from data repositories. In most cases, these visualization tools are tightly integrated with a specific data repository from which it provides the dataset. GBIF-MAPA [Flemons et al., 2007], i.e., GBIFs Mapping and Analysis Portal Application (MAPA), is a tool deployed at distributed GBIF database portals. It is a web-based biodiversity workflow application that provides users the means to semi-automate raw biodiversity data acquisition, shows geospatial visualization, and deployment of core biodiversity analyses based on that data. The GFBIO VAT (The Visualization, Analysis, and Transformation System) [Beilschmidt et al., 2017] is another such GIS system to visualize the GFBIO data providers and access the collections at different biodiversity data centers.

- **Visualization to represent the geographical species distribution:** There is an abundance of visualization research done to show the geographical distribution of different species. For example, antmaps.org [Janicki et al., 2016] is a client-server GIS-based web-mapping application to visualize and interact with all other ant species' geographic distributions throughout the globe and aggregate patterns of their diversity and biogeography. Ebird[1] is another GIS-based system to explore the birds and their hotspot all around the globe. Another similar one is inatualist[2], to visualize the distribution or abundance of multiple species. Herbaria [Auer et al., 2011], is another GIS-based application that aids in the visual exploration of the species California Flora's dataset to understand plant diversity patterns, distribution ranges of species, and vegetation associations for specimens held in physical collections.

- **Visualization for data exploration or analysis:** Here we have included those tools that help in the exploration and analysis of a single dataset. From our requirement analysis survey results, we have observed that most biodiversity researchers use R to visualize their dataset (Figure 8). As R is primarily a scripting language and is not a user interface, it does not fit our review criterion. Second to R is Microsoft Excel. Microsoft Excel is a spreadsheet software created initially for business intelligence usage. Though it has a large user base, it has limited visualization capabilities. Visualization is a small module included in the software. We have provided the comparison between the Microsoft Excel and our tool in Chapter 10. Wherein, because our tool was more domain-based and user friendly, it has outperformed Excel in almost all evaluated categories. Next to the Excel are the tools developed for the GIS (ArcGIS[3] and QGIS[4]) or spatial analysis purposes. Next to that are a bundle of software like (SIGMA PLOT[5] and SPSS[6]). All of them are statistical software that provides statistical visualizations to show the results of the analysis. For

[1] www.ebird.org

[2] www.inaturalist.org/observations

[3] www.arcgis.com

[4] www.qgis.org

[5] www.systatsoftware.com/products/sigmaplot

[6] www.ibm.com/products/spss-statistics

running them, one needs to have an adequate level of statistical and analytical knowledge, and one should already know what sort of analysis needs to be done. These software tools are not based on one particular scientific domain. Thus we again cannot consider them as knowledge-based visual exploratory tools.

From the first three groups in the provided list, we have observed that all these visualization tools are based on the synthesis and mapping of different datasets to fulfill certain goals, i.e., either showing hierarchy or showing aggregated data distribution or data publishers. The synthesis of the data is not a first step in the creation of such an application. First, one needs to explore, analyze, and evaluate each dataset to see if it fulfills the specific purpose or not. The last group in the above list is about the non-domain specific statistical tools. They are used for analytical purposes after knowing the dataset's suitability to the specific analysis task. However, none of the above shown software provides a simple way of visually exploring the dataset using the knowledge of the target domain.

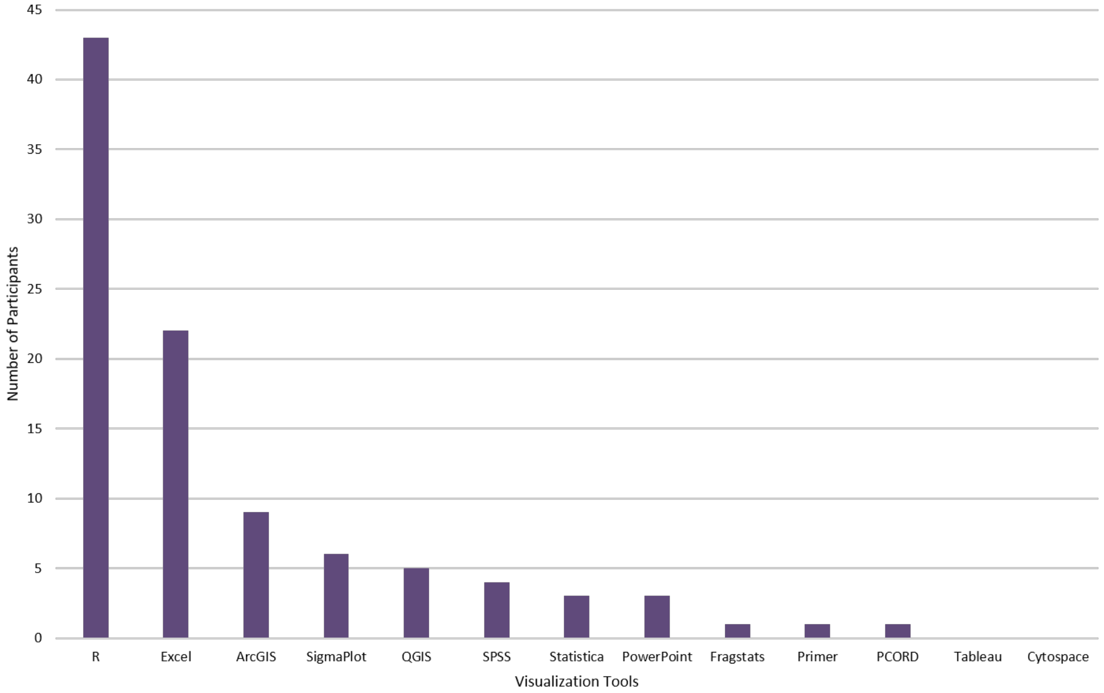


Figure 8: Bar chart showing typical visualization tools used by the biodiversity users.

3.2 Visualization Recommendation Systems

The potential utility of graphics can only be assessed if we are able to answer what type of graphic should be used? - Jacques Bertin [Bertin, 1983].

Based on the most distinguishing factors identified by [Vartak et al., 2017], we classify approaches to visualization recommendation into four distinct categories. These categories are defined according to the main contribution of their research in providing techniques, guidelines, or directions that assist in recommending visualizations.

1. **Data characteristics oriented:** Studies that fall in this category recommend visualizations based on data characteristics.
2. **Task oriented:** Studies that fall under this category use the representational goals along with the data characteristics to recommend visualizations.
3. **User preferences oriented:** Studies that fall under this category gather information about the user presentation goals and preferences explicitly through user interactions with the visualization system.
4. **Domain knowledge oriented:** Studies that fall under this category improve the visualization recommendation process with domain knowledge.

3.2.1 Data characteristics oriented

Visualization recommendation research studies in this category have tried to improve the understanding of the data, different relationships that exist within the data and procedures to represent them. The choice of variables to represent different aspects of the same information can significantly influence the perception and understanding of the presented information. Therefore, the research under this category focuses on the definition of new data dimensions or attributes, the formalization of the process of visual mapping from data attributes to visual marks, and the introduction of new techniques for visual mapping. The earliest known study that proposed automation of graphical designs was that of Gnanamgari's Bharat in 1981. As cited by [Bouali et al., 2016], Bharat proposed some rules for determining which type of visualization is appropriate for specific data attributes. However, their work was based on the limited set of visualizations available in 1981. Mackinlay's APT system [Mackinlay, 1986] proposed to formalize and codify the graphical design specifications to automate the graphics generation process. This was based on composition algebra, which consists of a basis set and composition operators. Before applying this algebra, data attributes need to be encoded with the respective visual mark, which should be consistent with the rules presented in Table 3.1.

Table 3.1: Data attributes to visual attributes mapping [Mackinlay, 1986]

Visual attributes	**Nominal**	**Ordinal**	**Quantitative**
Size	-	●	●
Saturation	-	●	●
Texture	●	●	●
Color	●	●	-
Orientation	●	-	-
Shape	●	-	-

In Composition Algebra, the basis set encodes data attributes to visual variables or attributes (Table 3.1). Compositional operators generate different presentations by composing different basis sets from different data attributes. They composed visualizations by merging parts that encode the same information. For example, two single-axis plots with a visual mark *'dot'*, can be composed of a 2D scatterplot. Later, the specifications based on Mackinlay's heuristics were used to develop a research system called Polaris [Stolte et al., 2002]. These specifications were then revised into a formal declarative visual language known as VizQL [Hanrahan, 2006].

The visualization software Tableau's[7] Show Me module [Mackinlay et al., 2007] uses VizQL specifications to recommend visualizations automatically. When the user selects the data attributes of its interest, Show Me uses Tableau's Visual Mapping rules shown in Table 1.1, to define the visualization types.

In order to enhance the understandability of the data and the process of visual encoding, [Roth and Mattis, 1990] argued that more structural and semantic information about the data relevant to the presentation design should be provided. Therefore, they proposed a richer set of data characterizations, divided into different data domains, to be used by humans or machines for designing visualizations. It includes original data measurement scales by [Mackinlay, 1986], along with new data descriptors: Spatial (coordinates, name of the city), Amount (count and discrete data), Range (duration). They have identified and grouped the data domains into coverage, cardinality, and uniqueness. Coverage conveys whether every element of a set can be mapped to at least one element of another set. Cardinality expresses the dependency and 'within' relationship between two or more attributes of the same dataset: one to one, one to many, and many to many. Uniqueness refers to the uniqueness of values within a set or data column. Their proposed characteristics are used in the SAGE (System for Automatic Graphical Explanation) software.

Unlike previous work, where researchers seek knowledge among different variables of the dataset, Shneiderman's theory [Shneiderman, 1996] emphasized on considering the dataset as a whole collection and understanding the overall relationship between a single collection (like hierarchical data) or within different data collections. He has categorized the data into seven dimensions: 1-dimensional, 2-dimensional, 3-dimensional, multi-dimensional, temporal, tree, and network data. This proposal serves as the basis of the implementation of the TIBCO Spotfire [Shneiderman, 1999]. In the previously mentioned studies and tools, visualizations were generated offline by specialists. The 'Many Eyes' had changed this trend and provides the first known public website where users may upload data and create interactive visualizations collaboratively [Viegas et al., 2007]. In Many Eyes, a visualization is created by matching a dataset with the visualization components (or visualization techniques). The list of visualization components is provided in Table 3.2.

Table 3.2: Many Eyes visual mapping scheme [Viegas et al., 2007]

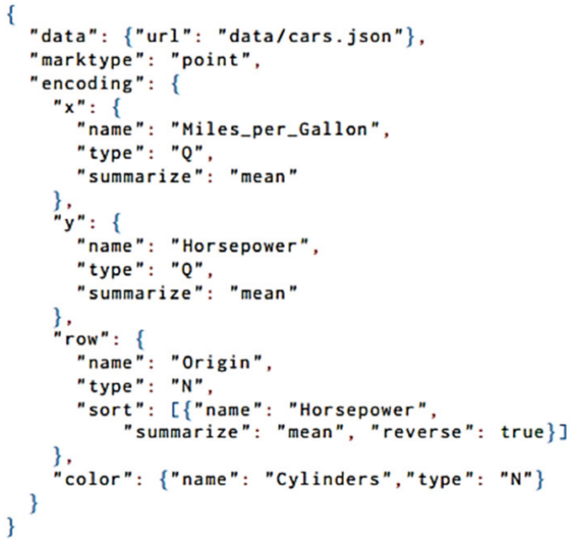
Technique	Data schema
Bubble Chart, Histogram, Pie Chart, Maps	Labels: T, Values: N
Bar Chart, Line Graph, Stack Graph	Axis: T, Values: N+
Network	T+, to : T
Scatterplot	Xaxis: N, Yaxis: N, Label: T, Dotsize: N
Stack Graph/Categories	Hierarchy: T+, Values: N+
Treemaps	Hierarchy: T+, Size: N, Color: N
Tag Cloud	U

Each row consists of a visualization technique that shares a common data schema. When the user selects some data columns, they are mapped with the data schema associated with some data visualizations. A data schema is a set of named, typed slots. For example: 'T' in the above table is single column textual data, and 'T+' means the dataset has more than one textual data column. Thus, a treemap (Table 3.2)

[7] www.tableau.com

can be expressed as an ordered set of textual columns, where each row in the set describes the path from the top of the hierarchy to the leaf item. The dataset and the produced visualizations can then be shared with other users for comments, feedback, and future improvement, thus providing a collaborative workbench for visualization creation.

Many Eyes popularity has proved the usability and ease of deploying visualization software as a web application. Along with that, the dashboard environment provided by Tableau also became a standard for visualization creation interfaces. Voyager [Wongsuphasawat et al., 2015] is a visualization recommendation web application based on the dashboard type environment. Voyager uses the Compass Recommendation Engine, which suggests visualizations based on the statistical properties of the data. The suggestions are produced in the form of Vega-lite specifications [Satyanarayan et al., 2016]. A Vega-lite specification is a JSON object (Figure 9) that describes a single data source, a mark type, visual encoding of data variables, key-value, and data transformations including filters and aggregate functions. The Compass Recommendation Engine first suggests a list of visualizations based on each variable's univariate summary in the dataset. Then the user can exclude or include variables from the list to focus on a particular variable set of interest. Similar to the study by [Satyanarayan et al., 2016], recent studies have tried to exploit the statistical characteristics of data as assistance to visualization recommendation.



```
{
  "data": {"url": "data/cars.json"},
  "marktype": "point",
  "encoding": {
    "x": {
      "name": "Miles_per_Gallon",
      "type": "Q",
      "summarize": "mean"
    },
    "y": {
      "name": "Horsepower",
      "type": "Q",
      "summarize": "mean"
    },
    "row": {
      "name": "Origin",
      "type": "N",
      "sort": [{"name": "Horsepower",
          "summarize": "mean", "reverse": true}]
    },
    "color": {"name": "Cylinders","type": "N"}
  }
}
```

Figure 9: Vega-lite JSON Object [Satyanarayan et al., 2016]

VizDeck [Key et al., 2012] is another such initiative. It automatically recommends ranked and coordinated visualizations based on the statistical properties of the data. VizDeck adopts a card game metaphor to organize multiple visualizations into interactive visual dashboard applications. When a user selects a data, the system initially presents the xy charts' small multiple views (scatterplot or line chart based on the data attributes). Users interact with these vizlets while keeping the good ones and discarding the unwanted vizlets. User interaction makes a system learn which vizlets are more likely to be useful for a dataset with particular features. The learned information enhances the system's ability to recommend more suitable visualizations when provided with similar data in the future. A study by [Vartak et al., 2017] used statistical methods of a probability distribution, distance matrices and deviations to suggest the different bar chart and line chart views. Their prototype SEEDB computes a deviation of the subset of the data in comparison to the whole dataset. It then recommends those visualizations for which the underlying data (a subset of data) has a high deviation from the current and regular trends reflected in the whole dataset. They argue that users find visualizations with high deviations more interesting and expressive.

More recent studies have used machine learning-based techniques to train a classifier on the historical data and their visualizations. Based on the features a classifier is trained on, these studies recommend visualizations for the provided data. With their trained binary classifiers on data and visualization constraints, application by [Luo et al., 2018] first recognizes if the visualization is good for a dataset or not. Through their supervised learning-to-rank model, they decide the appropriate ranking of the visualization, and then using rule-based optimization they select and provide the top-k suitable visualizations. VizML [Hu et al., 2019] is trained on design choices from a corpus of data-visualization pairs. They have described visualization recommendation as a problem of developing models that learn to make design choices.

In Table 3.3, the contributions provided by the studies in this section are classified into five broad areas based on their work towards better visualization recommendation:

1. **Data properties definition:** by providing richer sets of data dimension and characterization.

2. **Rule definition:** by providing rules, specifications and schemas to manipulate data and perform visual mapping.

3. **Language formalization:** by defining specifications in system understandable language to automate the process of visual mapping.

4. **Statistics based:** by using statistical and exploratory data analytics procedures to recommend visualization.

5. **Machine Learning:** by training machine learning models to learn data or chart characteristics or both to produce future recommendations.

Table 3.3: Classification table

Categories	**Studies**
Data Properties	SAGE, TIBCO Spotfire
Rule Definition	APT, Many Eyes
Language Formalization	VizQL, Vega-Lite
Statistics	Voyager, VizDeck, SeeDB
Machine Learning	DeepEye, VizML

3.2.2 Task oriented

Visualization recommendation research studies in this category have designed different techniques to infer the representational goal or user's intentions behind visualizing the data. Differences in goals can significantly alter the effectiveness of graphical designs. A study by [Roth and Mattis, 1990] was the first to contribute to the idea of instigating the user's information seeking goals in the visualization design process. Their study identified different domain-independent information-seeking goals, e.g., comparison, distribution, correlation and many more. Based on some sets of representational goals, a classification scheme for visualization recommendation was proposed by [Wehrend and Lewis, 1990] in the form of a 2D matrix of 'objects' vs. 'operations'. In this matrix, 'objects' are data attributes, 'operations' are representational goals, and cells contain visualization techniques. According to [Kerpedjiev et al., 1997], visualization recommendations can further be enhanced by using domain-level tasks. They introduced the idea of decomposing representational goals from the domain-specific goals. Hence, they proposed a model (Figure 10) to hierarchically decompose domain-specific user's goals (for the 'transportation scheduling' domain) into common domain-independent goals or representational goals, which are further associated with some graphical actions or operations. For example, in Figure 10, domain-specific goals like 'know-shortfalls' (which means to know the daily shortfalls in the goods transported) were decomposed to tasks that include 'know-difference'. In turn, 'know-difference' is associated with 'differentiate', a high-level domain-independent task or goal that acts on data. Actions associated with 'differentiate' include 'enable-lookup' on the value of individual days and 'enable-comparison' on those values. This approach was applied in the development of AutoBrief [Kerpedjiev et al., 1997], which is a multimedia presentation system that assists in data analysis.

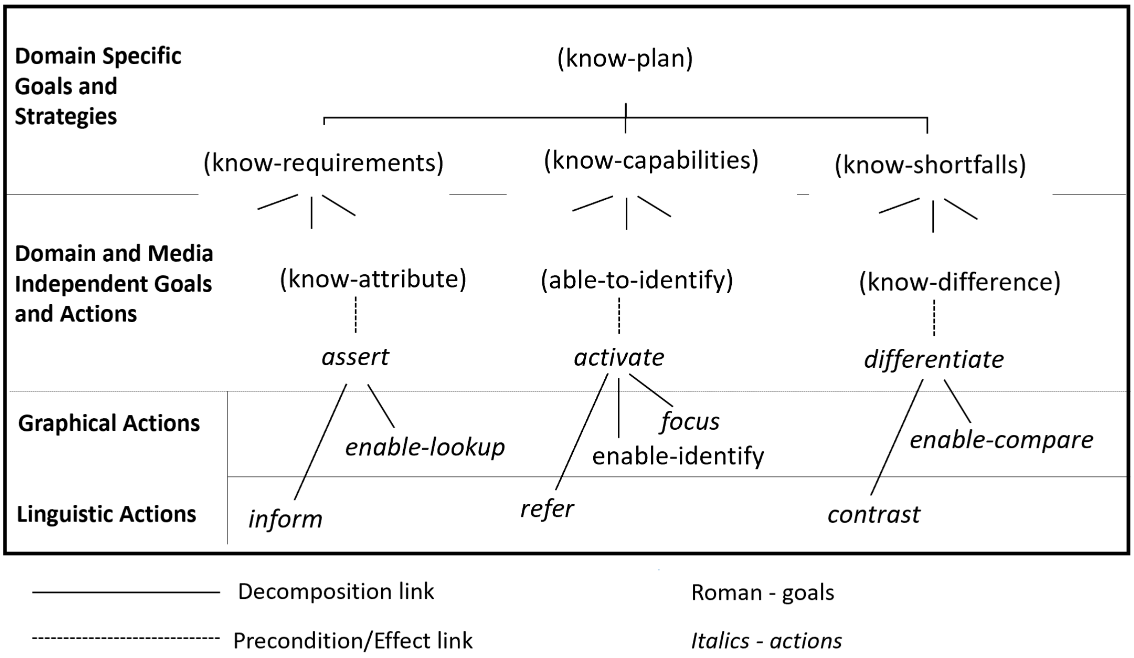


Figure 10: Decomposition of goals into actions by [Kerpedjiev et al., 1997]. Adapted from "Autobrief: a multimedia presentation system for assisting data analysis" by S Kerpedjiev, G Carenini, S. F Roth, and J. D Moore, 1997, Computer Standards Interfaces, 18(6-7):587.

In all the previous studies, the user task list was manually created. Advancements in linguistic research seek an opportunity to automate the user task's derivation from a natural language query in the visualization creation process. One such study [Zhou and Feiner, 1998], introduced visual task taxonomy to automate gaining a high level of presentation intents from the text. Their taxonomy associates high-level tasks (presentation intent) to low-level visualization techniques (visual action). For example, the visual task Focus<?x> implies that visual techniques such as Enlarge<?x> or Highlight<?x> could be used to focus attention on ?x. Their taxonomy and techniques are implemented in IMPROVISE (Illustrative Metaphor Production in Reactive Object-Oriented Visual Environments). Studies, e.g., Eviza [Setlur et al., 2016], and Datatone [Gao et al., 2015], are based on the advancement of this idea of generating visual tasks from the natural language interface at the run time. These studies have applied advanced text mining and natural language processing techniques and used semantic technologies based on existing knowledge-bases like Wolfgram[8] or Wordnet[9], to map different concepts with their defined tasks. Recent developments of deep learning models have been applied to understand the semantics and characteristics of the charts and the data. ChartSeer [Zhao et al., 2020] uses deep learning models to convert user-provided charts into semantic vectors defining visual goals (trend, pattern, outliers). Then based on these recognized chart semantics, it provides a system recommended charts for further visual exploration.

3.2.3 User preferences oriented

Here, those visualization recommendation strategies are grouped, which gather users' intentions explicitly from their behavior and interaction records while they communicate with the visualization system. They are also known as behavior-driven studies. Some studies also use probabilistic and machine learning techniques to predict the patterns of user choice from these records. The first known behavior-driven study is from [Gotz and Wen, 2009]. BDVR (Behaviour Driven Visualization Recommendation) consists of two distinct phases: Pattern Detection and Visualization Recommendation. In the first phase, user behavior while interacting with the visualization system is analyzed to find meaningful interaction patterns. These patterns are, e.g., scan, flip, swap, and drill-down. In the second phase, a recommendation engine infers a user's intent from these detected patterns. In the case of 'scan pattern', e.g., the user interactively 'inspects' values over a series of data. Then they 'compares' those series within themselves or over time. Visual tasks are inferred from these intents, suggesting an alternative visualization to the user that suits more accurately than their current visualization selection. A similar study conducted by [Steichen et al., 2013], has provided results on accumulating information from user eye gaze patterns. They recorded the interaction of the user with a given visualization to predict the users' visual goals, as well as user cognitive abilities, including perceptual speed (a measure of speed when performing simple tasks), verbal working memory (a measure of storage and manipulation of the capacity of verbal information), and visual working memory (a measure of storage and manipulation capacity of visual and spatial information). They have shown that such characteristics significantly affect task efficiency, user preference, and ease of use with visualization systems. These findings are presented in view of designing visualization systems that can adapt to each user in real-time. Towards the recommendation of more user-centric and user-adaptive visualization tools, many systems

[8] www.wolframalpha.com

[9] https://wordnet.princeton.edu/

have applied machine and probabilistic learning approaches from the user interactions while they browse through the recommended visualizations as in the case of [Key et al., 2012]. Study by [Mutlu et al., 2016] used techniques like collaborative and content-based filtering to suggest charts by deriving a similarity matrix according to the information needs of the user and chart characteristics. First, they have designed a crowdsource study to obtain personalized scores and tags on each visualization. Then a multi-dimensional scale is used to estimate the quality of charts for collaborative filtering, and a tag vector is used to recommend potentially interesting charts based on content. Recent work by [Xu, 2019] uses a machine learning model to train a system first on user stories and then user profiles. Their tool, ReViz, builds a multiple linear regression model based on saved user data and then uses this model to give new suggestions based on new user stories and metadata.

3.2.4 Domain knowledge oriented

In the visualization development process, it is essential to first characterize the task and data in the vocabulary of the problem domain so that the visualization can fulfill the requirements of users in any particular target domain [Munzner, 2009]. The objectives of domain knowledge-based approaches include sharing such knowledge among different designers and end-users and reducing users' burden to acquire knowledge about complex visualization techniques. Such approaches are not core techniques to produce a visualization, but they assist with other techniques for improving performance while recommending visualizations. The studies falling into this category deal with gaining the domain knowledge from existing knowledge sources or creating a new one, which further assist in the visualization recommendation process.

Though research is abundant in the field of knowledge-based visualization systems ([Federico et al., 2017, Wagner et al., 2018, Wagner et al., 2017]), there is minimal research in the subject of domain knowledge-based visualization recommendation systems. It is imperative to know that both approaches are very much different. Knowledge-based visualization systems use existing domain knowledge to answer a particular domain problem. Knowledge-based visualization recommendation systems do not specifically answer one particular type of domain question. Instead, it shows different visual insights into the datasets for better data exploration and understandability.

The earliest known knowledge-based visualization recommendation study is RAVE [Klumpar et al., 1994]. RAVE has been used for the visualization of in-situ measurement data captured by the NASA spacecraft. The user needs to select either a visualization type or a representational goal from a provided list. On user selection, RAVE triggers the visualization technique associated with the entries in a list and provides the resultant graphics. RAVE's knowledge-base contains: (1) a set of visualization objects that corresponds to a specific visual technique that can create that visualization, (2) a set of rules that corresponds to the selection of one particular visualization technique, (3) the high-level task that visualization can perform like correlation for a scatterplot, (4) the refinements that visualization can accept and (5) the domain(s) in which it can be used. For example, the visualization object that corresponds to the 2D scatterplot can satisfy the rule 'attribute x is related to attribute y', can accept zooming and color as refinements, and can be applied in any domain where numerical attributes are compared.

To include semantic abilities in the process of recommendations, studies from [Gilson et al., 2008] proposed a pragmatic approach for automatic generation of visualizations from domain-specific data available on the web in the form of ontologies. They described a pipeline that combines ontology mapping from three different ontologies. In their approach, a web page is first mapped to a 'domain ontology', which stores the specific subject domain's semantics. The 'domain ontology' is then mapped to one or more 'visual representation ontologies', each of which captures the semantics of visualization types. A 'semantic bridging ontology' bridges the information from the two ontologies and holds vital knowledge about the relationships between data entities of the source, the subject domain and the visual artifacts of the target visualizations. They have implemented the visualization pipeline in a prototype, SemViz, which functions end-to-end from a source web page to the target visualization. Building upon somewhat similar grounds, [Voigt et al., 2013] propose a novel approach for knowledge-based, context-aware visualization recommendation for semantic web data. VISO is a modular visualization ontology composed of seven modules that provide a vocabulary to annotate data sources and visualization components. *Graphic* module formalizes knowledge in the domain of visualization. *Data* module characterizes the data variables and structure. *Activity* module is concerned with the human aspects of visualization, i.e., tasks, actions, and operations. *System*, *user*, and *domain* module describe the data and visualization context and the domain information. Based on the different modules' shared knowledge, a recommendation algorithm covers both the discovery and context-aware ranking of suitable graphic representations.

3.3 Variable Selection Algorithm

In an era of data abundance of a complex nature, it is essential to extract useful and valuable knowledge from the data. One crucial step in this process is selecting relevant and non-redundant information or features from the dataset to clearly define the problem at hand and aim for its solution [Macedo et al., 2019].

Therefore, these days variable or feature selection have become important research application areas for which datasets with hundreds or thousands of variables are available. Some of its benefits are: better data visualization and understanding, and reducing the measurement and storage requirements. Benefits for machine learning applications are: reducing training and utilization times, defying the curse of dimensionality to improve prediction performance [Guyon and Elisseeff, 2003]. Feature selection techniques can be categorized as classifier-dependent (wrapper and embedded methods) and classifier-independent (filter methods). Wrapper methods [Kohavi et al., 1997] search the space of candidate feature subsets using a classifier's accuracy. There are clear disadvantages to using such an approach. The computational cost is enormous, while the selected features are specific for the considered classifier. Embedded methods [Guyon et al., 2008] exploit the structure of specific classes of classifiers to guide the feature selection process. In contrast, filter methods [Guyon et al., 2008] separate the classification and feature selection procedures and define a heuristic ranking criterion that acts as a measure of the classification accuracy.

As per the recommended checklist provided by [Guyon and Elisseeff, 2003], if it is possible to get the domain knowledge, then the best method for variable selection is to create an ad-hoc algorithm using that knowledge. In contrast to the data-driven methods discussed before, the ad-hoc domain-specific algorithms rely on the knowledge gathered directly from domain experts or other domain sources. Combining prior domain knowledge as a part of machine learning projects would complement the data-driven approaches [Bochare et al., 2014, Islam et al., 2018]. Different research domains use domain knowledge in feature or variable selection algorithms: breast cancer prediction model [Bochare et al., 2014], predicting airline ticket prices [Groves and Gini, 2013], oral disease prediction [Li et al., 2018], etc.

However, all these discussed applications are employed for machine learning problems, where the focus is on boosting the accuracy of feature selection methods [Georges et al., 2020]. Our application of variable selection is for data visualization and our goal is to provide interesting and relevant features that can provide useful data insights. It is important to note that visualization is not used for variable selection, but the variable selection is used to provide data insights through visualization.

To know which variable or feature selection techniques are used by the visualization recommendation systems, we reviewed some studies to understand their variable selection techniques. In the visualization recommendation tool by [Bouali et al., 2016], the user has to explicitly provide a score to the variables, for their algorithm to decide the relevance. Voyager [Wongsuphasawat et al., 2015] does not make any variable selection and shows the univariate summary of all the variables. SeeDB [Vartak et al., 2017] uses deviation from reference as a criterion for searching the interesting variables and finding the appropriate visualizations. Vizml [Hu et al., 2019] and DEEP EYE [Luo et al., 2018] have trained the algorithm on the set of the visualizations and the datasets they represent. Based on their classifier's decision, they pre-select the variables and corresponding visualizations for users.

3.4 Summary and Discussion

In this chapter, we have provided a review of different visualization techniques available for the biodiversity community. We have observed that the community is already leveraging different visualization software to fulfill their different purposes. However, we could not find any study or software that visually explores a dataset's multi-dimensionality without getting much into statistical analysis. Moreover, we did not find any available visualization recommendation tool or study for the biodiversity community.

The literature review on different visualization recommendation studies shows the scarcity of domain knowledge-based systems that can recommend or assist users in selecting suitable visualizations for exploring or analyzing their datasets. A more detailed discussion on these shortcomings and our solution is provided in the next chapter.

Chapter 4

Problem Statement

In Chapter 2, we tried to understand the needs and demands in the biodiversity community regarding current visualization practices. The main aspects of the survey analysis can be summarized as follows:

1. The biodiversity community starts realizing that visualization tools can be utilized for important data management tasks in addition to data presentation and analysis, like data exploration, data search, and quality assurance.

2. Biodiversity researchers wish to have software support to choose appropriate visualizations for their data.

3. Major challenges in visualization creation arise from the plethora of visualization options available and from the large size and complexity of the data to visualize.

Our survey confirms that the biodiversity community needs visualization support or a recommendation tool for visual data exploration. The visual exploratory tool should assist the domain users with choosing appropriate visualizations for the data. Such a tool must not be too complicated and should help users follow a simple and intuitive workflow to focus on the data insights.

A review of the literature regarding visualization recommendation studies shows that domain knowledge-based visualization recommendation studies are very scarce. Though visualization recommendation studies have used state-of-the-art machine learning technologies to train the model on the visualization and data attribute knowledge [Luo et al., 2018, Hu et al., 2019], they are not based on the target community's domain knowledge. Studies that have used data semantics from a specific community are highly dependent on the efficiency of the ontology-matching strategies. When the domain is as vast as biodiversity, ontologies can be very large (as in ENVO[1] with 6199 classes). Gathering domain knowledge only based on ontologies tends to be insufficient because not all biodiversity-related information is available in single ontology. For example in [Löffler et al., 2020], researchers had to use ten ontologies to gather information for only four biodiversity entity types, i.e., Environment, Process, Material, and Quality. Despite these constraints, the usefulness of semantic information obtained from ontologies about the data context can not be ignored.

[1] `www.bioportal.bioontology.org/ontologies/ENVO`

The literature review also revealed that very few studies used automated variable selection algorithms for visualization recommendation systems. Moreover, in these studies the automatic variable selection algorithms are based on the data variables' statistical properties or classification results from the trained datasets. The inclusion of context or domain-knowledge is limited in such algorithms.

After a thorough evaluation of the community's visualization demands, we proposed to provide a visualization system that meets the following requirements:

1. The tool provides a visual exploration of the dataset for better data understandability.

2. The tool provides visualization selection support to the researchers.

3. The tool should not only be based on visualization semantics but also on the domain knowledge and dataset context.

4. The tool provides a way to visualize high dimensional datasets.

To fulfill these requirements successfully, we created milestones that need to be reached. The milestones were systematically established so that each milestone matched one of the above requirements. All milestones helped to: progress towards achieving the overall solution, and contributed to the knowledge of visualization science. We have listed the milestones of this project as follows:

1. The first milestone of any domain-based study is to understand the requirements of the target community. Based on that, one can plan to provide appropriate solutions. We had conducted surveys within the biodiversity community to reach this milestone, as discussed in Chapter 2.

2. The second milestone was to conceptually model the complete visualization system that fulfills all requirements mentioned above. This model should include the various steps to develop our domain knowledge-based visualization recommendation system. The details are provided in Chapter 5.

3. The third milestone was to combine the domain knowledge, acquired from our user's data domain with the area of information visualization. As mentioned by Tamara Munzner [Munzner, 2014], *the abstract task of understanding distribution, outliers, trends, correlation, etc. are extremely common reasons to use visualization. Each of the tasks can be expressed by very diverse terms using domain-specific language.* Processing domain specific terminologies related to visualizations by using machine learning models can infer these tasks from domain-specific texts. In our work, we have chosen a similar approach. This approach is described in detail in Chapters 6 and 7.

4. The fourth milestone was to provide a solution to visualize large datasets. This can be achieved by reducing the dataset's dimensionality to a few key variables that provide interesting insights. We designed a variable or feature selection algorithm to select the relevant and contextually interesting variables. The details about this algorithm are described in Chapter 8.

5. The fifth and final milestone was the quantitative and qualitative evaluation of our system. The qualitative evaluation of such a visualization system is based on the insights gained from the datasets. As recommended by Spence et al. [Spence, 2001], the visualization systems should also be evaluated on the overall perceived insights. In Chapter 10, we have provided details about our quantitative and qualitative evaluations.

Based on the identified problems, mentioned requirements, and milestones set in for project, the scientific questions we have answered in this thesis are:

1. **How can the domain knowledge be integrated into the developmental stages of visualization recommendation systems?**
2. **Does the integration of domain knowledge into visualization recommendation systems improve the overall dataset insight?**

The contributions of this thesis are:

1. **Domain knowledge-based Visualization Recommendation Model:** One of the core contributions of this thesis is the construction of our visualization recommendation model that is based on the biodiversity domain knowledge and the context of the data. As a result of our interactions with the community about visualization requirements, the visualization tool also includes an element of support for selecting the most appropriate visualization. The suggestions for visualization are based on the target community's domain knowledge.
2. **Biodiversity Visualization Text Classifier:** Another main contribution of this thesis is the design of a visualization classifier that, to our knowledge, is the first to be based on textual data. The classifier automatically selects one of the 15 visualization types for any given biodiversity specialized text. Irrespective of the conventional chart type recognition techniques based on image identification, the text-based visualization classifier is the first to recognize different chart types from the text. The need for this classifier resulted from the limited data input by the domain scientists. We used a machine learning-based approach for obtaining domain knowledge from visualization captions available in biodiversity publications. By combining this classifier with our visualization taxonomy, we generated the visual goals for our system.
3. **Context-aware Variable Selection Algorithm:** A context-aware variable selection algorithm reduces a large number of variables to the most interesting and relevant one. The candidate variables are selected based on the metadata's contextual and semantic properties. This ad-hoc algorithm can be created for any domain that provides the metadata alongside the dataset by following our context-aware variable selection workflow.

Chapter 5

Domain Knowledge-based Visualization Recommendation Model

"My particular ability does not lie in mathematical calculation, but rather in visualizing effects, possibilities, and consequences." - Albert Einstein

A conceptual model is an application model that the designers want users to understand [Johnson, 2007]. It enables a software designer to provide an overview of the solution without getting into many details. It precisely discusses the various components that constitute the development of the system. In visualization science, the visualization pipeline is a general model for a typical visualization process structure [Haber and McNabb, 1990]. Starting from data to be visualized and a particular visualization task at hand, several steps are processed along the visualization pipeline, including data enhancement, visualization mapping, and rendering, to eventually achieve visuals of the data to serve the given visualization task through effectiveness, expressiveness, and appropriateness [Schumann and Müller, 2013]. This chapter presents our visualization recommendation model based on integrating the knowledge from the biodiversity domain. This model provides an overview and the importance of various elements that constitute the development of our knowledge-based visualization system. Before defining our model, we first need to understand the specific terms related to the knowledge-based systems in the visualization science.

The terms data, information, and knowledge are often extensively used in an interrelated context [Chen et al., 2008]. In visualization, [Chen et al., 2008] untangles these definitions, not only in perceptual and cognitive space but also in computational space. In computational space, they define knowledge as *"data that represents the results of computer-simulated cognitive process, such as perception, learning, association, and reasoning or the transcript of some knowledge acquired by human beings"*. Others describe knowledge as a combination of data and information complemented with expert opinion, skills, experience, expertise, and accumulated learning [Rowley, 2007].

Visualization researchers have repeatedly called for the integration of knowledge with visualization [Wagner, 2015]. ‘Integration of prior knowledge in the visualization systems’ is listed as one of the ten unsolved information visualization (InfoVis) problems [Chen, 2005]. He argues that InfoVis systems need to be adaptive for user’s accumulated knowledge, especially domain knowledge needed to interpret results. In their discussion on the 'science of interaction', [Pike et al., 2009] declare 'knowledge-based interfaces' as one of seven research challenges for the coming years.

Knowledge-assisted or knowledge-based visualizations are defined as:

"Knowledge: Data that represents the results of a computer-simulated cognitive process, such as perception, learning, association, and reasoning, or the transcripts of some knowledge acquired by human beings." [Chen et al., 2008]

There are various ways by which one can apply the user's knowledge in a visualization system. For example: choosing variables, views or charts for visualization recommendation systems, colors within the visualizations, etc. Knowledge-based visualization systems' objectives include sharing domain knowledge among different users and reducing users' burden to acquire knowledge about complex visualization techniques [Chen et al., 2008]. However, [Chen et al., 2008] also list the following shortcoming of such systems: 1) it is difficult to know what knowledge to capture and the inconvenience of collecting knowledge in bulk from the experts, 2) it leads to the development of a system only at a specific application level. Keeping these limitations in mind, we developed our system based on the simple rule-based visualization reference model by [Heer and Agrawala, 2006].

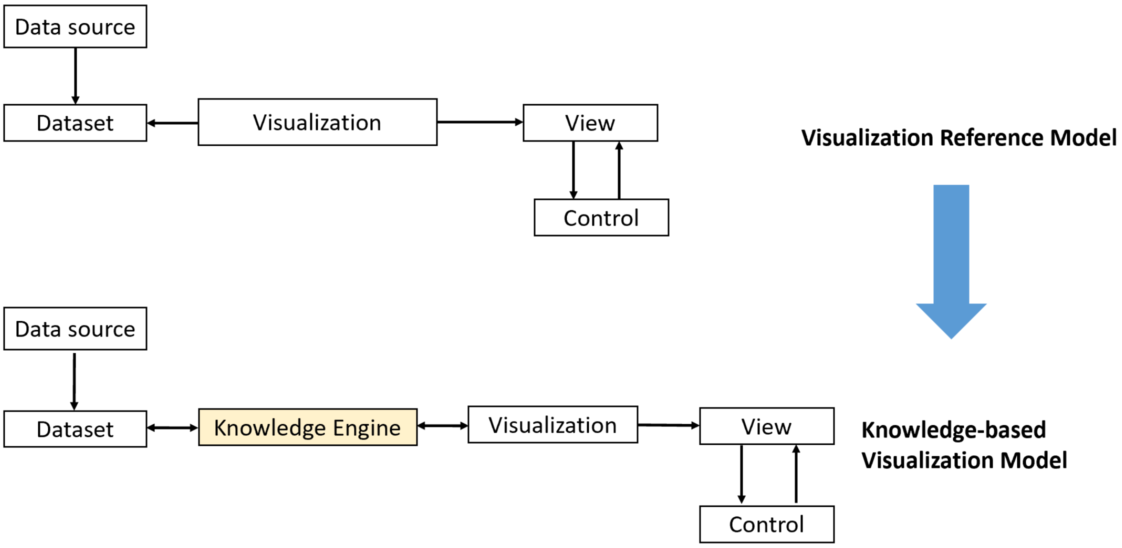
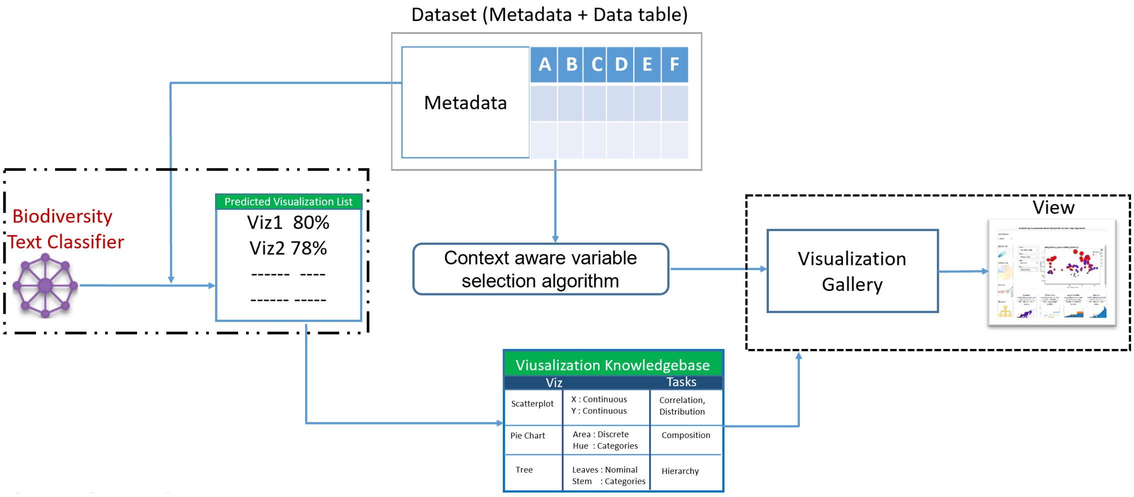


Figure 11: From a) the visualization reference model [Heer and Agrawala, 2006] to b) the knowledge-based visualization model.

The reference model (Figure 11(a) by [Heer and Agrawala, 2006]) provides a general template for structuring visualization applications that separate data models, visual models, views, and interactive controls. This separation of data and visual models enables multiple visualizations of a data source, separates visual models from displays to enable multiple views of visualization, and modular controllers to handle user input in a flexible and reusable fashion. As depicted in Figure 11(b), when we include our Knowledge Engine, the same rule-based visualization model becomes a knowledge-based visualization model. This Knowledge Engine is a central processor by which one can integrate the domain knowledge in any visualization system. Therefore without this unit, this model will lose the benefits provided by the domain knowledge in the system. For us, these benefits are to decrease the the variable space and provide domain-based data insights. This Knowledge Engine is a base of our Knowledge-based Visualization Recommendation Model.

5.1 Knowledge-based Visualization Recommendation Model

Figure 12 shows our visualization recommendation model, which uses the domain knowledge at its different stages.



Viusalization Knowledgebase

Viz		Tasks
Scatterplot	X : Continuous Y : Continuous	Correlation, Distribution
Pie Chart	Area : Discrete Hue : Categories	Composition
Tree	Leaves : Nominal Stem : Categories	Hierarchy

Figure 12: Domain Knowledge-based Visualization Recommendation Model

In the context of recommendation systems, we would like to divide the knowledge inclusion in two forms: data specific and technique specific. Data specific knowledge is the knowledge about the domain and the context of the data. In our model, Biodiversity Visualization Text Classifier and Context-aware Variable Selection Algorithm, include data specific knowledge in our system. Technique specific knowledge is about visualization specific technical knowledge: visual mapping, code to create the visualizations etc. In our model, Visualization Knowledgebase performs this task.

Biodiversity Visualization Text Classifier reads in the metadata from the dataset and classifies it into different visualization types. Based on the provided metadata, it performs the chart selection for the dataset. The classification is done based on the machine learning classifier trained on the abundance of visualization images and captions available in biodiversity publications. Detail about the construction of this classifier is provided in Chapter 6.

Context-aware variable selection algorithm (CVS) reads in the metadata. Based on this metadata's context and salient keywords from other metadata files within the project, it filters out the important variables. Thus it performs the variable selection for the dataset to be visualized with the charts selected by the classifier. This algorithm is explained in detail in Chapter 8.

Visualization Knowledgebase is composed of a visualization database and visualization conditions files. The visualization database is implemented in Neo4j[1] and is the database level view of the visualization taxonomy presented in Chapter 7. It gets the predicted ranked visualization list from the classifier and provides the ranked visual goal list. The detail about visual goals realization is also provided in

[1] www.neo4j.com

Chapter 7. The visualization conditions file reads in the visual tasks and the variables filtered by the CVS. Based on the data type of these variables, it filters the reasonable goals out. For example, for any network-related visualizations like alluvial diagrams, node-link diagrams, etc., it is essential to have at least two categorical variables. Only the overview goal has no restriction on a specific data attribute, as it provides a univariate analysis of a selected variable. The detail logic for the selection of the visual goals is provided in Table 5.1. Based on the data type of the variable, it ignores those goals which cannot be visualized. For example, ignoring the hierarchy visual goals when no categorical variables are present.

The output from all the above modules is input to the visualization gallery, which sends out the appropriate visual goals clubbed with related visualizations.

Table 5.1: Rules for the selection of visual goals

Visual Goals	**Rules**
Network	len(categorical)>=2 and (len(quantifiable)>=1 or len(discrete)>=1)
Composition	len(categorical)>=2 and (len(quantifiable) >2 and len(discrete) >=1)
Comparison	len(categorical)>=1 and (len(discrete)>=1 or len(quantifiable)>=2)
Hierarchy	len(categorical)>=1 and (len(discrete)>=1 or len(quantifiable)>=1)
Clustering	len(quantifiable)>=2 and len(discrete)>=1
Distribution	len(quantifiable)>=2 and len(discrete)>=1
Overview	all allowed

The rules presented in Table 5.1 are encoded according to the Python programming language. Here *len* is length. For example, *len(categorical)* means the number of variables that are of data type *categorical.* Symbol ">=" means either left expression is greater than or equal to the right expression or vice-versa. *and* and *or* are basic logical operators.

Each data variable is classified into three data attributes types. They are:

- **Categorical:** Categorical scales can be of type string or integer. They identify entities as belonging to mutually exclusive categories. Categorical columns are differentiated from other data columns or variables by the number of unique values. Though there is no one protocol on which this threshold of unique value is defined. However, for our application, we kept this value as 10. So, all variables which have ten or fewer unique values are categorical. For example, *leaves_dead* categorical column with 0 signifies the absence of dead leaves, and 1 signifies the presence of dead leaves. Same with variable *tree_type*, with different labels which are names of different tree types.

- **Quantifiable:** All quantitative variables are considered as a quantifiable scale. For our application, we have considered both ratio and interval scales as one. This distinction is not useful when designing a visual encoding system [Munzner, 2014]. Therefore we have merged both of them in one category. For example, temperature, height, weight, etc., are all considered as one type. Date format is also considered in this category.

- **Discrete:** Discrete is a numerical scale with a finite number of possible values and can only be expressed in whole numbers. Examples of such values are count, age, etc.

5.2 Recommendation Workflow

In the following, we have step-wise described the recommendation workflow within our system.

1. When a user selects a dataset, our Biodiversity Text Classifier gets activated and provides a list of all the suitable visualizations based on the metadata's text.

2. Then, the context-aware variable selection algorithm filters out the interesting variables from the dataset.

3. Based on the visualization knowledgebase and the filtered variables' data attributes, appropriate visual goals are sent to the user. Visual goals are explained in detail in Chapter 7.

4. Based on the user selected goal, the relevant visualizations are sent to the user interface along with the suitable variables that can be represented by these visualizations.

5.3 Summary

This chapter presents an overview of the solution to provide a visualization recommendation system for the biodiversity community. We have described our knowledge-based visualization recommendation model, which we have extended from the visualization reference model by [Heer and Agrawala, 2006]. The next chapters contain details about the construction of each component of this model.

Chapter 6

Biodiversity Visualization Text Classifier

**Part of this chapter is based on work published in the Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: IVAPP, 157-168, 2020, Valletta, Malta*

One of the core elements of our visualization recommendation system is to derive the visual goals from the datasets automatically. In our work, these visual goals are derived based on the domain knowledge and the context of the data. Our need to gather this domain knowledge in bulk has led to the creation of the biodiversity visualization text classifier. This chapter describes in detail the development of this classifier.

From our requirement analysis survey (Chapter 2), we found that a spectrum of different visualizations can represent a single domain-specific task. On the other hand, there are always typically one or two tasks prominent to each visualization (Table 2.1). Most of these tasks are domain-dependent tasks. For example, for a dendrogram, it is a phylogenetic analysis. For a scatterplot, it is spatial distribution along with other analytical tasks like PCA, Regression Analysis. It shows that each visualization type can be identified by its typical visual and domain-specific tasks and vice versa. The abstract task of distribution, correlation or so, can be expressed in very diverse terms using a domain-specific language [Munzner, 2014]. As these tasks are described in a human language, a text containing such terms can be identified into visualization types. Typically, a biodiversity dataset comes with a metadata file. These metadata files provide information about the what, why, when, and who of data and context, methodology, keywords related to the dataset, and research. Thus, these files are a good source of textual information that can provide dataset-specific visual goals when appropriately processed. Classifying this metadata information with an intelligent system that understands the domain and the visualization vocabulary can produce interesting visual insights into the data. To develop a system that can simultaneously understand the biodiversity and visualization tasks, we needed much training data.

As also mentioned in Chapter 2, several attempts to gather this training data from users failed. Therefore, we decided to gather this information from already published biodiversity literature. Reviewing the literature for eliciting possible domain tasks is a core data generation strategy in different visualization studies

[Kerracher and Kennedy, 2017, McKenna et al., 2014]. For developing a classification system that understands both visualization and domain-specific vocabularies, textual training data needs to be an amalgamation of a) terminologies from different visualizations b) domain-specific terminologies used in conjunction with the visualization terms. In scientific publications, visualization captions contain such information. For our training data, we extracted the following information content from the biodiversity publications: a) visualization or chart images, by which we can identify the chart type in it, and b) the related visualization image captions. Visualization captions provide multitudes of information: a) representative goals of the author which are not directly visible from the image itself, b) domain-specific tasks, depicted through the chart, and c) chart layout and characteristics (e.g., text, colors, lines, shapes). Consider, for instance, Figure 13 and the original caption to the visualization taken from [Moody and Jones, 2000].

"Fig. 5. Boxplots comparing the distribution of the measured soil variables at the different canopy positions at trunk, midcanopy, the canopy edge, and outside the canopy, respectively. The upper and lower boundaries of each box represent the interquartile distance (IQD). The horizontal midline is the median value. The whiskers extend to 1.5x IQD. Outliers are displayed as horizontal lines beyond the range of the whiskers. If the notches of any two boxes do not overlap vertically, this suggests a significant difference at a rough 5% confidence interval."

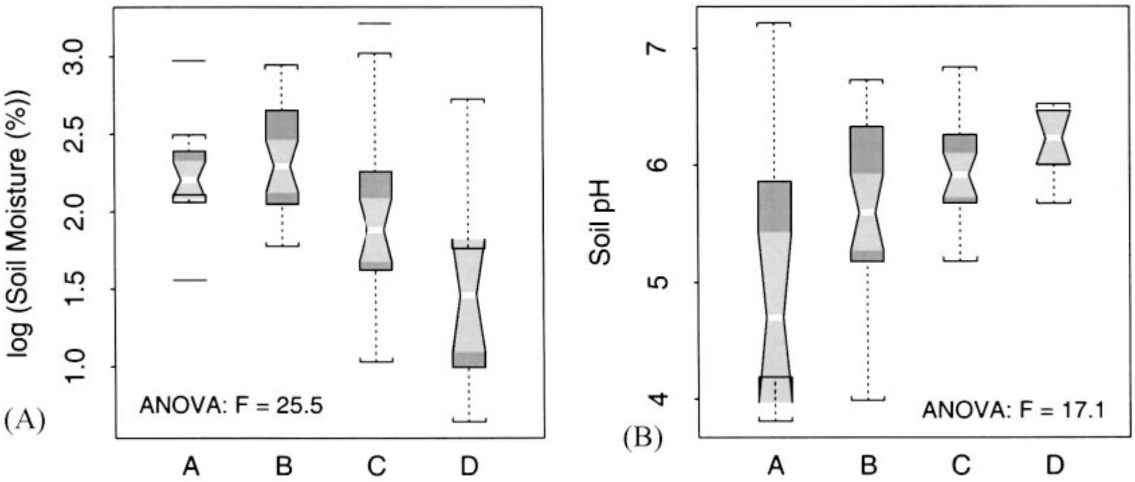


Figure 13: Example image adapted from "Soil response to canopy position and feral pig disturbance beneath quercus agrifolia on santa cruz island, california" by A Moody and J. A. Jones, 2000, Applied Soil Ecology, 14(3):269 – 281.

The caption of this figure provides clues about the following information:

Chart type: boxplots, box, horizontal midline, whiskers, horizontal lines, notches.

Representative goals: comparing, distribution.

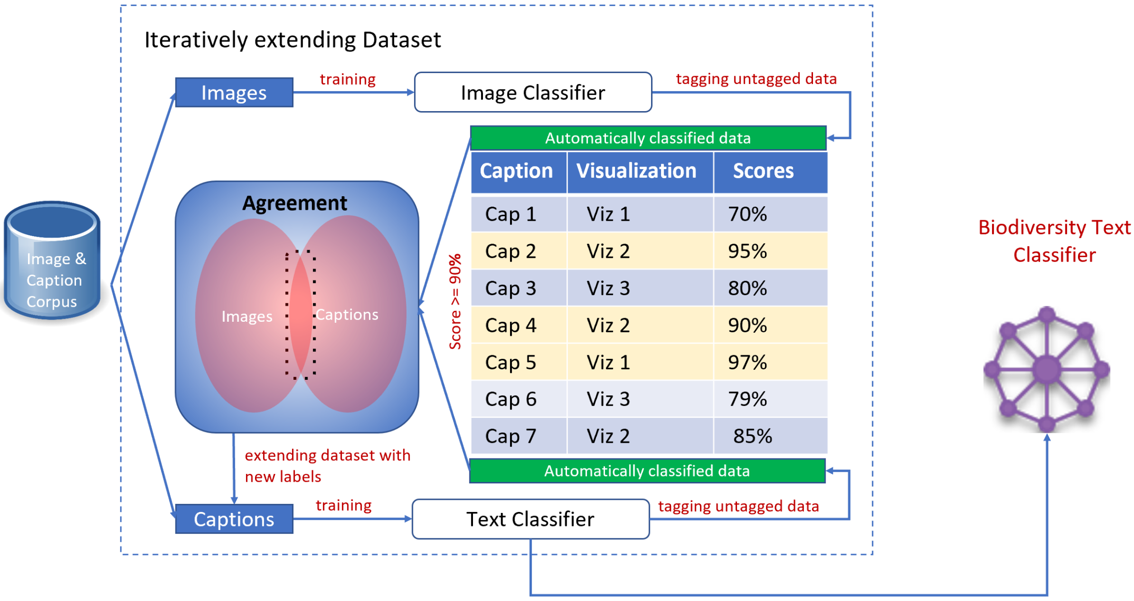
Domain specific variables: soil variables, canopy positions, trunk, midcanopy, canopy edge.

Statistical or analytic information: interquartile distance (IQD), median value, confidence interval.

If a system knows this information for each chart type, then with the provided biodiversity text, one can infer different chart types. Once chart type is known, then using a visualization task taxonomy, visual goals can be inferred. This idea leads to the foundation of the creation of the very first biodiversity visualization text classifier. As we are unaware of any past studies on visualization classifier based on text, we can claim it to be the first visualization text classifier in the visualization research. Previously, visualization or chart type classification or recognition has only been done based on chart images [Savva et al., 2011, Balaji et al., 2018].

6.1 Classification Process

The process of creating the biodiversity text classifier consists of a sequence of complex steps, visualized in Figure 14. The first step was to manually create a starting dataset, that associates caption texts with their respective chart types. This set is then incrementally extended using a combination of image and caption classification techniques, in order to gain the highest possible quality on the automatic labeling of unlabeled data. The resulting dataset is then used as training set for the biodiversity text classifier, that can be integrated into the biodiversity knowledge-based visualization recommendation system.



Caption	Visualization	Scores
Cap 1	Viz 1	70%
Cap 2	Viz 2	95%
Cap 3	Viz 3	80%
Cap 4	Viz 2	90%
Cap 5	Viz 1	97%
Cap 6	Viz 3	79%
Cap 7	Viz 2	85%

Figure 14: Workflow for the creation of the biodiversity visualization text classifier

6.1.1 Data preparation

In our data collection process, first we selected reputed biodiversity journals representing different biodiversity sub-domains. The breakdown of the downloaded publications is shown in Appendix I.

We had downloaded all available volumes and issues of these journals till 2016, which is the year when this download was done. For creating the initial dataset, we downloaded 26 588 biodiversity publications through Elsevier ScienceDirect article retrieval API[1], which allows the download of a complete publication in an XML format. From these 26 588 downloaded publications, 96 837 images and their captions were extracted using a python script.

6.1.2 Class formation and annotation process

Out of 96 837 image and caption samples, we created our training data by randomly selecting a subset of 4 073 visualization image captions and labeling them with their respective visualization types. Due to the sheer richness of different visualization types – a closer study revealed the sample to contain 59 different visualizations (see Appendix C) – we continued our annotation process in the following stages:

Class grouping: In order to gain adequate sample sizes for each of the visualization types or classes, we split/merged the original 59 classes into super/sub classes:

> 50 **samples in class** Since we considered 50 examples to be sufficient for classification, all classes with same or more examples were kept as super classes.

< 10 **samples in class** These classes had very small set of examples and were not suitable match for our super classes. Therefore, they were rejected from the further annotation process.

all other classes All classes, that do not figure frequently enough to suffice for the classification task have been merged into super classes either based on their visual similarity or their representational goals. For example, chart types that use the same coordinate space (e.g., xy plot) and same visual marks (e.g., bars) were considered visually similar and then were merged. This way, all the chart types which are visually similar to Column Chart e.g., Bar Chart, Stacked Bar Chart, Multiset Bar Chart etc. were all merged into the super class ‘Column Chart’. On the other hand, Chord Diagrams, Alluvial Diagrams and Network Diagrams are visually dissimilar but have the common representational goal of connecting entities. Thus, they all were grouped in the class Network’. All non-visualization images (e.g., camera-clicked pictures, conceptual diagrams etc.) were grouped into the ‘NoViz’ class. Due to the variant structure of non-visualization images, ‘NoViz’ class was also excluded from the image classification process. An overview of retained classes is provided in Appendix D.

Doing so, we ended up with 15 different super classes.

Assignment of classes for caption classification: Once, we had formed the classes, we did another round of annotation. We have now labeled our selected corpus of 4 073 captions with these 15 classes. For detailed information about these classes, refer to Appendix E.

Assignment of classes for image classification: For creating a training set for image classification, we had to ignore the visually similar classes. Histogram is visually similar to the column chart and timeseries is visually similar to the line chart (see Appendix E). Thus, histograms and timeseries were ignored from the image classification process. Alongside, due to the variant structure of non-visualization images, ‘NoViz’ class was also excluded from the image classification process.

[1] `https://dev.elsevier.com/sciencedirect.html#/Article_Retrieval`

We have provided the frequency distribution of classes for image and caption classification in Table 6.1. In Appendix E, we have shown examples for each class, that consist of the replicated original image and caption from open-access publications. Due to copyright issues, we are unable to provide original examples from our dataset.

Table 6.1: Frequency distribution of our manually annotated training dataset for caption and image classification.

Classes	**Caption Classes**	**Image Classes**
Ordination Plot	503	278
Map	529	277
Scatterplot	399	272
Line Chart	320	283
Dendrogram	282	243
Column Chart	427	302
Heatmap	147	124
Boxplot	210	104
Area Chart	159	95
Network	58	32
Histogram	57	-
Timeseries	319	-
Noviz	511	-
Pie Chart	-	134
Proportion	157	-
Total	4073	2144

6.1.3 Image classification

For image classification, we have used Convolutional Neural Networks (CNNs). CNNs are a specialized kind of neural networks for processing data that has a known grid-like topology. Since images can also be thought of as 2D-grid of pictures [Goodfellow et al., 2016], CNNs have been tremendously successful in application to image data.

For training, we have used reusable pre-trained neural network modules provided by TensorFlow Hub[2]. TensorFlow Hub is a library for the publication, discovery, and consumption of reusable parts of machine learning models. Each TensorFlow Hub module is a self-contained piece of a TensorFlow graph, along with its weights and assets, that can be reused across different tasks in a process known as transfer learning. Out of the available CNN modules in TFHub, we chose MobileNet_V2 [Sandler et al., 2018] as our CNN architecture. MobileNet_V2 is a family of neural network architectures for efficient on-device image classification and related tasks. MobileNet has achieved a similar accuracy to VGG-16 [Simonyan and Zisserman, 2014] using far fewer parameters on ImageNet dataset [Russakovsky et al., 2015]. MobileNet_v2 module of TensorFlow Hub contains a trained instance of the network, packaged to do the image classification. This TensorFlow Hub module uses the TensorFlow Slim implementation of 'MobileNet_v2' with a depth multiplier of 1.0 and an input size of 224x224 pixels. For

[2] www.tfhub.dev/google/imagenet/mobilenet_v2_050_96/feature_vector/2

training the classifier, we used Keras[3] with TensorFlow [Abadi et al., 2016] backend. To train the classification network on our data, we resized the images to a fixed size of 224 x 224 x 3 and normalized them before feeding them into the network. We used Adam optimizing function [Kingma and Ba, 2014] with the learning rate of 0.001. We have trained with the default batch size of 32 for 60 epochs. This network was trained on Intel Core i7 - 8550U CPU 1.80GHz with 16GB memory for approximately 1 hour.

6.1.3.1 Results from image classification

For evaluation, we have used Keras' in-build evaluation function. When provided with suitable parameters, Keras separates and retains a portion from the training data and then uses that unseen retained data for evaluating the model. For evaluating our image model, 20% of the examples from the image dataset were retained from training. Our model has achieved a classification accuracy of 75% on an automatically selected batch of 100 images. Then this classifier was used to classify the original corpus of 96837 IDs. Our image classifier was able to annotate 54%, i.e., 52921 IDs out of 96837 IDs, with confidence of 95% and more.

6.1.4 Caption classification

The 4 073 manually labeled image captions served as training set for the initial supervised classifier. In order to be able to optimize the classifier for each of the identified classes separately, we decided to build binary classifiers, that can distinguish one specific class from all others. From these specialized binary classifiers, an assembly classifier is constructed (see Figure 15, Training Step). Given an input, the assembly asks each classifier to process the input separately (as detailed in Figure 15, Classification Step), and receives a probability score that states how likely it is, that the given sample is of the respective class. The classes of all classifiers, that give a positive response with a certain preset confidence (in our case usually 90%), will then be returned as result vector.

To find out which binary classifiers to incorporate into the assembly, we implemented and optimized three standard classifiers in text classification [Joachims, 1998, Sebastiani, 2002]: Support Vector Machines (SVM), CNN and Random Forests. SVMs [Cortes and Vapnik, 1995] are inherently binary supervised learners. In their linear form, they find the maximum-margin hyperplane in data space that best separates the data points of one class from the data points of the other class. Kernels [Boser et al., 1992] have been introduced to generalize the principle to polynomial, radial or sigmoid functions. Random Forests [Ho, 1995] are assemblies of a – usually rather large – number of Decision Trees that contribute to the main decision in form of a majority vote. Additionally, in resemblance to the image classifier, a neural network solution – specifically a multilayer perceptron classifier with the same stochastic gradient-based optimizer as the image classifier and the same constant learning rate of 0.001 – was used.

[3] www.keras.io/api/

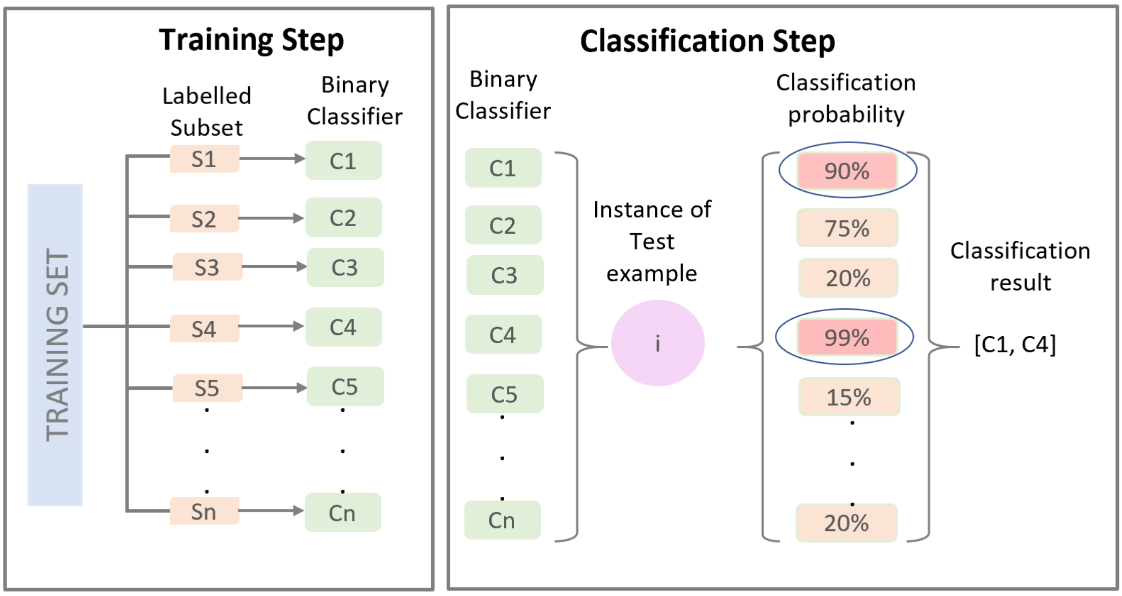


Figure 15: Caption classification process

6.1.4.1 Preprocessing

As is standard in natural language processing [Aggarwal and Zhai, 2012], the labels have been broken into tokens, stemmed and stop words have been removed before processing them. Additionally, some standard phrases that have been identified during manual n-gram evaluation of the data and are unrelated to the contents of the image, like phrases to make people aware of the modalities of the online version of the paper – e. g. *"For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article"*, – have been automatically removed. In order to keep the training data as pure as possible, captions referring to multiple visualizations were filtered out, leaving a dataset of 4 066.

The resulting word vectors contain the term frequency - inverse document frequency (tf-idf) scores [Ramos et al., 2003] per word.

6.1.4.2 Model optimization or parameterization

Each binary classifier has been trained separately on a data set consisting of all samples of the target class and an equal number of samples uniformly distributed over all other classes. Classifiers have been evaluated using a 5-fold-cross-validation, that splits the data set into 5 equal parts training on 4 parts and testing on the last. The final evaluation result constitutes as the average of all five runs.

In order to reach the best results, we optimized both the pre-processing of the data as well as the parameters of the classifier itself. On data level, we optimized the maximum size of the vocabulary, the minimum number of documents each word figures in and which n-grams should be included into the analysis. Applying an exhaustive grid search over the range of sensible parameters for each feature (vocabulary size: [250 to 1250 (steps of 250)], minimum document frequency: [0 to 4], n-grams: [1 to 6]), we achieved the best results using a base vocabulary that consists of the 750 most important words and 2-grams, that occur in at least 3 documents in the whole corpus.

We also optimized SVM for its kernel function (linear, polygonal, sigmoid, and

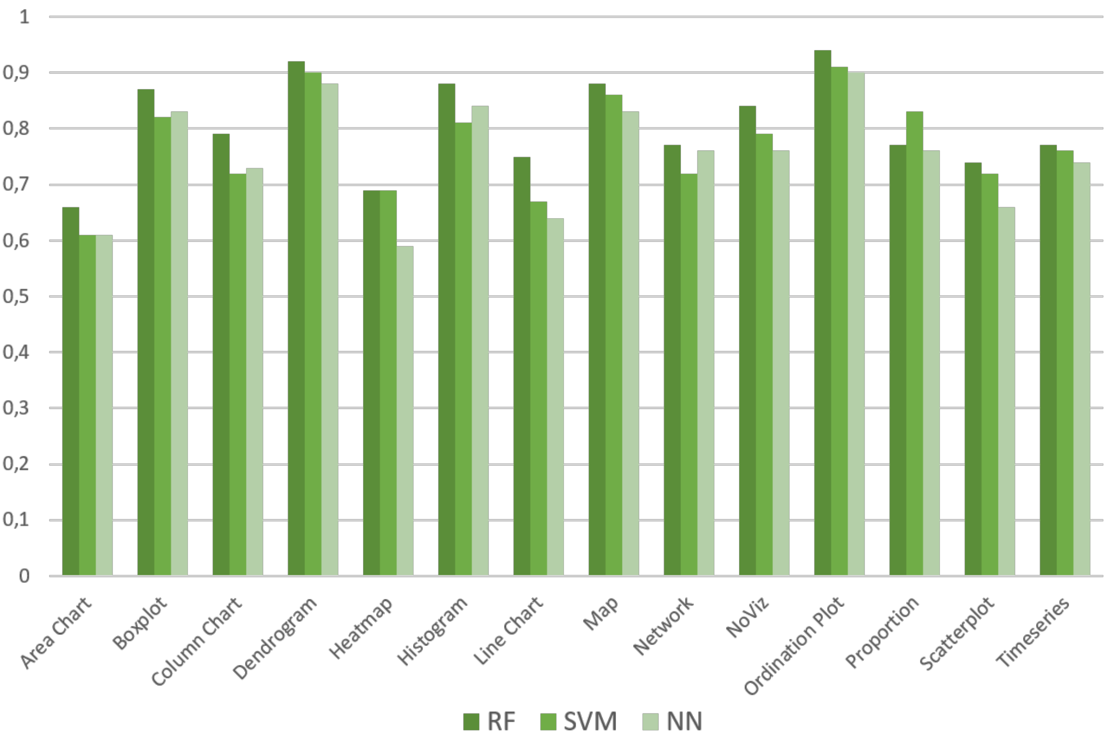


Figure 16: Classification results (F1 score) of the Random Forest (RF), Support Vecor Machine (SVM) and Neural Network (NN) classifiers for each class in our corpus.

radial basis function), finding a linear kernel to give the best results, and the Random Forest classifier for the number of Decision Trees in the assembly (100 to 2000 in steps of 100), finding that the impact on the classification accuracy is rather small. The neural network has been tested with different node sizes in its hidden layers (2, 10, 15, 50). The best result has been achieved with 15 nodes.

Figure 16 shows the best results on each class of the classifiers. The results show that Random Forests outperform the results of SVM and the neural network in all classes with up to 7% increase in the F1 score. One possible conclusion to draw from these results is that, the linguistic properties of caption data can be modelled more precisely through a series of parallel boolean operations than through a maximum-margin method. Following this finding, we will use Random Forests as classifier for the assembly of binary classifier.

6.1.5 Incremental learning and caption dataset extension

The purpose of the caption classification is twofold. First, we want to extend the existing dataset to reduce the risk of overfitting the single binary classifier. Second, to train the binary classifiers to understand the words and phrases describing the underlying data of the chart in order to finally recommend chart types based on data set descriptions.

Incremental learning and an additional agreement step with the classification result of the image classifier (see Figure 14) was used to increase the size of the original 4 066 captions to a dataset of 22 881 captions.

One iteration of the incremental learning algorithm includes the following steps:

Learning: Conduct 5-fold cross-validation on the current dataset to evaluate the quality of the set (results see Figure 17). Train all binary classifiers on the whole dataset.

Annotation: Use the assembly to label as many captions of the remainder of the untagged data as possible with at least 90% confidence. Include new labels and captions into the extended dataset.

The two steps are repeated until a finishing criterion is met. Since we were focusing on extending the dataset in this phase, we stopped the algorithm when the number of newly included tags fell underneath a preset threshold (0.01% of the whole corpus in our case). This way, the caption classifier was able to annotate 44% of the total corpus which amounts to 43 256 IDs with a confidence interval of 90%.

6.1.6 Refining the knowledgebase

In order to ensure highest quality in the creation of our knowledge base, we refined the resultant data from image and caption classification in a multi-step process:

- We started with 52 921 labeled images in Image Corpus (IC), and 43 256 labeled captions in Caption Corpus (CC).

- To get only the visualization image IDs, first we removed the 'NoViz' labelled IDs from the caption corpus. Leaving behind 43 256-451= 42 805 to be merged. After the merging process, these IDs were put back to the corpus for iterative learning.

- We merged the two corpora by only keeping those IDs that have been unanimously tagged by both classifiers. This set contains a total of 22 817 common IDs.

- This set was then reduced to only contain the most reliable ID/label pairs:

  1. ID/label pairs with full or partial agreement in classified labels from image and caption classifier (11 108 samples). Partial agreement is reached if the label given by the image classifier is contained in the class list provided by the caption classifier; full agreement is reached if the caption classifier only provides one label and this label matches the class of the image classifier.

  2. ID/label pairs with more than 98% confidence from image classifier (10 728 samples).

  3. ID/label pairs whose classes were absent in image classifiers (Area Chart, Time Series, Histogram, Proportion), if the source of disagreement between image and caption classifiers stems from these classes, like 'Timeseries' in CC and 'Line Chart' in IC or 'Histogram' in CC and 'Column Chart' in IC. All other conflicts have been resolved manually. In our manual verification, 'Proportion' has performed bad due to its similar vocabulary with 'Pie Chart' and all other classes that represent some 'Proportion' or 'Composition' representation goals, for example different stack chart

types: Stack Area or Stack Column. To avoid such confusions for incremental learning round, we had to merge some of these example to ‘Pie Chart’, ‘Stack Area Chart’ and ‘Column Chart’. Rest of the examples were ignored.

4. ID/label pairs that have been manually checked upon due to the multi-assignment of the caption classifier were assigned the single true class if possible. Captions representing multiple visualizations have been rejected.

This leaves us with 22 248 high-quality ID/label pairs.

- Finally, the automatically created dataset has been merged with the manually annotated dataset to further increase the quality and size of our knowledge-base (22 866 samples in total, 1 468 Ordination Plots, 4 989 Maps, 1 669 Scatterplots, 6 173 Line Charts, 452 Dendrograms, 5 459 Column Charts, 603 Heatmaps, 303 Boxplots, 99 Area Charts, 187 Network Diagrams, 69 Histograms, 330 Timeseries, 448 Noviz, 304 Pie Charts and 313 Stack Area Charts).

6.2 Results and Discussion

6.2.1 Results

Figures 17 and 18 show the development of the quality of the classifiers as well as the number of samples for each label over the course of the 41 iterations necessary to reach the ending criterion (a tag rate of less then 0.01 % of the unlabeled samples of the corpus). In most cases, the quality of the classifiers rises the most within the first 3 iterations. After that phase, most classifiers do not change in quality any more. Exceptions are the line chart, with a drop after the steep rise in the beginning, the time series, with a drop at the eleventh iteration, and the histogram and area charts that fluctuate around 80% accuracy. The drops in the performances of both line chart and time series classifiers coincide with steep rises in the numbers of examples for the respective classes, suggesting that the classifier needed some iterations to adapt to the new dataset. The fluctuations in the quality of histogram and area chart classifiers stem from the small sample sizes for the respective classes.

Figure 19 shows which classes have been mixed up by the final classifiers, that have been trained on the entire resulting caption dataset before the agreement step. Rows represent the classification results, while the columns shows the names of the actual class of the misclassified sample. For example, the row denoted with N (Network) shows only zeros, meaning that no samples have been wrongly classified as network in this run. The most consistent confusions can be seen between boxplots and column charts, and maps and pie charts. The confusion between boxplots and column charts could stem from the presence of error bars in boxplots and a special type of column charts. The confusion between maps and pie charts could be explained with the presence of certain visualizations where pie charts were overlaid on the maps. Another similar cluster can be seen between pie charts and stacked area charts. The reason could be because both visualizations share a similar representation goal as ‘Proportion’ and the division of some examples from ‘Proportion’ into these two charts at the previous stage (see subsection 6.1.6).

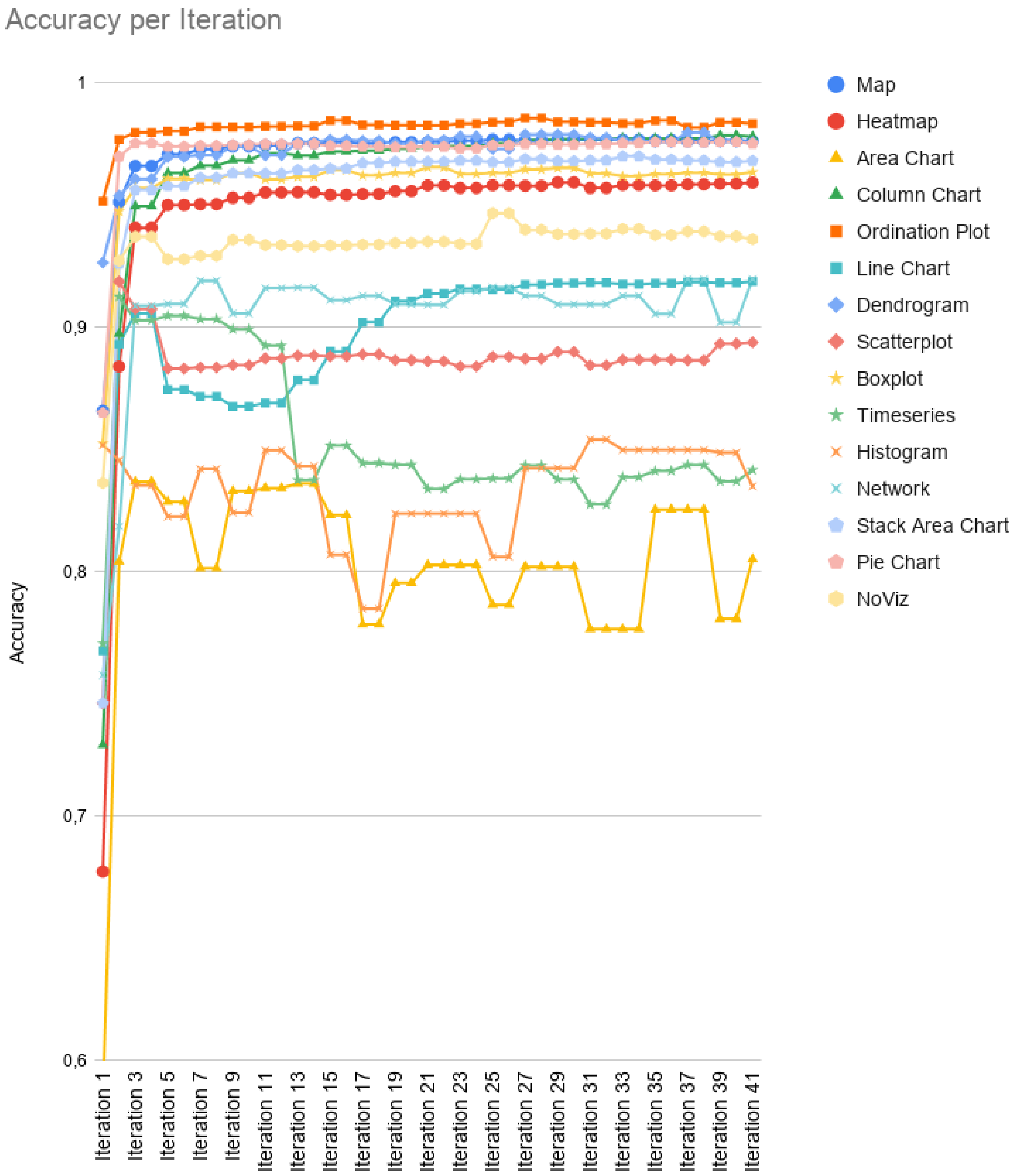


Figure 17: Line graph of the development of the accuracy of each binary classifier during the iterative learning phase. Notably, even though most classifiers began with classification accuracy of less than 80%, almost all of them increase their accuracy drastically after the first five iterations.

6.2.2 Reasons for misclassifications

In our extensive study of the misclassified cases, we extracted several reasons for such cases.

Mixed vocabulary from different chart types: The main reasons for this problem were a) often, multiple different visualizations are used in conjunction in one image, showing, for example, pie charts on different locations on a map. We have observed that the classifier could not perform well on those image cap-

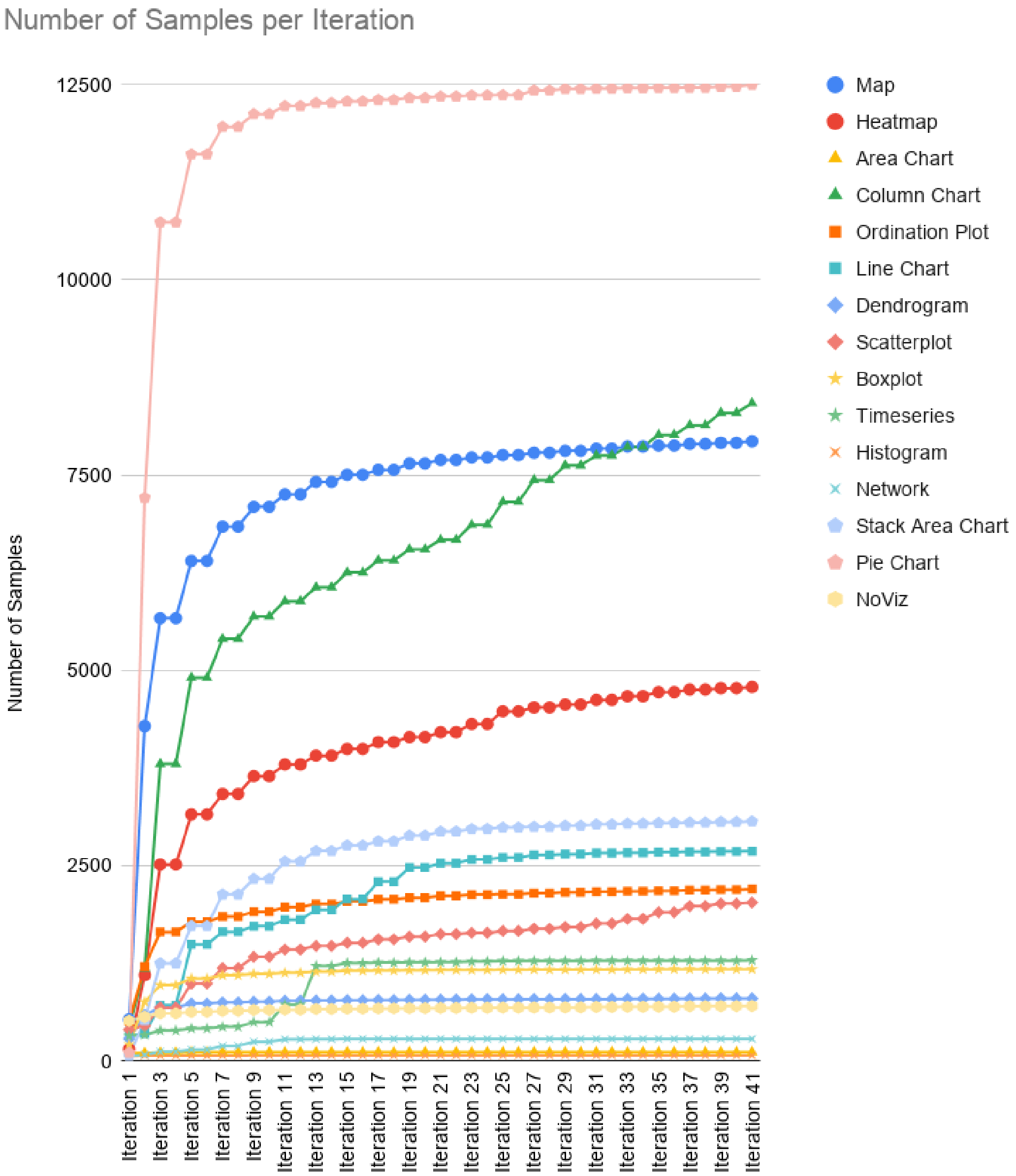


Figure 18: The development of the sample sizes for each class during the iterative annotation. Similar to the increase in accuracy of the binary classifiers in Figure 17, the numbers of sample sizes increase very quickly within the first few iterations.

tions, as the information about multiple chart types in the same text seemed to offer conflicting clues. b) In some images, multiple different visualizations are used to represent multidimensionality of the results. For example, the use of scatterplot for showing the distribution of some species and in the same image use of column chart for illustrating the comparison with other species. Although all efforts were made to remove such instances from our training set, however, we can't deny their existence in the rest of the corpus.

Similar representational goal: Histograms, boxplot and scatterplot share similar goal of showing distribution among continuous variables. Where, histo-

Table 6.2: Scores from incremental learning

Classes	Accuracy
Ordination Plot	0.98
Map	0.97
Scatterplot	0.89
Line Chart	0.91
Dendrogram	0.97
Column Chart	0.97
Heatmap	0.95
Boxplot	0.96
Area Chart	0.80
Network	0.91
Histogram	0.83
Timeseries	0.84
Noviz	0.93
Pie Chart	0.97
Stack Area Chart	0.96

gram shows the frequency distribution of a variable, boxplot provides detail information about this distribution among different quartiles. Then, scatterplot shows relationship and causation of this distribution with other variable/s. Unfortunately, although the visual representation is different, the language describing both visualizations tends to use similar wording, likely causing misclassifications.

Mixture of definition/description and interpretation A caption can be used to fulfill different tasks: define/describe the contents and/or interpret them. As the language differs very heavily from one task to the other and the ratio between definitions and interpretations varies from sample to sample even within a given class, a classifier might be drawn to either specialize in the definition/interpretation parts of the samples (high precision, low recall) or generalize to a point that it cannot exclude other classes (low precision, high recall).

Level of abstraction of some classes: Due to limited examples for some of the classes, we had to form superclasses of visualization types. For example, 'Column Chart' class is created by merging examples from 14 visualizations subclasses. This leads to the source of confusion among other classes. For example boxplots are confused with column charts due to the presence of error bars in certain types of column chart.

Wrongly mentioned visualization types: In addition to the regular vocabulary, the binary classifiers also look for their specific visualization name in the caption texts. Unfortunately in some captions, wrong visualization names are referred, mistaking for example a column chart for a histogram.

6.2.3 Comparison

Table 6.2 provides individual scores for different classes. Due to the special goal and characteristics of our study, currently we do not have any base study to compare our

	A	B	C	D	HM	HG	L	M	N	NV	O	P	S	SA	T
Area Chart (A)	0	0.20	0	0	0.20	0	0.20	0	0	0	0.20	0	0	0	0.20
Boxplot (B)	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Column Chart (C)	0.03	0.39	0	0.08	0	0	0.08	0	0.03	0	0.05	0.05	0.13	0	0.16
Dendrogram (D)	0	0	0	0	0	0	0	0	0	0.40	0.20	0.20	0	0	0.20
Heatmap (HM)	0.03	0	0	0	0	0	0.03	0	0.03	0.03	0.05	0.30	0	0.49	0.05
Histogram (HG)	0	0.50	0	0	0	0	0	0	0	0	0	0	0.50	0	0
Line Chart(L)	0	0	0	0	0.04	0	0	0.13	0	0	0	0	0.50	0	0.33
Map (M)	0	0.03	0.03	0.06	0.03	0	0	0.03	0	0.03	0.10	0.45	0	0	0.16
Network (N)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NoViz (NV)	0	0.10	0	0.10	0.10	0	0	0.20	0.10	0	0.30	0.10	0	0	0
Ordination Plot (O)	0	0.29	0.14	0.29	0	0	0	0	0	0.14	0	0	0.14	0	0
Pie Chart (P)	0.03	0	0.08	0.03	0.38	0	0.03	0.28	0.03	0.03	0	0	0.05	0.08	0
Scatterplot (S)	0	0.08	0	0	0	0	0.54	0.08	0	0	0.13	0	0	0	0.17
Stack Area Chart (SA)	0	0.06	0	0	0.17	0	0	0.11	0	0	0.06	0.56	0.06	0	0
Timeseries (T)	0.10	0.10	0.10	0	0	0	0.20	0.20	0	0.20	0	0	0.10	0	0

Figure 19: Distributions of the false positive cases over their actual classes. Each row contains the distribution for one binary classifier. The values run from 0.0 to 1.0 and sum to one per row, where 0.0 is no false positive case. For visual clarity, values more than 0.20 are shown in the white font.

results with. None of the previous studies have considered both aspects of charts (visuals from images and chart semantics from captions) for chart classification. In Figure 20, we have provided the comparison among scores from common classes in 3 different studies. In this figure, Revision refers to [Savva et al., 2011], ChartSense refers to [Jung et al., 2017] and DocFigure refers to [Jobin et al., 2019]

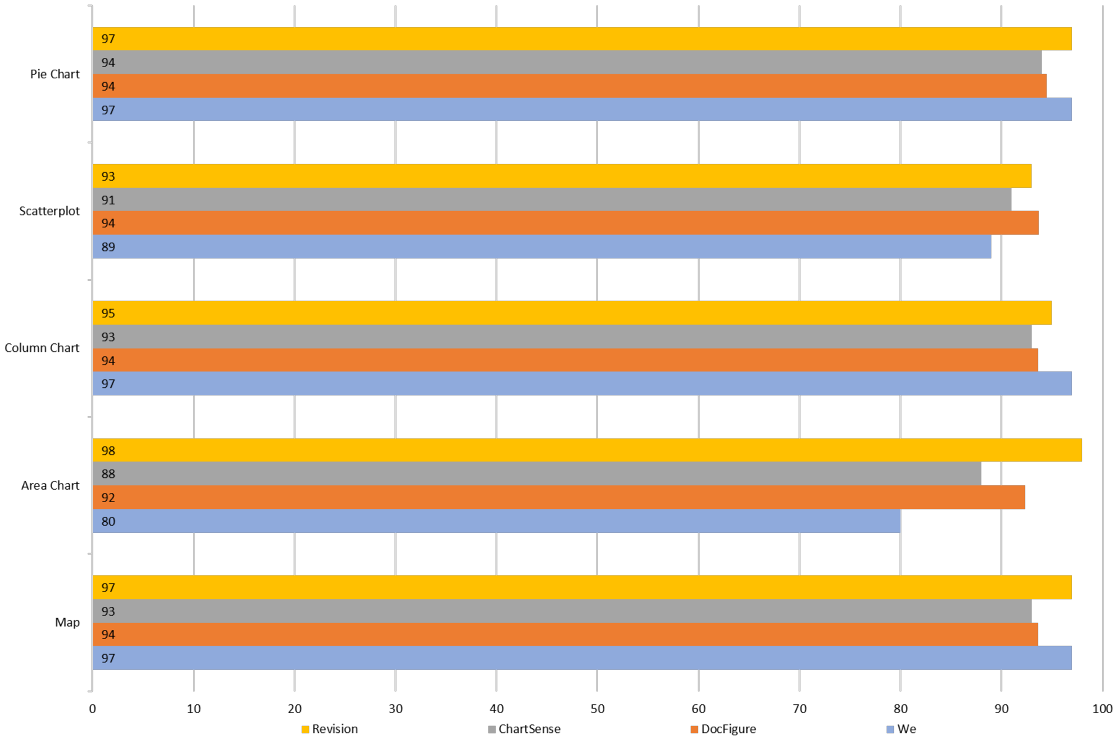


Figure 20: Comparison with other studies

Figure 20 shows that in comparison to other studies, we are only lacking in two classes i.e Scatterplot and Area Chart. In our work, Ordination Plot and Stack Area

Chart which is similar to Scatterplot and Area Chart respectively, were considered as a separate class based on their representative goal and visual dissimilarities. No such fine differences were made in the other studies. Scores of Ordination Plot is 98% and Stack Area Chart is 96%, and if we compare them with the other studies, then our performance is better. With an average accuracy (F1-score) of 92.2% we have proved that our approach of chart classification is better than only chart image classification.

6.3 Summary

In this chapter, we have presented one of the key contributions of this thesis i.e., biodiversity visualization text classifier. To gather the domain specific visualization vocabulary, we had to classify the chart types in the biodiversity publications. To do so, along with the visual similarity of different chart types, we have also considered the charts' conceptual similarities. We have manually labeled the chart images and captions from biodiversity publications. We have trained both the image and chart classifiers on this data. From the best results of these two classifiers, we have incrementally trained our assembly of text classifiers. Doing so, we have achieved an average F1-score of 92.2% from assembly of binary caption classifiers. In chapter 10, we have presented the qualitative evaluation of this classifier. Our result proves that conceptual/semantic chart classifier can efficiently differentiate between those chart types which are visually similar and are as efficient as image classifier. Due to the conceptual understanding of such classifiers, they can be used as a domain knowledge source for knowledge-based visualization studies.

By using visualization taxonomy and the visualization predictions from this classifier, a system can infer the visual goals from the domain specific textual data. In the next chapter, we have described this visual goals generation workflow.

Chapter 7

Visual Goals Generation

**Part of this chapter is based on work published in the Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: EuroVis, 29–31, 2017, Barcelona, Spain*

In the previous chapter, we have presented our biodiversity visualization text classifier that can classify a biodiversity text into different visualization types. Each visualization type illustrates some abstract visual tasks [Munzner, 2014]. To know these visual tasks or goals for different visualization types, one needs to have a visualization knowledge source or a taxonomy. Visualization task taxonomy maps different visualization types to their respective visual goals. This chapter presents our visualization taxonomy, to get the visual goals from the classifier's predicted visualization list.

Visual goals generation refers to obtaining a set of analytical tasks to be performed on the visualizations. Understanding which tasks or goals an analyst wishes to carry out is a non-trivial problem [Kerracher and Kennedy, 2017]. In a typical design scenario, [Wijk, 2006] noted that visualization researchers must spend time and effort bridging 'the knowledge gap' between themselves and the domain expert. In reviewing the literature, [Kerracher and Kennedy, 2017] found the most prevalent approach to task generation involved literature-based methods. As it was impossible to collect the needed mass knowledge directly from the biodiversity community, we obtained these tasks from already published visualizations in the biodiversity literature for our visualization recommendation system. We did this by using visualization caption classification, which resulted in the creation of biodiversity visualization text classifier. We have already discussed the creation of this classifier in Chapter 6. The produced classifier understand the vocabulary of high-level visual goals and domain-level goals, as shown in Figure 21.

The word clouds in Figure 21 are generated from the top few words of our four binary classifiers' vocabulary. These images show that the classifiers have learned high-level visual goals and the domain goals within each visualization. For example, for Dendrogram, high-level goals are clustering and hierarchy, and some of the domain level goals are upgma or phylogenetic analysis. Thus, these biodiversity visualization text classifier (an assembly of binary chart classifiers), automatically map the high-level visual goals with the domain goals and provide relevant visualizations based on the provided text.

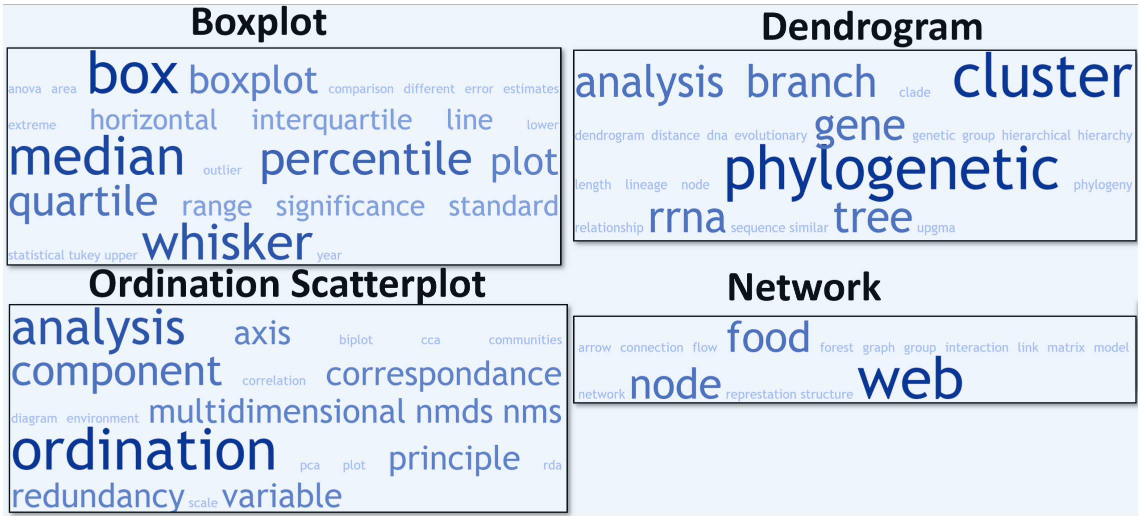


Figure 21: Word clouds show the prominent keywords in the four visualization text classifiers.

7.1 Visualization Taxonomy

Once we get the visualizations list from our classifier, we map them with their high-level visual goals using our visualization taxonomy. We gathered these visual goals while manually annotating the captions for the visualization text classifier and doing some literature studies.

However, we have seen that these domain-independent information-seeking goals (high-level visual goals) are very generic and are used in many different studies ([Roth and Mattis, 1990, Amar et al., 2005, Schulz et al., 2013]). Different visualizations represent these goals differently. Consider a goal to show distribution. If a user is interested in viewing distribution over the spatial scale, distribution maps are suitable. If she is interested in the distribution among two variables, then scatterplots are suitable. Whereas if she is interested in finding further patterns from the data distribution and especially for multiple variables, this might lead to further sub-goal of either correlation or clustering. Scatterplot matrices or parallel coordinates represent the correlation among multiple variables. Ordination plots or dendrograms and their variants represent clustering. Moreover, one visualization can represent more than one user goal depending upon the type of the data. For example, a treemap can represent hierarchical levels and part-to-the whole composition relationships when provided with categorical data. The advent of different visualizations has made it possible to classify the visualizations according to the data and the user goals they represent.

This work is inspired by a mind map presented by Andrew V.Abela[1]. Here, the author has presented the four main user goals, which he has decomposed into different data domains to suggest the visualizations. This mind map covers limited user goals and limited data domains. However, it provides us a clue of hierarchically decomposing the goals into subgoals and data domains for effective data visualization suggestions. In contrast to, e.g., [Nusrat and Kobourov, 2015], which focuses on geographical data, we aimed for a taxonomy that is not based on a specific data domain. Unlike [Lee et al., 2006], we also tried to avoid limitation to some specific

[1] https://extremepresentation.typepad.com/blog/2006/09/choosing_a_good.html

visualizations set like network visualizations. The typology presented by [?] helps users formulating a scientific inquiry that leads to a generic user task. Our taxonomy identifies and classifies such generic user tasks or visual goals and recommends the related visualizations based on the specifications of each task. For example, a specification for the composition task is different from that of the distribution or the comparison task. Each task has different specifications and is represented differently for different data types. Once the visualization is presented, a user may choose different operators like filter, zoom, etc., to get insight into the data and the graphics or change his task and choose another visualization.

We reviewed more than 5000 statistical/scientific data visualization images from different scientific journals for the creation of this taxonomy. We have generated a database of more than 59 different data visualizations (see Appendix C) through this review. Further, this review has led us to understand six basic visual goals or user intentions for visualizing data. These goals can further vary into subgoals. Representing these subgoals into different data dimensions generates different visualizations. For example, a user goal of distribution can be decomposed into clustering, correlation, and distribution. Here, the visualizations for clustering of two or more variables will be different. The visualizations for correlation among one, two, and more variable will be different. The visualizations for the distribution of one, two, and more variables will be different. The visualizations for the spatial and temporal variables will be different. Hence all these different visualizations represent different facets of one main goal, i.e., Distribution. To further clarify this point, let us look at the example of network visualizations. Someone interested in understanding a network might want to know the hierarchy within the network or might be interested in the flow within the network. The visualizations for representing the hierarchy are different from the visualizations for representing the flow. Furthermore, the visualizations that represent the flow into temporal, non-temporal, and spatial domains are different. We have provided the taxonomy as a list view in Figure 22. It has six main visual goals:

1. **Network** is either in the form of Hierarchy or Flow. Flow can be seen on the temporal, non-temporal and spatial scales.

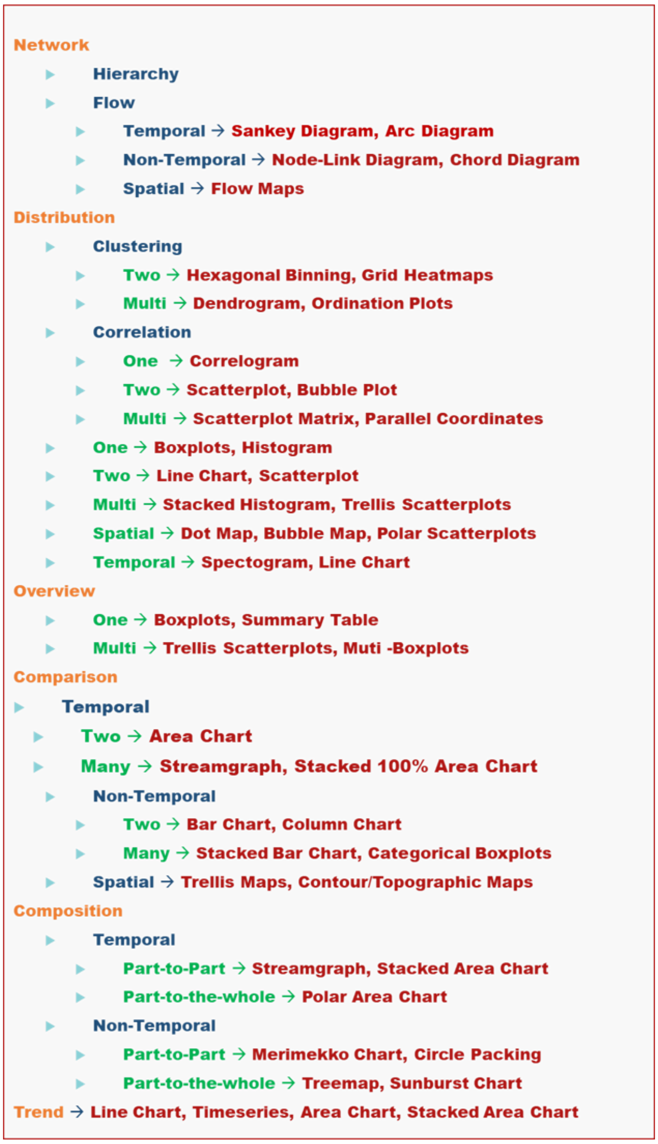
2. **Comparison** can be viewed between two or more temporal and non-temporal entities. Furthermore, the change in one entity over temporal or spatial scales can also be visually depicted under the goal of comparison.

3. **Overview** of one variable and overview of multiple variables can be displayed according to different visualizations.

4. **Composition** It can be defined either as a part-to-the-whole or part-to-the-part relationship. This further can be represented differently on temporal, non-temporal, and spatial dimensions.

5. **Distribution** can be viewed for one, two or multiple variables with different visualizations. Geospatial distribution can also be represented by different versions of distribution maps, e.g., Choropleth Maps, Dot Map or Bubble Map. Distribution then leads to either depict correlation or clustering. Correlation can further be visualized for one, two or multiple variables. At the same time, clustering can only happen between two or multiple variables.

6. **Trend** are all temporal trend represented by specialized visualizations like Timeseries, Line Charts etc.



Network
- Hierarchy
- Flow
  - Temporal → Sankey Diagram, Arc Diagram
  - Non-Temporal → Node-Link Diagram, Chord Diagram
  - Spatial → Flow Maps

Distribution
- Clustering
  - Two → Hexagonal Binning, Grid Heatmaps
  - Multi → Dendrogram, Ordination Plots
- Correlation
  - One → Correlogram
  - Two → Scatterplot, Bubble Plot
  - Multi → Scatterplot Matrix, Parallel Coordinates
- One → Boxplots, Histogram
- Two → Line Chart, Scatterplot
- Multi → Stacked Histogram, Trellis Scatterplots
- Spatial → Dot Map, Bubble Map, Polar Scatterplots
- Temporal → Spectogram, Line Chart

Overview
- One → Boxplots, Summary Table
- Multi → Trellis Scatterplots, Muti -Boxplots

Comparison
- Temporal
  - Two → Area Chart
  - Many → Streamgraph, Stacked 100% Area Chart
- Non-Temporal
  - Two → Bar Chart, Column Chart
  - Many → Stacked Bar Chart, Categorical Boxplots
- Spatial → Trellis Maps, Contour/Topographic Maps

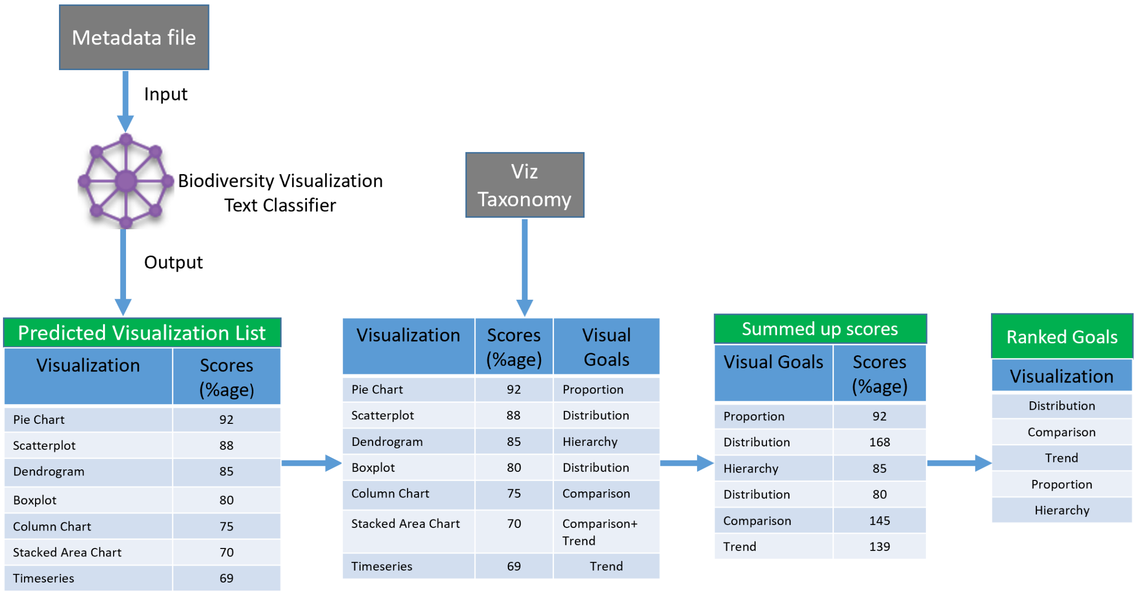
Composition
- Temporal
  - Part-to-Part → Streamgraph, Stacked Area Chart
  - Part-to-the-whole → Polar Area Chart
- Non-Temporal
  - Part-to-Part → Merimekko Chart, Circle Packing
  - Part-to-the-whole → Treemap, Sunburst Chart

Trend → Line Chart, Timeseries, Area Chart, Stacked Area Chart

Figure 22: Visualization Taxonomy

7.2 Visualization Goals Realization

Our biodiversity visualization text classifier produces a list of visualizations with the probabilities of its suitability to the text. Once we get these visualizations, we map them with their visual goals. We have implemented this taxonomy on the Neo4j network database. When a query is sent to the database with input as a visualization list, visual goals related to each visualization are sent back. Then the algorithm adds up the visualization probabilities based on these goals. These summed up probability scores then comes up as a ranked visual goal list. This visual goal realization workflow is presented in Figure 23. Apart from the realized goals, we have included the Overview goal on each visual goal list. This is due to the feedback we received from our domain scientists at the requirement analysis phase



Predicted Visualization List

Visualization	Scores (%age)
Pie Chart	92
Scatterplot	88
Dendrogram	85
Boxplot	80
Column Chart	75
Stacked Area Chart	70
Timeseries	69

Visualization	Scores (%age)	Visual Goals
Pie Chart	92	Proportion
Scatterplot	88	Distribution
Dendrogram	85	Hierarchy
Boxplot	80	Distribution
Column Chart	75	Comparison
Stacked Area Chart	70	Comparison+ Trend
Timeseries	69	Trend

Summed up scores

Visual Goals	Scores (%age)
Proportion	92
Distribution	168
Hierarchy	85
Distribution	80
Comparison	145
Trend	139

Ranked Goals

Visualization
Distribution
Comparison
Trend
Proportion
Hierarchy

Figure 23: Visual Goals Realization workflow

of this work. They mentioned that after data refinement, the preliminary task they perform is data overview (see subsection 2.1.4).

7.3 Summary

In this chapter, we have presented our visualization taxonomy through which we derive our visual goals. We have presented a workflow that explains the process of getting ranked visual goals for the biodiversity text. Once our biodiversity visualization text classifier classifies the biodiversity text. It then provides a list of ranked visualizations. The ranking is based on the classification scores of each visualization type. Then based on the visualization taxonomy, these charts are mapped to different visual goals.

Chapter 8

Variable Selection Algorithm

Our visualization recommendation system's first core element is to generate visual goals from the biodiversity text automatically. Chapter 6 and Chapter 7 present a complete workflow, to attain this by using our biodiversity visualization text classifier and our visualization taxonomy. The second core element of our visualization recommendation system is to visualize large multi-dimensional datasets efficiently. For that purpose, we have devised a variable selection algorithm based on the biodiversity context. This chapter discusses the development of this algorithm in detail.

Variable selection algorithm is similar to the feature selection algorithms in machine learning. A feature selection algorithm decreases the dataset's feature space by selecting appropriate features [Venkatesh and Anuradha, 2019], to increase the efficiency of a machine learning model. A variable selection algorithm is the one that chooses the subset of the variables that can be visualized for a better understanding of the dataset.

Let us consider the following scenario to understand the importance of the variable selection algorithm for a visualization system. Given a:

- Dataset with 40 quantitative variables
- 2-d visualization with x and y axis. For example: a scatterplot.

The possible number of combination for one visualization is the permutation relationship [Gilson et al., 2008] and is represented by:

$$\binom{n}{r} = \frac{n!}{(n-r)!} \tag{8.1}$$

-where n is the number of variables and r is the number of axes or dimensions. This, in our scenario will yield:

$$40!/(40-2)! = 1560$$

These many combinations for one visualization type will increase the visualization search space [Chen et al., 2008] and make it more challenging to explore and understand the dataset. Therefore, to reduce this space, different studies have applied various techniques that we have already presented in Chapter 3.

Our approach to reducing this space is by applying the context from the metadata files which come with each dataset. We have used text mining techniques on these files to filter contextually relevant variable sets that can provide a good understanding of the dataset. Our context-aware variable selection approach is based on two important assumptions:

1. Those terms which are important for the dataset are mentioned more frequently than others in the metadata.

2. Not all variables are equally important in the exploration of the dataset.

Based on the first assumption, a corpus of important terms can be used to significantly reduce the variable space and aids in selecting important variables for the visualization of the dataset. The second assumption allows the reduction of the variable space based on 1) filtering the important variables, 2) disregarding variables that cannot directly contribute to the analysis. Examples of such variables are comments, record numbers or row identifiers.

8.1 Context-aware Variable Selection Algorithm

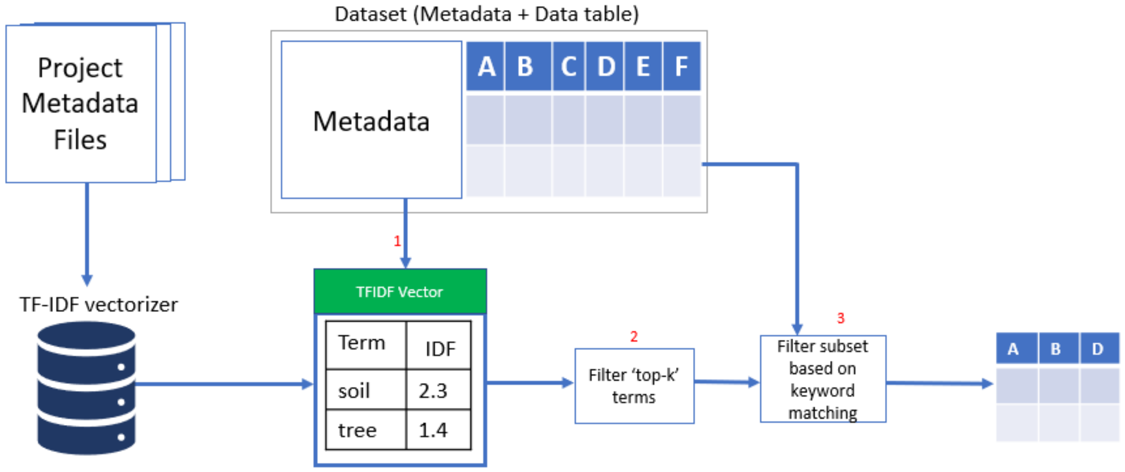


Figure 24: Workflow of the context-aware variable selection algorithm

The workflow of our context-aware variable selection algorithm is presented in Figure 24. This workflow consists of three different steps:

1. TF-IDF vectorization from project metadata

2. Selection of 'top-k' terms as keywords

3. Filtering variable subset based on keyword similarity measures

8.1.1 Data preparation

For creating and testing our algorithm, we took eight publicly available metadata files from the BEF-China[1] project and thirteen publicly available metadata files from the Biodiversity Exploratories[2] project. From the pool of publicly available

[1] www.bef-china.de

[2] www.biodiversity-exploratories.de

metadata files, we have selected a number of those metadata files which have rich content and are well-formatted. These files were then semantically enriched by annotating with the biodiversity domain specific tagger. It is essential to mention that these files were only used as a helper in creating this algorithm. The efficiency of the algorithm does not depend upon the content of these files. Our algorithm can be used with any biodiversity metadata and data files.

8.1.2 TFIDF vectorization from project metadata files

For filtering the interesting variables from the metadata files, we assume that those words which are really important for the dataset must have been mentioned more frequently than others in the metadata. We used the widely popular measure TF-IDF [Berry and Kogan, 2010] to filter these words.

TF-IDF is an established measure from the field of information retrieval and stands for term frequency (TF) and inverse document frequency (IDF). Text documents can be TF-IDF encoded as vectors in multidimensional euclidean space. The space dimensions correspond to keywords (also called terms or tokens) appearing in the documents. The coordinates of a given document in each dimension (i.e., for each term) are calculated as a product of two sub measures: term frequency and inverse document frequency [Jannach et al., 2010].

Term frequency describes how often a specific term appears in a document (metadata file for us). *Inverse document frequency* reduces the weight of terms that appear very often in all documents in the collection. This collection for us is all metadata files belonging to the same project.

Performing a TFIDF-vectorization on the collection of metadata files, produces a list of terms with the IDF scores for each metadata file.

8.1.3 Selection of top IDF scored keywords

Once we have a list of keywords with IDF scores for each metadata file, the next important step was to choose the *top-k* terms from this list. In most cases, this k can be manually selected. Automatic selection of a threshold is a well-studied problem in information retrieval. Current threshold selection methods for example, Verne method [Vergne, 2004], Zipf curve [Piantadosi, 2014] and Otsu's threshold [Eler and Garcia, 2013], are all based on count of the frequent terms and document length. In our case, we want to select a threshold that is not based on the document length but the number of variables, as this threshold will not be used in document extraction but will be used in the variable extraction. After analyzing various techniques, we have realized that the square root of the Binomial Coefficient of the total variable count (Equation 8.3) might be a good threshold for our problem. Our threshold is:

$$Threshold(\theta) = \sqrt{\binom{n}{r}} \tag{8.2}$$

where:

- n is number of variables in a dataset.

- r is the number of axis or visual marks for one visualization type.

Let us consider Equation 8.1 to understand why it could be an appropriate value. We know that the resultant combinations or visualizations for one 2-dimensional (2-d) visualization of 5 variables are 20, considering they all have the same data attribute. In this formula, axis combination has been counted twice. Which is true because if *a* and *b* are the variables for one scatterplot, then at one time *a* will be horizontal axis and another time *b* will be horizontal axis. The same goes for the vertical axis. However, such a swap of variables is accomplished in seconds with the modern visualization tools. Therefore we decreased the effect of this swap in the Equation 8.1 by dividing *n!* by an extra *r!*. It transformed the Equation 8.1 to the binomial coefficient (bc) in Equation 8.3.

$$\binom{n}{r} = \frac{n!}{r!(n-r)!} \tag{8.3}$$

The binomial coefficient is the number of ways of picking r unordered outcomes from *n* possibilities, also known as a combination or combinatorial number [Fowler, 1996]. In our situation, it creates combinatorial number of visualizations that are possible with the provided variable count. Consider a dataset with five variables and a two dimensional visualization *(n=5, r=2)*. Adding it to the Equation 8.3, will yield 10 combinations of one visualization type. However, as the variable count grows, so does the value from Equation 8.3, which is number of visualizations for the dataset. Therefore, we need a function that can reduce this growing number of combinations. In statistics and data analysis, a transformation function is used to replace a variable to reduce the effect of the growth of the original variable to the one side of the axis or, in other terms, to reduce the right skewness of the distribution. In comparison to other transformation functions, the square root *x* to *x^(1/2) = sqrt(x)*, is one with a moderate effect on distribution shape. It is weaker than the logarithm and the cube root. These functions are also used to reduce right skewness and are commonly applied to discrete data, especially if the values are mostly rather small [Emerson and Stoto, 1983]. Therefore for our problem, we use the Square Root (SQRT) function to reduce the ever-growing combination of binomial function.

Table 8.1: Binomial Coefficient (BC) and an effect of Square root (SQRT) transformation. VC is variable count.

VC	BC	SQRT
2	1	1
3	3	1.73
4	6	2.44
5	10	3.16
6	15	3.87
7	21	4.58
8	28	5.29
9	36	6
10	45	6.70

Figure 25 shows the effect of the SQRT function in reducing the exponential growth of the combinations. Table 8.1 shows the SQRT of the binary coefficient is always close to (but not exactly) half of the original variable count.

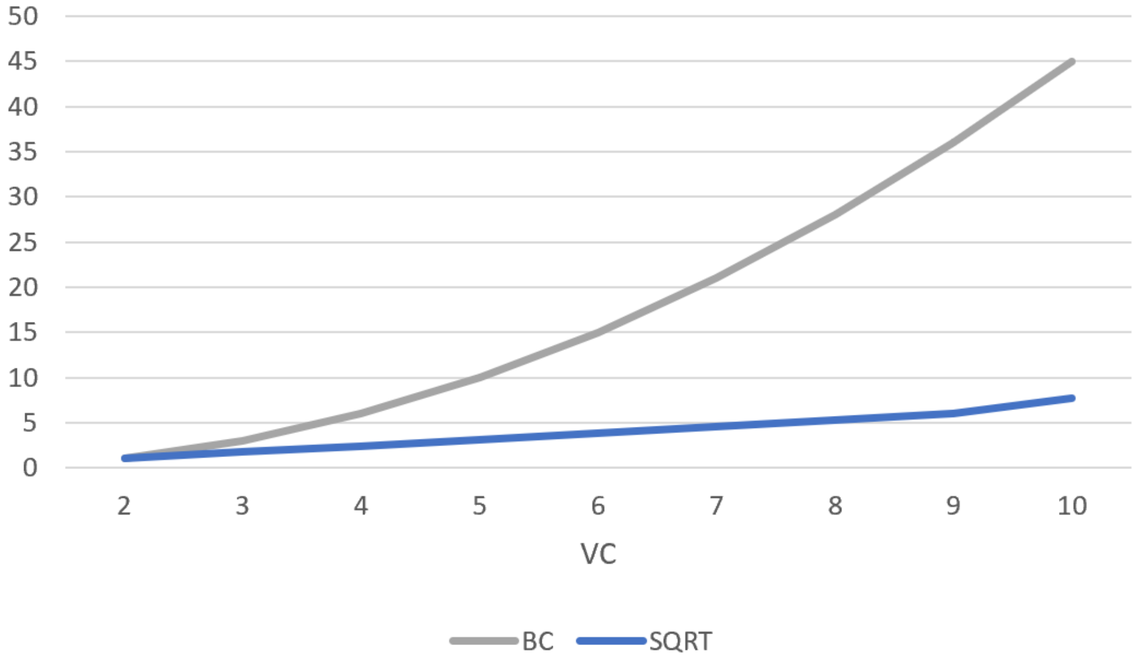


Figure 25: Effect of square root (SQRT) function in reducing the growth of combinations from binomial function (BC).

Getting back to our context of a function to find threshold or 'top-k' from a provided tf-idf list. Considering a case if one visualization need a minimum of two variables, then for 10 variables, Equation 8.3 is 5 which is also our *top-k*. Once we have a list of *top-k* keywords, we refine them using BiodivTagger [Löffler et al., 2020]. BiodivTagger is a biodiversity domain ontology-based annotation pipeline. It recognizes biological, physical, and chemical processes, environmental terms, data parameters, phenotypes, and materials and chemical compounds and links them to concepts in dedicated ontologies. All these terms have some analytical significance that can assist in providing insight into the data. Thus, these tagged terms were kept, and the rest were ignored. To see the effect of using BiodivTagger on the terms, we will consider an example of one of our test dataset[3]. After processing the metadata of this dataset, applying Equation 8.2 on the variable count from the dataset, 19 terms were filtered from the idf list. These terms are:
['area', 'species', 'leaf', 'plot', 'scientific', 'seedling', 'plant', 'site','number', 'helper', 'biomass', 'damage', 'height', 'dead', 'ground','name', 'understand', 'set', 'data']

Running this list through the BiodivTagger, we removed all unwanted terms like *"scientific", "data", "number", "helper"* etc. and kept useful terms as shown below:
['plot','species','area','biomass','height' 'leaf','dead','seedling','damage']

8.1.4 Filtering the variable subset based on keyword similarity

Once we have a refined list of keywords, the next step is to use them to filter the variables. Standard similarity checks are used to find the term-similarity between

[3] https://data.botanik.uni-halle.de/bef-china/datasets/577

the keywords and the tokens of definitions and ids. They are presented in the Algorithm 1.

Algorithm 1 Variable Selection Algorithm

Ensure: Set KEYWORD as Keyword list
Ensure: Set ID as Variable Id set
Ensure: Set DEF as Variable Definition set

```
 1: for k in KEYWORD do
 2:     for id in ID and def in DEF do
 3:         if k in id then
 4:             $FIL\_VAR \leftarrow ID$
 5:         else if length(id) == length(def) then
 6:             if k in def then
 7:                 $FIL\_VAR \leftarrow ID$
 8:             end if
 9:         else if length(id) != length(def) then
10:             if SIM_SCORE(id,k) >= 0.8 then
11:                 $FIL\_VAR \leftarrow ID$
12:             end if
13:             if SIM_SCORE(id,k) < 0.8 then
14:                 $SIM\_NOM \leftarrow SIM\_NORM(def, k)$
15:                 if SIM_NOM >=0.7 then
16:                     $FIL\_VAR \leftarrow ID$
17:                 end if
18:             end if
19:         end if
20:         if length(FIL_VAR)<=2 then
21:             if  k in id/def then
22:                 $NEW\_FILVAR \leftarrow ID$
23:             end if
24:         end if
25:     end for
26: end for
```

8.1.4.1 Similarity Condition

In Algorithm 1, we have applied three types of keyword similarities. After careful analysis of different styles by which text in variable ids and definitions are defined in biodiversity metadata files, these conditions were formed.

1. **Keyword similarity:** It is a straightforward similarity condition, stated in Algorithm 1 Line 3. If any of the keywords are present in the *i*d term, then, those ids are filtered. For example: consider we have an *id soil_d* with *def soil depth.* If we have keyword *s*oil, then, this id will be filtered.

2. **Length based similarity:** It is a special ad-hoc similarity condition that we have introduced in our algorithm (see Algorithm 1 Line 5). It is based on

the similarity between the length of *id* tokens and respective *def* tokens. For example, continuing with the same example above, if we now have a keyword *depth*, then the 'Keyword similarity' will not work. Therefore we introduce this condition, wherein if the length of the *id* tokens are same as length of *def* tokens (which in our case is true after normalization or removing underscore), then the keyword similarity with *def* tokens will be checked. This will then filter out the variables.

3. **Distance based similarity:** At this stage of the similarity check, we used different empirical similarity measures to do an intensive keyword search (Line 10, 13 and 14). It is named *SIM_SCORE* and *SIM_NORM_SCORE*. *SIM_SCORE* function takes two tokens as input and produces a similarity score between 0.0 to 1.0. To find the similarity, this function uses two state-of-the-art string distance metrics.

   Levenshtein: Levenshtein function finds the smallest number of insertions, deletions, and substitutions required to change one string or tree into another. It is a $\Theta(m\ddot{O}n)$ algorithm to compute the distance between strings, where m and n are the lengths of the strings [Levenshtein, 1966].

   Jaro-Winkler: Jaro method measures the weighted sum of percentage of matched and transposed characters from two strings. Winkler modified this algorithm to support the idea that differences near the start of the string are more significant than differences near the end of the string [Winkler, 1999]. The Jaro-Winkler distance uses a prefix scale which gives more favourable ratings to strings that match from the beginning for a set prefix length.

```
function SIM_SCORE(id,k)
    $LEV\_SIM \leftarrow levenshtein(id, k)$
    $ed \leftarrow 1 - LEV\_SIM$
    $JW\_DIS \leftarrow JARO\_WINKLER(id, k)$
    $MAXI\_DIS \leftarrow maximum\_value(LEV\_SIM, JW\_DIS)$
end function
```

SIM_SCORE function gets id and keyword tokens as an input. Then it calculates the Levenshtein and Jaro-Winkler distance. From these two distances, it only selects the maximum value (MAX function i.e., maximum of two functions). For a variable to be selected at this stage, its *id* should have atleast 80% similarity to the keyword. Apart from using the MAX function, we have also tried to calculate the average of these two scores with an average function. However, we found that MAX function can filter more true positive variables. Therefore we used it for further development.

SIM_NORM_SCORE is a short form for normalized similarity score. In the case of *def* where there are more than one token, this function uses the SIM_SCORE to calculate the distance between two tokens and normalize the score based on the token count in the definition. This reduces the effect of definition length on the similarity scores. If this SIM_NORM_SCORE is

```
function SIM_NORM_SCORE(def,k)
    $CAL\_FREQ \leftarrow TOKEN\_FREQ$
    $NORM\_WGT \leftarrow \frac{CAL\_FREQ}{def\_TOKEN\_COUNT}$
    $alpha \leftarrow 0$
    $SIM\_SCORE \leftarrow SIM\_SCORE(k, def)$
    $SIM\_NORM\_SCORE \leftarrow alpha + (SIM\_SCORE * NORM\_WGT)$
end function
```

more than 70% or 0.7 for any keyword and its definition, then the variable will be filtered out else it would not.

These filtered variables are then represented by different visualizations based on the provided visual goals from Chapter 7.

8.2 Summary

In this chapter, we have presented our biodiversity context-aware variable selection algorithm. We have presented a workflow by which any domain-based variable selection algorithm can be created to visualize the datasets. While observing the state-of-the-art of the feature selection algorithms in Chapter 3, we have observed that such algorithms are rarely used for the development of the visualization recommendation system. In this chapter, we have discussed the importance of these algorithms in decreasing the variable space and in return the visualization space in the recommendation systems. This algorithm is strictly based on the data's context by extracting important keywords from the metadata files, labeling them with the BiodivTagger, and using different text-matching techniques to filter the variables based on these keywords. It is an ad-hoc algorithm that is not based on some pre-derived corpus. Therefore, it's efficiency is based on the currently available metadata files and their quality. In Chapter 10, we have presented this algorithm's evaluation based on the ground truth gathered from the biodiversity community.

Chapter 9

Knowledge-based Visualization Recommendation System

This research provides the biodiversity community with a visualization recommendation tool for visual data exploration. These recommendations are based on the domain as well as the context of the dataset. In Chapter 6, we presented the mechanism through which we have gathered the biodiversity domain knowledge from the publications. These text-based visualization classifier gets a text from the field of biodiversity and suggests appropriate visualizations. Chapter 7 showed how we used these predictions from the classifier to realize the visual goals or visual tasks. In Chapter 8, we showed the mechanism through which we contextually select the subset of the variables to reduce the dimensions of the dataset to be visualized. This chapter discusses our final software solution, which has been created by connected all these modules.

9.1 Architecture

As also discussed in Chapter 5, we have extended the original visualization reference model provided by [Heer and Agrawala, 2006] into a Knowledge-based visualization model by including a knowledge engine module (Figure 26).

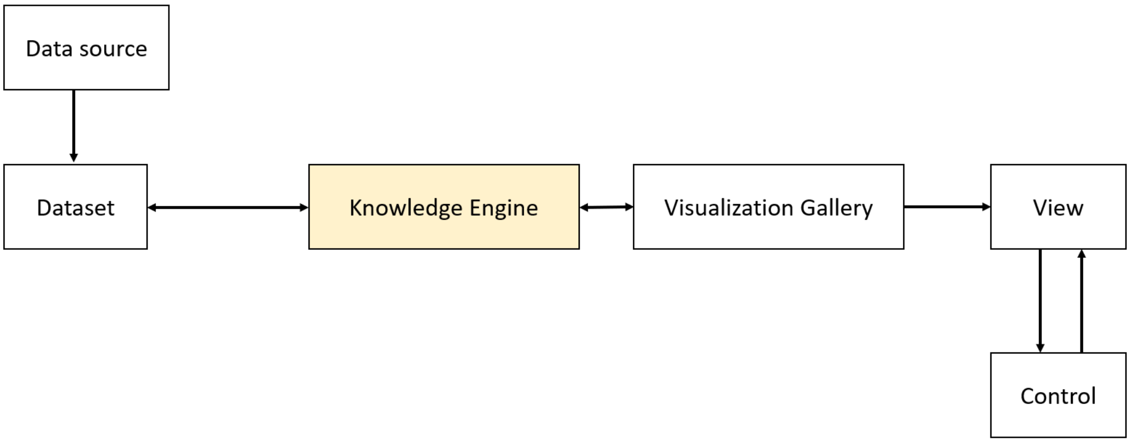


Figure 26: Knowledge-based visualization recommendation model

In the following, we describe each component of this model:

- **Data source:** Data source component is a database connectivity interface or a file reader that loads a dataset to be visualized.
- **Dataset:** It is a dataset that needs to be visualized. For us, it is a combo of *.xml*[1] formatted metadata files and tabular raw data files in a *.csv*[2] format.
- **Knowledge Engine:** This component is a core component of our model. It integrates the domain knowledge in the model. It is an intelligent system that takes in the *.xml* data files and infers two essential components of our visualization system: visual tasks and a subset of variables to be visualized.
- **Visualization Gallery:** The visualization gallery provides the necessary code and visual mappings to run a visualization. As per the defined visual goals, appropriate visualizations from the visualization gallery is selected. Visualization code then calls the actual dataset to retrieve the needed data columns.
- **View:** View is an interactive interface to show the visualizations to the user.
- **Control:** A user uses different controls provided on the interface to select and configure the visualization. Control callbacks to the visualization gallery whenever a new visualization is selected. This triggers different functions as per the selected visualization, which calls for the dataset columns to show up on the interface.

9.2 System Configuration

In Figure 27, we show the configuration of our visualization recommendation system. It runs on the flask[3] application programming interface (API), which can be either deployed online or offline. Flask is a lightweight WSGI (Web Server Gateway Interface) web application framework. It is designed to make getting started quick and easy, with the ability to scale up complex applications. It has become one of the most popular Python web application frameworks. Flask offers suggestions but does not enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that makes adding new functionality easy. The flask server initiates the main file, which reads in the request from the front end or the view and sends it to the different functions that activate the back-end files. These files then send their responses to the main file, which are further sent to the front-end or view.

The backend files are all Python files, which are directly connected to the dataset or data source. For our system, these backend files perform the whole logic of the Knowledge Engine as shown in Figure 12. The front end (see Figure 28) is a javascript file that defines the layout and the different graphical elements. Using different javascript event handlers, these elements speak with the main file, which further calls to the backend files. Further, we have defined the various calls that it makes to the API:

[1] Extensible Markup Language

[2] Comma-separated Value Files

[3] `https://palletsprojects.com/p/flask/`

1. **Selected Data:** A drop-down is provided with the list of different datasets, configured to be visualized on this interface. As the user clicks on any dataset number, it sends that number to the API. API further sends this to the backend files.

2. **Visual Goals:** Based on the selected dataset number, the backend files send out the list of visual goals that could be appropriate to explore this dataset. This list is further sent to the interface in the sequence of its suitability to the dataset, which is then displayed on the screen (see Figure 28).

3. **Selected Visual Goal:** From the displayed list of visual goals, the user selects a goal of its interest. This information is sent back to the API, which further triggers the backend files.

4. **Related Visualization:** As a response to the call of the selected visual goal, the backend files then send out the list of all the corresponding visualizations that can visualize the selected goal. The list is then displayed as clickable thumbnail pictures of the visualizations (see Figure 28).

5. **Selected Visualization:** The user then selects the visualization of its interest. This information is sent to the backend files via flask API.

6. **Rendering:** The backend file then sends out the python code for the selected visualization to render it on the screen. The user then explores these interactive visualizations and use different controls to manipulate them.

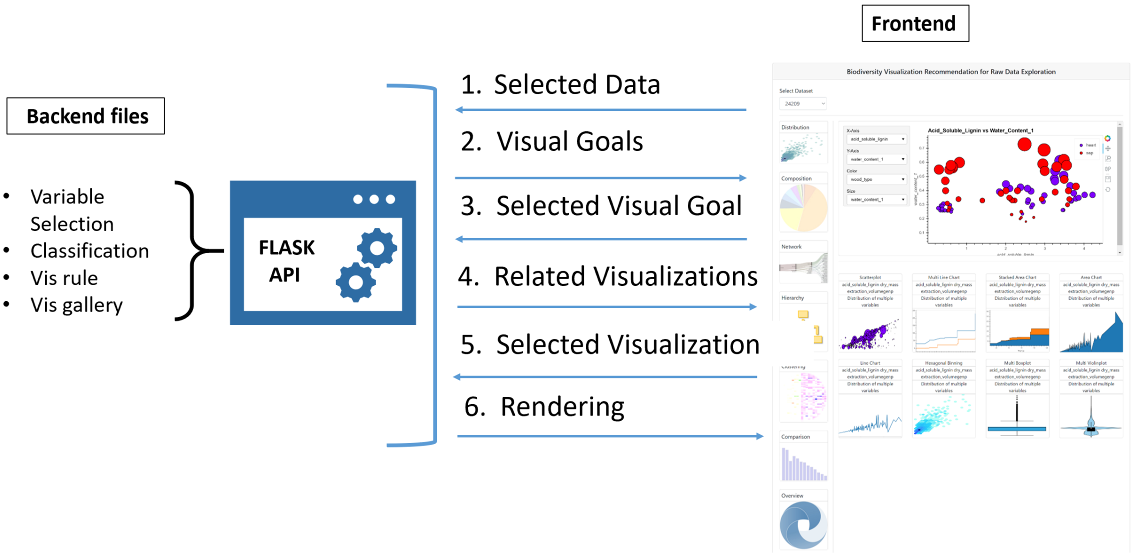
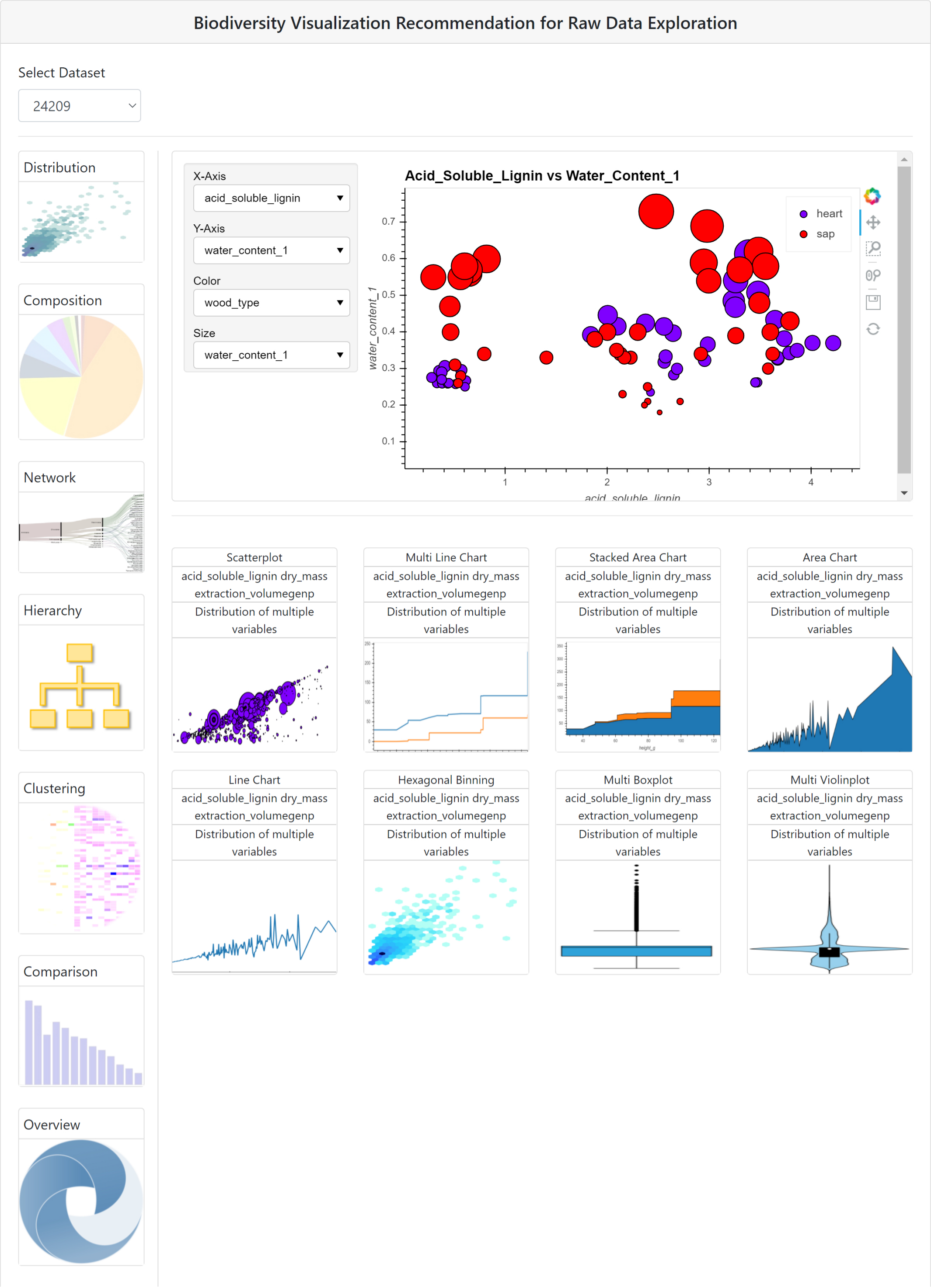


Figure 27: System configuration



Biodiversity Visualization Recommendation for Raw Data Exploration

Select Dataset
24209

X-Axis
acid_soluble_lignin

Y-Axis
water_content_1

Color
wood_type

Size
water_content_1

Scatterplot	Multi Line Chart	Stacked Area Chart	Area Chart
acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp
Distribution of multiple variables	Distribution of multiple variables	Distribution of multiple variables	Distribution of multiple variables

Line Chart	Hexagonal Binning	Multi Boxplot	Multi Violinplot
acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp	acid_soluble_lignin dry_mass extraction_volumegenp
Distribution of multiple variables	Distribution of multiple variables	Distribution of multiple variables	Distribution of multiple variables

Figure 28: Biodiversity visualization recommendation tool

We have used different Python visualization libraries to render visualizations and show different controls to interact with the application:

HoloViews is an open-source Python library designed to make data analysis and visualization seamless and straightforward. In HoloViews[4], the declaration of data is entirely independent of the plotting implementation. HoloViews plots are

[4] www.holoviews.org

based on another visualization library Bokeh[5]. *bokeh.models* component provides a powerful platform to generate interactive plots using HTML5 canvas and WebGL and is ideally suited towards an interactive exploration of data. By combining the ease of generating interactive, high-dimensional visualizations with interactive widgets and fast rendering provided by Bokeh, HoloViews is a powerful visualization library. **hvPlot** is a high-level plotting API built on HoloViews that provides a general and consistent API for plotting data in different data formats. hvPlot[6] can integrate neatly with the individual libraries if an extension mechanism for the native plot APIs is offered or used as a standalone component. hvPlot provides an alternative for the static plotting API provided by Pandas and other libraries, with an interactive Bokeh-based plotting API that supports panning, zooming, hovering, and clickable/selectable legends. **Plotly Express** is a terse, consistent, high-level API for creating figures. The *plotly.express*[7] module contains functions that create seamless figures and is referred to as Plotly Express. Plotly Express is built as a part of the plotly library. Every Plotly Express function uses graph objects internally and returns a *plotly.graph_objects.Figure* instance, which then creates different graphics. **Panel** is an open-source Python library that can create custom interactive web applications and dashboards by connecting user-defined widgets to plots, images, tables, or text. Panel[8] supports nearly all plotting libraries, can work both in a Jupyter notebook as on a standalone secure web server, uses the same code for both those cases, supports both python backend and static HTML/JavaScript exported applications and can be used to develop rich interactive applications without tying domain-specific code to any particular GUI or web tools. The panel provides a wide range of widgets to provide precise control over parameter values. The *widget* classes use a consistent API that allows treating broad categories of widgets as interchangeable. For instance, to select a value from a list of options, one can interchangeably use a *Select* widget, a *RadioButtonGroup*, or a range of other equivalent widgets. Like all other Panel components, *widget* objects render and sync their state both in the notebook and on the Bokeh server. By the use of the Bokeh visualization server, we put different visualization libraries on one platform. The purpose of the *bokeh.server* is to make it easy for Python users to create interactive web applications that can connect the front-end to running Python code. A bokeh application is a Python code run by a *bokeh.server* when new sessions are created. Bokeh's architecture is such that high-level *model objects* (representing plots, ranges, axes, glyphs and other chart elements) are made in Python and converted to a JSON. This capability to synchronize between Python and the browser is the primary purpose of the Bokeh server.

9.3 Summary

This chapter provides a system configuration of our biodiversity knowledge-based visualization recommendation system, which results from our research presented in this thesis. We presented various technologies through which we were able to develop our research prototype. It is a flask-based application with a backend of Python

[5] `http://docs.bokeh.org/`

[6] `http://hvplot.holoviz.org`

[7] `www.plotly.com/python/plotly-express`

[8] `www.github.com/holoviz/panel`

script and a front-end of a JavaScript application. This prototype is available to explore online[9].

[9] www.visapps.de

Chapter 10

Evaluation

This chapter presents a quantitative and qualitative evaluation of our Knowledge-based Visualization Recommendation System. We estimate the quality of our recommendation system in three sub evaluations. These three sub evaluations correspond to the three important aspects of this system and are three main contributions of this thesis:

- The Biodiversity Visualization Text Classifier
- The Context-aware Variable Selection Algorithm
- The Knowledge-based Visualization Recommendation System

10.1 Evaluation of the Biodiversity Visualization Text Classifier

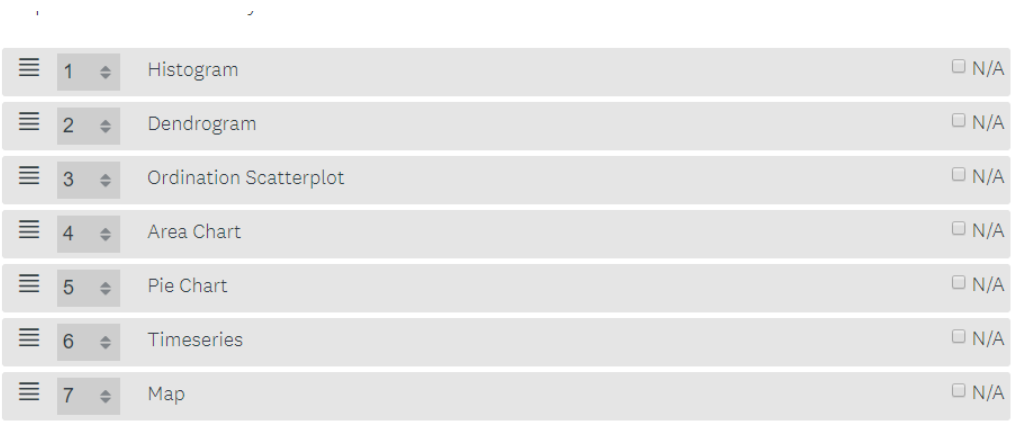
In Chapter 6, we have provided the quantitative evaluation of our biodiversity visualization text classifier. From Table 6.2, we know that our classifier has an average F1-score (accuracy) of 92.2% on the test dataset. Next, we were interested to see how well it performs with our users from the biodiversity domain? This evaluation aimed to know the level of agreement between the classifier's learned concepts and human understanding. For this evaluation, we conducted an online survey. The preview version of which is available online[1]. We randomly selected five publicly available metadata files from the BEF-China data portal [Klein and Staab, 2017, Seitz, 2017, Kühn et al., 2016, Staab et al., 2016, Staab et al., 2019]. Content from these metadata files were then fed to our biodiversity visualization text classifier[2]. The classifier's output is a list of visualization labels ordered by decreasing probabilities of their suitability to the dataset.

There were five questions in the survey, each of which corresponded to the five metadata files. Each question contained information about one metadata (dataset abstract and dataset design) and the classifier's predicted list. A sample of the questions can be accessed at the survey preview. In order to avoid information overload, in this survey, only the first seven visualization options from the predicted list were provided. These options were presented in the form of a drop-down list.

[1] `http://tinyurl.com/bvcpreview`

[2] `http://github.com/fusion-jena/Biodiv-Visualization-Classifier`

Survey participants were required to go through the description and then choose the suitable visualization types, i.e., those that can depict the information in the description well. Participants had to rank these options in the decreasing order of their suitability to the question. Drag and drop, and drop-down buttons were available for the participants to allocate a rank (a number) to each option. An example of a completed answer is provided in Figure 29, where the user assigned sequence numbers to each visualization name or label.



1	Histogram	□ N/A
2	Dendrogram	□ N/A
3	Ordination Scatterplot	□ N/A
4	Area Chart	□ N/A
5	Pie Chart	□ N/A
6	Timeseries	□ N/A
7	Map	□ N/A

Figure 29: Screenshot of the survey answer

Out of the seven provided options, participants were asked to choose at least the five most suitable visualizations. For the rest of the options, they could also choose N/A. Complete information about the survey and the underlying research was provided on the welcome screen of a survey. The survey was open from October 2019 until December 2019. It was advertised to biodiversity domain scientists via various mediums: the most important German biodiversity research conference GFÖ2019[3], the 2019 assembly of the GFBio[4] project, tagged to the social media accounts for different biodiversity research institutions via Twitter and Facebook, sent via email to the mailing lists of various biodiversity research projects and scientists.

10.1.1 Results

In total, 37 responses were received from the survey. Out of these 37 responses, only 11 respondents completed the survey. We consider a survey to be completed if the respondents have spent at least 5 minutes to answer 3-5 questions. A question wise breakdown of the responses is presented in Table 10.1. A resultant data sheet is available online[5].

As not all respondents had selected five options, which was a prerequisite for this survey, therefore, for our primary analysis, we had only included those responses where the participants had at least selected the first four options. The frequency per question for those filtered responses are provided in the last column of Table 10.1.

[3] `http://gfoe.org/de/node/1562`

[4] `www.gfbio.org`

[5] `https://github.com/PawandeepKaur/Biodiversity-Visualization-Recommendaton-Tool`

Table 10.1: Questionwise breakdown of the responses

Questions	Number of responses	Filtered responses
1	11	9
2	7	4
3	6	4
4	7	4
5	6	3

10.1.1.1 Metrics

Before we look into the metrics, it is important to recall what the goal of the survey is: by looking just at the domain-specific text, can our algorithm predict the same visualization type as humans select? To evaluate this, we aim to answer two main questions: 1) how many users have selected the same visualization options from the list as predicted by the classifier? 2) how similar is the ordering or ranking between the classification prediction and the human responses?
To answer the first question, we have used classical precision scores.

Precision: In our case, precision is based on the total number of relevant options (*Relevant_Options*) chosen by the participants from the seven provided visualization options (*Retrieved_Options*). The relevant options are the ones that fall between the rank from 1 to 4 as predicted by the classifier.

$$Precision = \frac{Relevant\_Options \sqcap Retrieved\_Options}{Retrieved\_Options} \tag{10.1}$$

To answer the second question, we have used.

Ranked Biased Overlap (RBO): RBO is the similarity metric that counts the ratio between the overlap (in terms of a number of predicted outputs) at the top-k ranks of the retrieved ranked lists [Webber et al., 2010]. In our case, it is the similarity ratio between the predicted list until the first four options and the user-provided list. There are two methods for evaluating the effectiveness of the ranked list: (1) Rank Correlation and (2) Set Based Measures. Rank correlation-based approaches such as Kendall Tau measure the probability of two items being in the same order in the two ranked lists. However, there is a problem with the top-weightedness with this approach. The problem is that an item's rank or position does not affect the final similarity score. In set-based measures, the concept of a set intersection has been used to quantify the similarity between two ranked lists. The idea is to determine the fraction of content overlapping at different depths or descending positions in the ranked list [Agrawal, 2013]. RBO measure is one such method. The advantages of using it compared to other methods are: (1) it removes the problem of top weightedness by using geometric series whose values decrease with the increasing depth or number of options in the ranked list. Thus it explicitly models the likelihood of going from a given rank position to position i+1.

$$\sum_{d=1}^{\infty} p^{d+1} = \frac{1}{1-p} \tag{10.2}$$

Rank-biased overlap scores as computed with Equation 10.2, fall in the range [0, 1], where 0 means disjoint or different, and 1 means identical. The parameter p determines how steep is the decline in weights: the smaller p, the more top-weighted the metric is. When $p = 0$, only the top-ranked item is considered, and the score is either zero or one. On the other hand, as p approaches arbitrarily close to 1, the weights become arbitrarily flat, and the evaluation becomes arbitrarily deep. For in-depth knowledge about this metric, readers are encouraged to follow this publication [Webber et al., 2010]. For our calculation, we have kept the p high, i.e, 0.98, to assign equal weights to all the options in the list.

10.1.1.2 Metrics results

Table 10.2 provides the result of precision and metrics. The average precision is 62% and the average RBO is 61% (decimals are transformed into percentiles). The results show that there is no correlation between the number of respondents and the scores. Q1 has the highest responses, but it has an average of 61.8%, whereas Q5 has the least responses but has the highest average score of 80%.

We have also observed that our classifier has scored better for those questions where more information was provided than the one with lesser information content. We investigated further to see any relationship between the length of the information and the scores. The result is shown in Figure 30. It shows a somewhat positive connection between the word count and the average scores from RBO and Precision. The only exception to this result is with Q2, which has a word count of 146, but upon closer inspection, it contains less information content and more references. We found that our classifier has worked better on those questions where information quantity was dense, and quality was good. This confirms that if enough proper textual content is provided to the classifier, then it performs best. When we started this survey, we intended not to provide lengthy questions that can take a long time for participants to read and then answer.

Table 10.2: Questionwise Precision and RBO results

Questions	Q1	Q2	Q3	Q4	Q5	Mean
Responses	9	4	4	4	3	-
RBO Mean	0.62	0.54	0.61	0.48	0.81	0.61
Precision Mean	0.67	0.56	0.63	0.50	0.75	0.62

10.1.1.3 Treatment of N/A's

For each question in our survey, we asked the users to select the first five visualization options and then leave the rest blank. That was the sole purpose of providing N/A as an option in the survey design, which for us meant non-applicable and was planned to be eliminated in the analysis process. However, some participants had only selected two to three options and had left the rest as N/A. Therefore, after the initial analysis, we wanted to know the effect of those N/A's. In our analysis, we have used two state-of-the-art techniques to deal with incomplete or missing responses. First is listwise deletion [Roth, 1994], which is presented in the previous section wherein we have excluded all N/A options from the analysis. The second approach to deal with N/A is to consider it as another category or factor and analyze the whole result set. In this section, we will present the result on the whole dataset, including N/A's. We considered all N/A's and calculated Precision and RBO on that. It was done in such a way that each N/A is treated as a wrong answer. Thus, we have considered

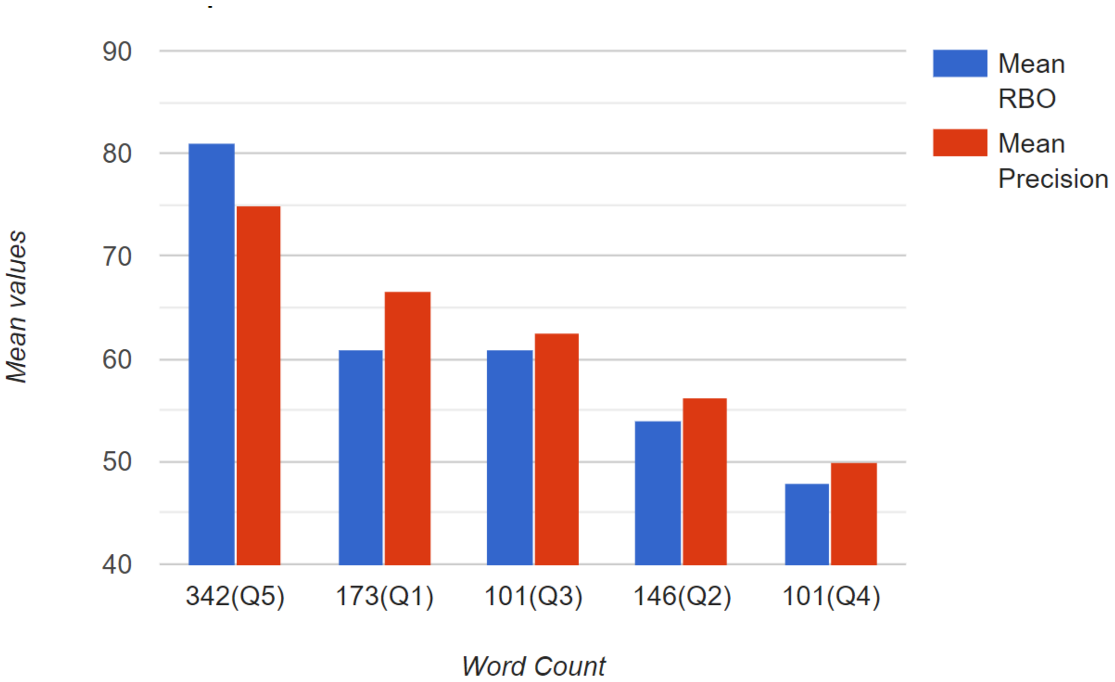


Figure 30: Bar chart depicting dependency of scores on the questions' length

the complete dataset as presented in Table 10.1 column 2. The result of applying Precision and RBO calculation is as follows:

Table 10.3: Questionwise results from Precision and RBO metrics on the whole dataset including N/A's.

Questions	Q1	Q2	Q3	Q4	Q5	Mean
Responses	11	7	6	7	6	-
Precision Mean	0.60	0.48	0.49	0.48	0.60	0.53
RBO Mean	0.64	0.54	0.50	0.50	0.58	0.55

The table shows the mean scores of RBO (55%) and Precision (53%) on the whole dataset. This result shows some differences from the result presented in Table 10.2, where the average Precision is 62%, and the average RBO is 61%, which is better than the analysis with NA's. We used a two-tailed Wilcoxon matched-pairs signed-rank test (WSR) to see if these differences are significant. WSR is a special case of a non-parametric test which checks whether two dependent distributions are the same or not. It is used when the data size is small and has repeated responses. In our case, we are now comparing the mean values of 5 questions of two groups: 1) with N/A's and 2) without N/A's. Our dataset is small, and these groups are not independent of each other. Thus our dataset complies with the condition of the WSR test. Our null hypothesis for this test is, does the difference between two mean distributions (as shown in Table 10.3 and Table 10.2) is significant or not? In other words, are two distributions (mean values) from the two groups are significantly different?

For this significance test, if the *p-value* is less than 0.05, then two mean distributions are significantly different. However in our case *p-value = 0.125*. We can conclude that there is no statistical evidence that these two distributions are different. Thus, we can not reject our null hypothesis with a reasonable probability.

10.1.2 Discussion on the evaluation of the Biodiversity Visualization Text Classifier

From our evaluation, we can conclude that:

- 62% of the time, our classifier has selected the same visualizations as what humans have selected.
- 61% of the time, our classifier has ranked the visualizations in the same order as humans do.
- Even on the metadata files with minimal content, our classifier could be accurate for at-least half of the time.

For getting statistically significant results, evaluation with more participants needs to be done, which we could not accomplish within our timeframe. Moreover, the visualization list from the classifier is not directly used in our system. First, we derive the visual goals from the predicted visualization list and then use these goals in our system. Based on the user-selected visual goal, a richer set of visualizations are presented. The visual goal generation workflow is presented in Chapter 7. In Section 10.3.3.5, we evaluated the recommended visual goals based on the visualization list predicted by our classifier. Wherein our Precision scores are not as good as this evaluation (56%). However, our RBO scores have incredibly improved (86%). The improved RBO score shows that the methodology of visual goals derivation has improved the overall system performance. The improved scores could be described as though the visualizations are different, but their cumulative visual goals are similar. As visual goals directly represent the user's intent, therefore it is more significant for our system. For example, Scatterplot, Histogram, Boxplot, Hexagonal Binning derive a common goal of the data distribution.

10.2 Evaluation of Context-aware Variable Selection Algorithm (CVS)

The Context-aware Variable Selection Algorithm reduces the dataset's dimension by choosing only a subset of the variables that could provide a good overview and understanding of the data. After developing this algorithm (presented in Chapter 8), we wanted to compare the results with the ground truth data. We generated the ground truth from the participants of the 2020 Biodiversity Exploratories (BEO)[6] Assembly in Germany. We selected four datasets — 577[7], 376 [Seitz et al., 2016], 20106 [Schall and Ammer, 2017], 24209 [Noll et al., 2016] — from two different biodiversity projects and printed the content from the metadata files on a sheet in a booklet format. The exemplar questionnaire file is attached in Appendix F. Resultant data sheet is available online[8]. On one side of the sheet were the contents from the individual metadata files. While, on the other side, a table was provided with all the variable ids and descriptions from the same file. Participants were asked to read

[6] `https://www.biodiversity-exploratories.de/`

[7] `https://data.botanik.uni-halle.de/bef-china/datasets/577`

[8] `https://github.com/PawandeepKaur/Biodiversity-Visualization-Recommendaton-Tool`

the metadata's content and then mark the variables that they would like to explore from this dataset.

10.2.1 Results

In total, we received 41 responses from this evaluation. The distribution of the responses per dataset is provided in Table 10.4.

Table 10.4: Total number of responses for each dataset

Dataset	Responses
376	12
577	9
24209	10
20106	10

From these responses, we wanted to calculate the following metrics.

1. **Coverage:** The fraction of the variables filtered from the total available variables [Valdez et al., 2016].
2. S**tandard evaluation metrics:** Precision, Recall or Sensitivity, Accuracy and False Positive Rate (FPR).
3. **Inter-rater reliability:** It is the extent to which two or more raters (or observers, coders, participants) agree. We needed it to see the consistency of our ground truth data.

10.2.1.1 Coverage

Coverage is the fraction of the values that have been covered by a recommendation algorithm [Valdez et al., 2016]. In our case, we are interested to know what percentage of variables have been filtered from the original variable set? Then, we compared this value from our algorithm and the participants. The interpretation of the resultant values differs based on the task. For our task, it should not be too little to the point of information loss. It should also be not too big to diminish the purpose of the algorithm. It should be optimized to the level where it conveys the necessary information about the dataset well. Coverage for our algorithm is calculated by dividing the total number of predicted variables from the dataset's total variable count. Coverage for one user for one dataset is: division of count of user filtered variables (*UserCount*) and total number of variables in the dataset (*VariableCount*). The cumulative Coverage for each dataset is calculated according to the Equation 10.3. It is an average of an individual rater's coverage score. The result is presented in Table 10.5.

$$Coverage_{dataset} = \sum_{i=1}^{n} \overline{(\frac{UserCount_i}{VariableCount})} \tag{10.3}$$

where

- n = number of raters for one dataset.

- UserCount$_i$ = number of variables filtered by one rater for one dataset.

Table 10.5: This table shows for each evaluated dataset: actual variable count, an average of variable count from all the raters (User Count), count of predicted variables by our algorithm (CVS Count), an average of Coverage from all the raters (User Cov) and Coverage scores from our algorithm (CVS Cov).

Dataset	**Variable Count**	**User Count**	**CVS Count**	**User Cov**	**CVS Cov**
376	40	13	15	32.9	37.5
577	27	11	10	41.5	37
24209	52	19	16	36.09	30.7
20106	35	10	8	33.1	22.8
Grand Mean	-	-	-	36	32

The Grand Mean is the mean of the means. Table 10.5 shows that the User Cov covers 36% of the whole Variable Count. The user coverage score is also close to the CVS coverage scores. This shows that our algorithm has filtered an optimal number of the variables from the dataset alike users. In Figure 31, we show the Coverage distribution from all four datasets. In comparison to other datasets, 20106 has the lowest number of range and variance. It conveys that for 20106 raters' coverage scores are uniform. The mean coverage of this dataset is 33. This figure also shows no dependency of raters' count to the range of the coverage scores. For example, dataset 577 has the lowest number of respondents and has the highest maximum value.

Coverage analysis does not ensure the similarity between the filtered variables and the ground truth from the raters. To get those results, we have used empirical metrics, as described in the next section.

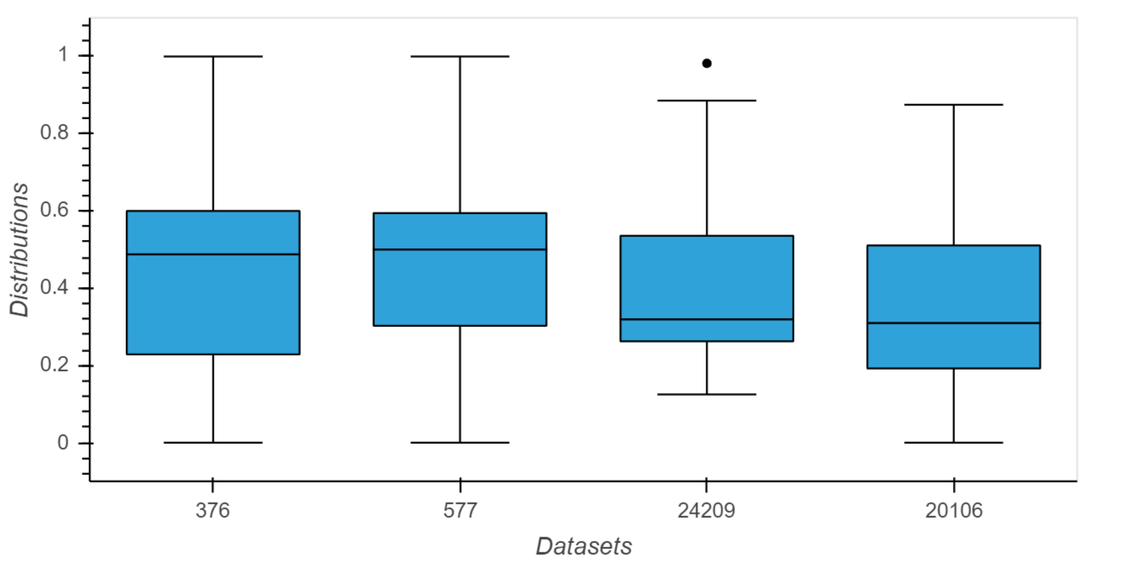


Figure 31: Coverage distribution for all raters for all four datasets

10.2.1.2 Empirical Metrics

To evaluate our Context-aware Variable Selection Algorithm, we have calculated Precision, Recall, Accuracy, and False Positive Rate [Bolón-Canedo et al., 2015]. To get these scores, we need to first consider the results from the CVS as a binary classification problem [Bolón-Canedo et al., 2015], where positive values are those filtered variables that match with the ground truth. The negative values are the one which do not. With this assumption, we can calculate the following scores:

- **True Positive (TP):** Percentage of variables filtered by CVS which were also there in the ground truth.
- **False Positive (FP):** Percentage of variables filtered by CVS which were not there in the ground truth.
- **True Negative (TN):** Percentage of variables that were not filtered by CVS and that were not in the ground truth.
- **False Negative (FN):** Percentage of variables that were not filtered by CVS and that were present in the ground truth.

Based on these scores, we have calculated the following metrics:

Precision: Precision is the positive predicted value and is calculated by the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{10.4}$$

Sensitivity: It is also known as recall. It indicates how well the results predict the actual positives. It is also known as a true positive rate.

$$Sensitivity = \frac{TP}{TP + FN} \tag{10.5}$$

Accuracy: It measures how well the algorithm has predicted both above categories.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{10.6}$$

FPR: FPR is also known as False Positive Rate. The value indicates the number of times the negatives values are wrongly categorized as positive.

$$FPR = \frac{FP}{FP + TN} \tag{10.7}$$

In Table 10.6, we show the scores from each of these measures on all four datasets. Like the Coverage scores, each measure in Table 10.6 is first calculated for individual rater and then averaged for all raters. Grand Mean is then calculated as a cumulative average from all the datasets.

The mean Precision value is 36% and the mean Accuracy of our algorithm on this evaluation is 56%. Figure 32 shows the individual distribution over all the raters for each dataset and each metrics. The provided data conveys that out of all the datasets, our algorithm has performed well on 577. Moreover, the accuracy of our algorithm is relatively uniform for all the datasets.

Table 10.6: This table shows for each dataset the following scores: Precision, Sensitivity, Accuracy and FPR scores.

Dataset	Precision	Sensitivity	Accuracy	FPR
376	0.32	0.28	0.54	0.69
577	0.53	0.41	0.60	0.24
24209	0.30	0.39	0.51	0.40
20106	0.32	0.24	0.62	0.21
Grand Mean	0.36	0.33	0.56	0.38

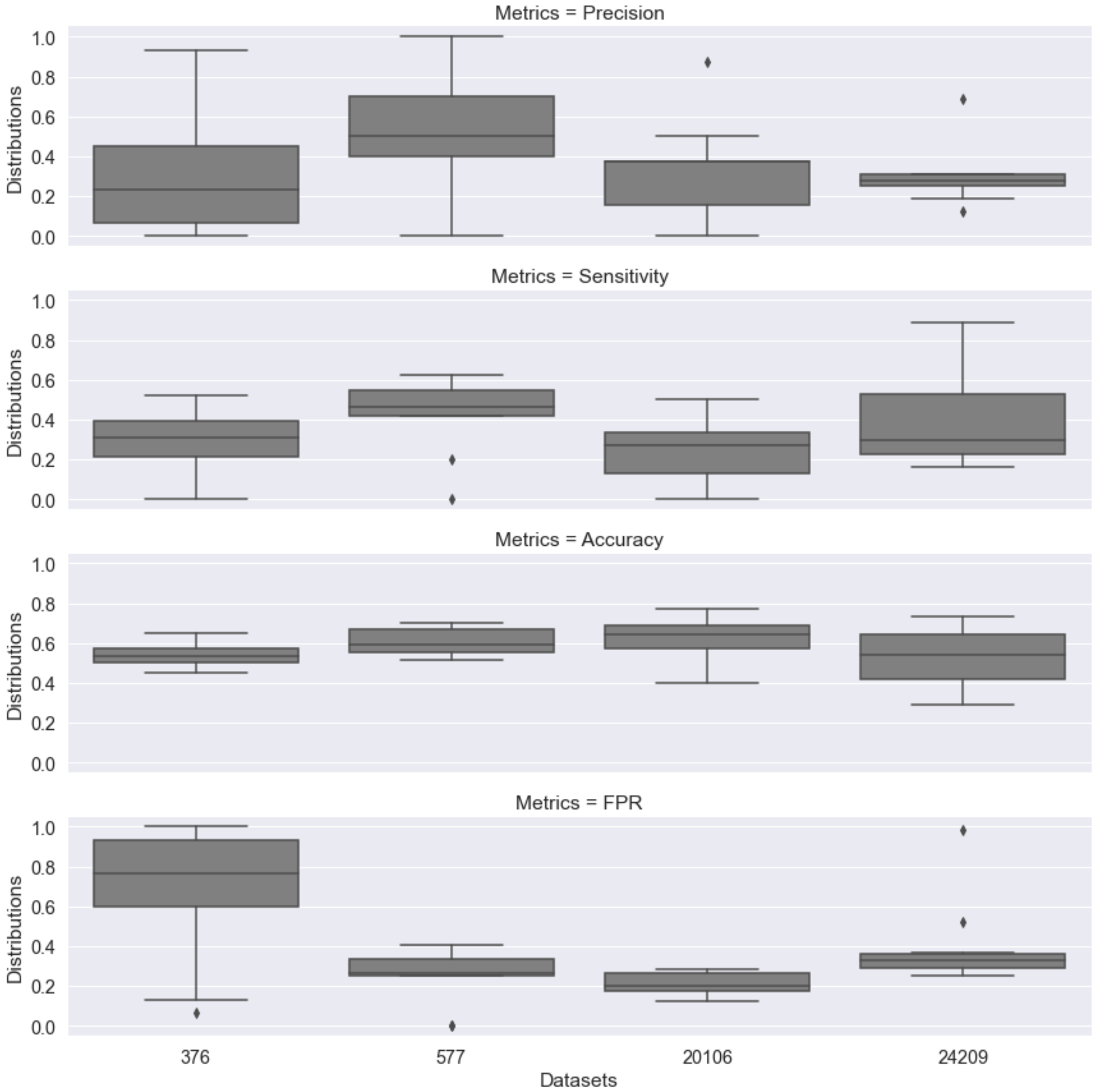


Figure 32: Boxplot shows the distribution of Precision, Sensitivity, Accuracy and FPR from all four datasets.

It is important to note that it is not a machine learning model trained on some preset data. It is an algorithm whose performance can considerably change by various input factors like better semantically annotated metadata, quality of the metadata files or dataset variable count. Elements used in the algorithm's construction have been discussed in Chapter 8. The other important factor that had directly affected the algorithm's performance is the agreement among the participants, which we explain in the next section.

10.2.1.3 Inter-rater reliability

Inter-rater reliability (IRR) or agreement is the extent to which two or more raters (or observers, participants, examiners) agree. As we have used the data from the biodiversity experts as ground truth; therefore, the rater reliability is very significant to us. It shows us the extend of variability among human observers. It is seldom to achieve a perfect agreement, and confidence in study results is partly a function of the amount of disagreement [McHugh, 2012].

For measuring the IRR, we have used the most famous kappa (k) statistics. The kappa is a squared correlation coefficient known as the coefficient of determination (COD) [McHugh, 2012]. Cohen's Kappa is a robust statistic useful for either inter-rater or intra-rater reliability testing. However, it is only limited to measure agreement between 3 raters. In our study, for each dataset, we have more than three participants. Therefore, we have used the Fleiss Kappa measure, an adaptation of Cohen Kappa for three or more raters. The interpretable value usually falls between 0 to 1, where 0 is no agreement.

$$Kappa(k) = \frac{Pr(a) - Pr(e)}{1 - Pr(e)} \tag{10.8}$$

– where Pr(a) represents the observed agreement, and Pr(e) represents a chance agreement

Continuing with our assumption of the binary classification problem, the raters (participants) have to select a variable in the evaluation form to indicate if they want to explore them or not. If the variable is selected, then it is considered positive, and if not, then it is deemed to be negative. Here Kappa will count the correlation between all the raters' agreement over the variables taken from datasets' variables. The results are presented in Table 10.7.

Table 10.7: Inter-rater reliability or agreement

Dataset	**Variable Count**	***k***
376	40	0.10
577	27	0.14
24209	52	0.11
20106	35	0.05
Grand Mean	-	0.10

From the interpretation table by [Landis and Koch, 1977], we know that if the value falls between 0.0 - 0.20, there is only a little agreement among the participants. This is true in our case, as our grand mean for this evaluation is 0.10.

Further, we wanted to look for any particular patterns that can describe the current results. For that, we looked at the percentage of agreed upon variables that were also predicted by our algorithm. The result is presented in Figure 33.

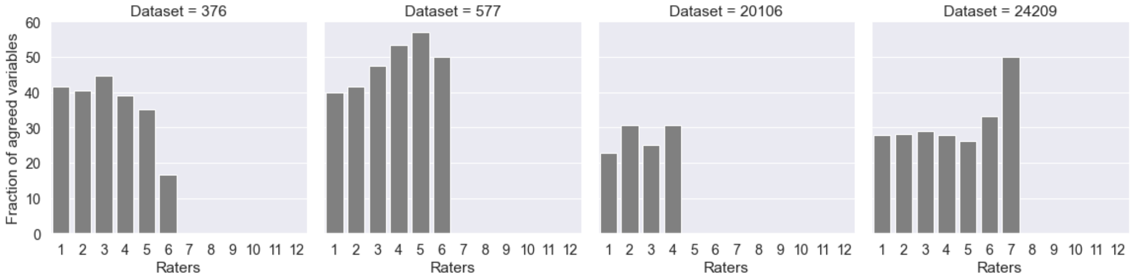


Figure 33: Percentage (Fraction of agreed variables) of variables the raters agreed upon and our algorithm also predicted that.

Each plot in this figure, shows for one dataset – the fraction of variables predicted by our algorithm and selected by at least one participant. From Figure 33, we can only provide one explanation that though our algorithm is not able to select the most voted variables, it has chosen more diverse variables, which are at least picked up by one rater. Even if we cannot fully explain the reason for this behavior of our algorithm, we can foresee the main problem in the future development of this algorithm, i.e., if not even humans can agree on which variables are important, how can we expect an algorithm to find them. What would need to be investigated here is:

1. Is the algorithm smarter than the users? That is, if users look at the variables they picked and the ones the algorithm picked do they say "Oh yes, that choice is much better"?

   OR

2. Maybe what variables one picks depends on what one wants to learn from the dataset. Do we need to provide as input not only the dataset but also the purpose before selecting variables?

10.2.2 Discussion on the evaluation of the Context-aware Variable Selection Algorithm

To summarize the result of this evaluation, we can say that our algorithm has filtered the optimum number of variables from the dataset. It has sub-optimal Precision, Recall, Accuracy, and FPR score with the ground truth. It has low inter-rater reliability scores, which could also be the reason for its bad quantitative results. We have also observed that though the raters' consensus on the predicted variables is not high, the predicted variables still cover the diverse variables within the dataset. More extensive experiments and tests, and further improvement of the algorithm are needed.

Also it is important to note that our algorithm produces results not based on some pre-derived corpus. The results are entirely based on the quality and quantity of the provided input — chosen metadata file, the project metadata files, and the known biodiversity terms. Thus, its efficiency can change based on the current set of inputs. There are no benchmark techniques to evaluate such domain-specific algorithms. In our evaluation, we used the same metrics as used by feature selection algorithms for machine learning. Empirical methods for variable or feature selection do not provide statistics based on the diversity of information filtered.

Such domain-specific algorithms can be improved by using machine learned domain-specific concepts and their related vocabulary (for example, which variables and terms are more often used to describe a particular domain concept). Apart from that, by using interactive machine learning techniques, such algorithms can be online trained by the users at the run time. This will improve the performance of the algorithm for the future recommendations. Moreover, we should also not forget the main aim of this algorithm: to provide a subset that can help get a better insight into the dataset. From this evaluation, we are not sure how the overall system will help in data exploration. The comprehensive system evaluation is provided in the next section.

10.3 Evaluation of the Knowledge-based Visualization Recommendation System

On the evaluation of recommender systems, [Valdez et al., 2016] emphasized that when considering the whole system in real usage scenarios, it is not the algorithm that needs to be evaluated but also the other related factors: interface, HCI, and technology acceptance [Pu et al., 2012]. The technological acceptance criteria include a perceived quality based on users' beliefs and attitudes. The users' beliefs concern the perceived ease of use, perceived usefulness, and control of the system. The users' attitudes are overall satisfaction, confidence, and trust in the recommendations. As the primary purpose of visualization is insight generation [Spence, 2001], for a visualization system, apart from the quality mentioned above, the overall perceived insight measure is the most prominent one. The primary consideration for any life science researcher is discovery [Saraiya et al., 2005]. Arriving at an insight often sparks the critical breakthrough that leads to discovery: suddenly seeing something previously passed unnoticed or seeing something familiar in a new light. The primary function of any visualization and analysis tool is to make it easier for an investigator to glean insight, whether from their data or external databanks [Saraiya et al., 2005]. Beyond predefined data analysis tasks, a measure of an effective visualization can also be its ability to generate unpredicted new insights. The visualization should not only enable biologists to find answers but also to find questions that identify new hypotheses [Saraiya et al., 2005].

For evaluating insights generated through our tool, we partially used some evaluation protocols and insight criteria from the study by [Saraiya et al., 2004]. We then evaluated our system based on the user beliefs and attitudes, which we had captured via user comments and feedback. We will be comparing the insight generated from our tool with that of Microsoft Excel. From the result of our previous study [Kaur et al., 2018], we found that Excel is the second most prominent visualization tool used by biodiversity scientists for data visualization. The first one is R, and because it is a scripting language, it cannot be compared with a graphical user interface. To quantify the qualitative insights, we have used some insight categories developed by [Saraiya et al., 2005]. They consider an insight as an individual observation, a unit of the discovery of the data. We will be using different levels of observations and will count the scores for each dataset. As domain scientists were not comfortable recording their voice or video; therefore, the insight calculation is done based on the answers provided in the questionnaire and the screen recording analysis. For this evaluation, we will be analyzing the following insight categories:

- **Observation:** The actual finding of the data from the visualization. Based on that, we counted the total number of insights for each dataset.
- **Domain value:** The value, importance or significance of the insight. Simple observations such as *"intense precipitation is negatively correlated to precipitation duration"* is a relatively trivial observation that one can directly know by looking at the chart. Whereas, more global observation of ecological significance is *"intense precipitation leads to more soil erosion than less precipitation for a longer duration of time"*. The frequency of such responses was counted for each dataset and then was compared between Excel and our tool.
- **Hypothesis:** Some insights lead users to identify a new ecologically relevant hypothesis and direction of research. These are most critical because they suggest an in-depth data understanding, relationship to ecology, and inference. They lead participants to analyze the data with the next experimental iteration [Heath and Ramakrishnan, 2002].
- **Unexpected insight:** They are those insights that unexpectedly pop up while doing the data exploration. Unexpected insights are additional exploratory or serendipitous discoveries that were not being specifically searched for [Saraiya et al., 2005].
- **New insight:** New insight is a new observation that participants have found during their data exploration. New insight conforms to a discovery about data that is fulfilled by the dataset's information and does not need further examination like hypotheses testing.
- **Time:** The duration of data exploration on Excel as well as on our tool.

10.3.1 Experimental setup

The main aim of the study is to evaluate the effectiveness of the visualization recommendation tool based on its insights generation. The above-mentioned parameters measure this insight about the data on the basis of the used tool. Along with these parameters, a new tool must perform better than the standard tool being used by the biodiversity community. For us, the tool for comparison is Microsoft Excel's graphing module (Figure 34). For this evaluation, we had:

1. Four publicly available biodiversity datasets.
   (a) 577[9] is a seedling dataset from the grasslands. Data dimensions are 60000*27[10]. However, we had to reduce this dataset to only 25000 observations due to latency issues while loading and manipulating it on both Excel and our visualization tool.

[9] `https://data.botanik.uni-halle.de/bef-china/datasets/577`

[10] rows*columns

   (b) 376[11] is a soil erosion dataset from forests. Data dimensions are 1295*40[10].

   (c) 20109[12] is a dataset related to forest management and forest structure attributes. Data dimensions are 150*35[10].

   (d) 24209[13] deals with the enzymatic reactions on different wood types and tree types in forests. Data dimensions are 82*49[10].

2. **Tools:** We created a research prototype for the implementation of our biodiversity visualization recommendation system. It is available to explore online[14]. The front-end of this interface is shown in Figure 28.

   Detailed information about the construction of this tool is available in Chapter 9. As shown in Figure 28, first, the user has to select the dataset, then based on the selection, the visual goals are shown in the left panel. Based on the selected goals, various visualizations are displayed as thumbnail images. Each thumbnail has the name of the visualization, its picture, represented variables, and its short description. Then the user chooses the desired visualization by clicking on the thumbnail. Once clicked, it shows the visualization with a drop-down list to change the dimensions.

   The second tool was the Microsoft Excel charting tool. We used Microsoft Excel 2013 on Windows 10. Within the Excel tool, our target module of comparison was the Microsoft Excel Chart Recommendation module (Figure 34). Users were asked to start their visual exploration via only this module first. If they cannot find their suitable charts or if the recommendation provides nothing, they were allowed to use the manual charting modules or other functionalities available in Excel.

3. **Visualizations:** In Excel charting tool, there are 49 different visualizations. In our tool, we only had 20 visualizations. These visualizations were grouped and placed under different visual goals based on our visualization taxonomy (Chapter 7). The visualizations used for the evaluation are enlisted in Table 10.8. The goal list was shown based on 1) visual goal realization (topic explained in Chapter 7) and 2) suitability of variables' data types filtered by our variable selection algorithm (see Table 5.1).

4. **Participants:** An invitation email was sent to various biodiversity scientists. A personal meeting was requested from all the interested scientists who are at the level of Ph.D. or above and who have some knowledge of Microsoft Excel. The evaluations were conducted between 22nd June till 10th August 2020. Out of all invitations, we received eight interested participants. 50% of whom were Ph.D. students, and the rest had different positions at post-doctorate levels.

[11] https://data.botanik.uni-halle.de/bef-china/datasets/376

[12] https://www.bexis.uni-jena.de/PublicData/ShowPublicXml.aspx?DatasetId=20106

[13] https://www.bexis.uni-jena.de/PublicData/ShowPublicXml.aspx?DatasetId=24209

[14] www.visapps.de

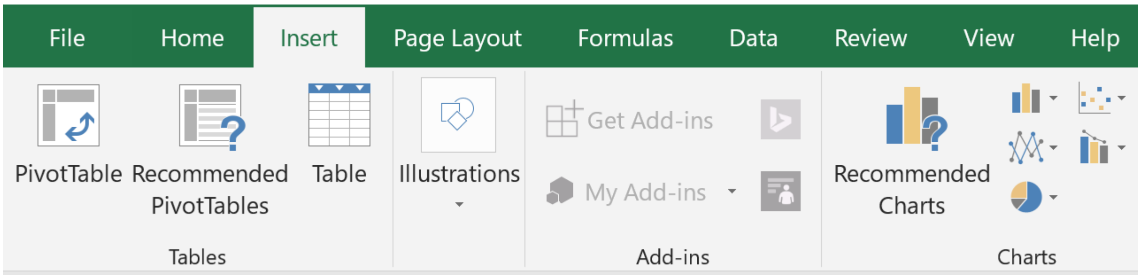


Figure 34: Microsoft Excel Chart Recommendation module

Table 10.8: Visual goals and their respective visualizations

Visual Goals	Visualizations
Distribution	Scatterplot, Hexagonal Binning, Line Chart, Multiline Chart, Area Chart, Stacked Area Chart, Multi Boxplot and Multi Violinplot
Clustering	Hexagonal Binning, Heatmap
Composition	Stacked Area Chart, Stacked Bar Chart, Sunburst Diagram, Tree Diagram, 2D-Pie Charts
Comparison	Bar Chart, Grouped Bar Chart, Stacked Bar Chart, Stack Area Chart, Multi Boxplot and Multi Violinplot
Network	Network Diagram, Alluvial Diagrams
Hierarchy	Tree Diagram, Sunburst Diagram
Overview	Histogram, Kernel Density Plot, Boxplot, Violinplot and Pie Chart

10.3.2 Protocols and measures

To evaluate our tool in terms of its ability to generate insights and compare it with Excel, a set of experimental measures were used:

Lab usability testing: Participants were encouraged to test the tool in a peaceful laboratory environment. A moderator who is also the author of this thesis was present throughout the session, helping them with the overall process, making notes of user interactions, and specific queries.

Interviewing or contextual inquiry: Participants were allowed to perform the task and mention their findings in the think-aloud format by answering questionnaires and directly providing feedback.

Repeated measures within/between groups: Each participant was assigned two different datasets, one to explore on our tool and one on Excel. The participant was asked to perform the evaluation in the provided sequence; for example, the first participant will start with the tool and finish with the Excel. Then this sequence of tools and datasets was shuffled for the second participant. So, the dataset assigned to the Excel with the previous participant would be used by the new participant on our tool. However, now, the participant first has to evaluate Excel and then with the tool. In this way, we evaluated four datasets from 8 participants and have eight evaluations on the tool and the same on Excel. Wherein four times, the evaluation was started with Excel, and for the next four times, the evaluation was started with the tool.

Session recording: Screen recording of each session was done for an in-depth analysis of variables and visualizations used. The initial 10-20 minutes were used to introduce the whole evaluation process and a short tutorial of our tool. No introduction for Excel was needed as it was one of the qualifying factors to participate in this study. First, they were asked to read the metadata for the assigned data and then mention the analytic questions they would like to ask from this dataset. Afterward, they were told to perform their exploration via tool and Excel and mention their insights in the questionnaire. The sample questionnaire is provided in Appendix E. No external tasks were assigned, and participants were encouraged to visually explore the data as per their understanding of the metadata. There was no time limit, and the participants were instructed to continue to examine the data until they are satisfied.

Once participants were done with the exploration, they were asked to fill in the questionnaire regarding all the key findings from the dataset and their overall experience. At the end of the session, they were also asked to provide feedback about our tool's strengths and weaknesses.

10.3.3 Results

Results are presented in terms of the different insight criteria, visualization usage, user background, and experiences. Resultant data sheet is available online[15].

10.3.3.1 Initial questions

For each dataset, users were told to study the metadata first and formulate the questions that they would like to explore in the dataset. They were told to mention these questions in the questionnaire. They were not allowed to explore the tool at this stage.

We have observed that in all the datasets and for all the participants, there were at least two common questions. In Table 10.9 column Count, common denominator 4 implies the number of participants who observed this dataset. So *3/4* for first row means that out of 4 participants, 3 had the same question.

Table 10.9: Common questions from each dataset by the participants

Dataset	Common Questions	Count
577	How do different response variables (height, biomass, leaves) differ between sites (A and B)?	3/4
577	How does density affect height, leaves and biomass measurements?	2/4
376	Correlation or relationship between runoff to species number, LAI, altitude and precipitation.	2/4
376	Identity of species affecting the soil erosion	2/4
24209	How do solid content or biomass are affected by enzymes or enzymatic reactions?	3/4
24201	What is the distribution of the variables?	2/4

As shown in Table 10.9 for dataset number 577 and 24209, more than 75% had the same questions. Moreover, most of these questions fall under the category of correlations and distribution or relationship between two or more variables. In our visualization taxonomy, these goals fall under the broad category of distribution. For all these four datasets, distribution was recommended as the prominent visual goal, by our tool. The sequence of the goal in a list is based on each visualization's

[15] https://github.com/PawandeepKaur/Biodiversity-Visualization-Recommendaton-Tool

probability score predicted by our biodiversity visualization text classifier. Please see Chapter 7 for more details. Therefore, we can confidently claim that our biodiversity visualization classifier has done a good job from this perspective.

10.3.3.2 Evaluation on insight characteristics

In Figure 35, we have presented the scores for different insight characteristics, both for Excel and our visualization recommendation tool. Since we are doing a qualitative analysis; the general comparison of tendencies in the results is most appropriate as it was also in our reference study [Saraiya et al., 2005]. We have further analyzed these insights from different dependent variables (number of visualizations used, the dataset used, and their expertise level) in the later sections. In the following, we have discussed the results of each insight category shown in Figure 35.

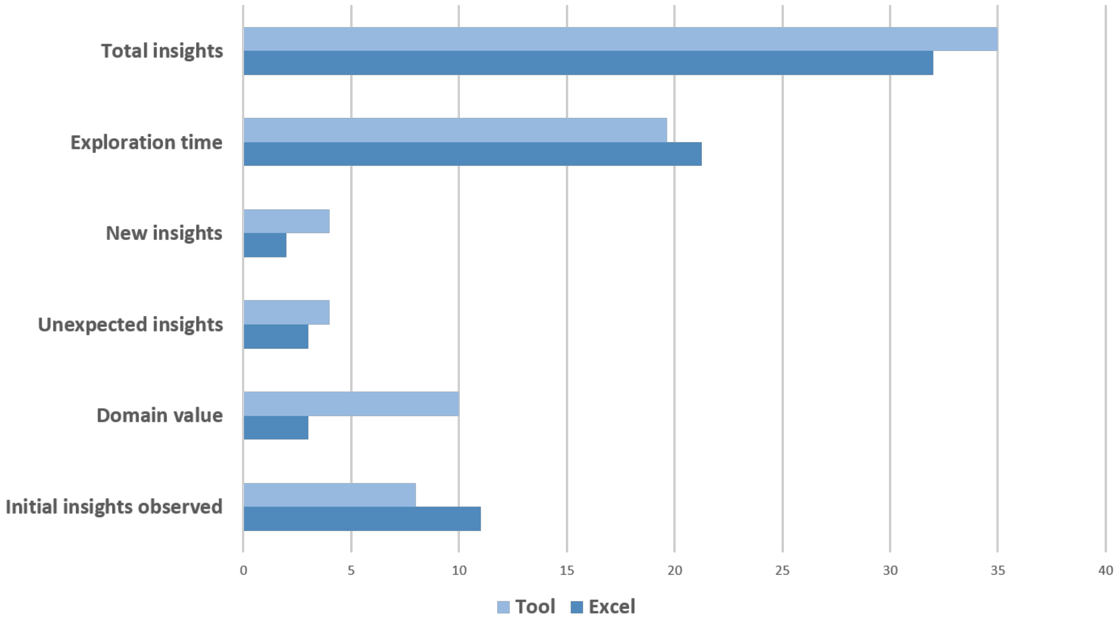


Figure 35: Bar Chart showing the scores of different characteristics of the insight evaluation both for our tool and Excel.

- **Insights:** We counted the total number of insights, i.e., distinct observations from each participant's data for all the datasets. Moreover, we have also compared it with those insights that they initially wanted to get after reading the metadata (Initial insights observed). As shown in Figure 35, Excel has outperformed our tool in this category. Nonetheless, Total insights from our tool are more than Excel. The reason for low Initial insights observed could be, most participants have views against showing only the subset of the whole variable set and not showing all the dataset variables. They said that the recommended subset is relevant for the dataset and provides general or obvious information. Yet, not useful for more in-depth analysis. However, when we looked deeper into this issue, we could not find a consensus on the same variables from even two participants. In any case, we understand that our algorithm needs to be better configured to show as many diverse variables as possible.

- **Domain Value:** It shows the total number of those insights which has some domain value. Domain value insight was incredibly higher for our tool. It shows that the participants using our tool have gained significantly more domain-relevant insights than the Excel.
- **New and unexpected insight:** The total number of new and unexpected insights is higher for our tool than Excel. Though the difference is not significant here, however, it must be noted that most of the participants (6 out of 8) have used different transformations and built-in Excel analytical functions, which was not available on our tool. From our tool, participants gained all these insights by only mouse-operated visual exploration on the raw datasets that were directly visualized without any middleware functions or transformation.
- **Exploration time:** Exploration time is the average total time users spent on the tool until they felt they could not gain more insight. The lower the time, the more efficient it is, or possibly that users gave up on the tool due to a lack of further insight. However, considering that our tool has a lower exploration time but more total insights, the latter does not seem to be the case. Ideally, a visualization tool should provide the maximum amount of information in the shortest possible time [Saraiya et al., 2005]. This criterion is better fulfilled by our tool in comparison to Excel.

Apart from the above criteria, we have also observed that out of 8 participants, 5 (60%) have deduced some hypothetical questions they would like to analyze further. These insights are vital because they suggest future research areas and could result in real scientific contributions. For example, from dataset number 577, one user would like to know, *"What drives allometry between above and below ground?"*, another participant wanted to know from dataset 24209 *"Do all enzymes affect dry mass the same way as Xylosidase does?"*. Some more users mentioned yes to this question but could not formulate it better because they wanted to see more multi-dimensional correlations or further in-depth analysis to better formulate their hypothesis. From 376 dataset, two users observed the same phenomenon: *"when it rains less but more intense, there is more soil erosion than when the rainfall is less intensive for a long time."* However, one user wanted to investigate this further, for which he needed more variables and analytical functions.

Altogether, our tool has resulted in producing the most qualitative insights in lesser time. Thus its performance is better than Excel charting module. Furthermore, we have also observed that the total number of visualization created by our tool for exploration is also more than Excel, which might be the reason for lesser observations in Excel.

10.3.3.3 Insights per dataset

In Figure 36, we have shown the comparison among different datasets based on different insight criteria, on our tool. There are some hints but not very distinctive patterns to hypothesize much. We have observed that dataset number 20106 has the lowest value in all the insight categories except the Initial insights observed. Moreover, the average time spent on this dataset is also the lowest in comparison to other datasets. The reason could be, this dataset has the lowest number of the recommended variables, i.e., 8 (see Table 10.5). We have found that users have spent more time exploring dataset 376. Furthermore, 376 has not many Total insights but has the highest insights with domain value. Total number of recommended variables for 376 is 15, which is similar to 24206, i.e. 16. Dataset 24206 had most Total insight and Initial insight observed. Moreover it is second to 376 in Exploration Time.

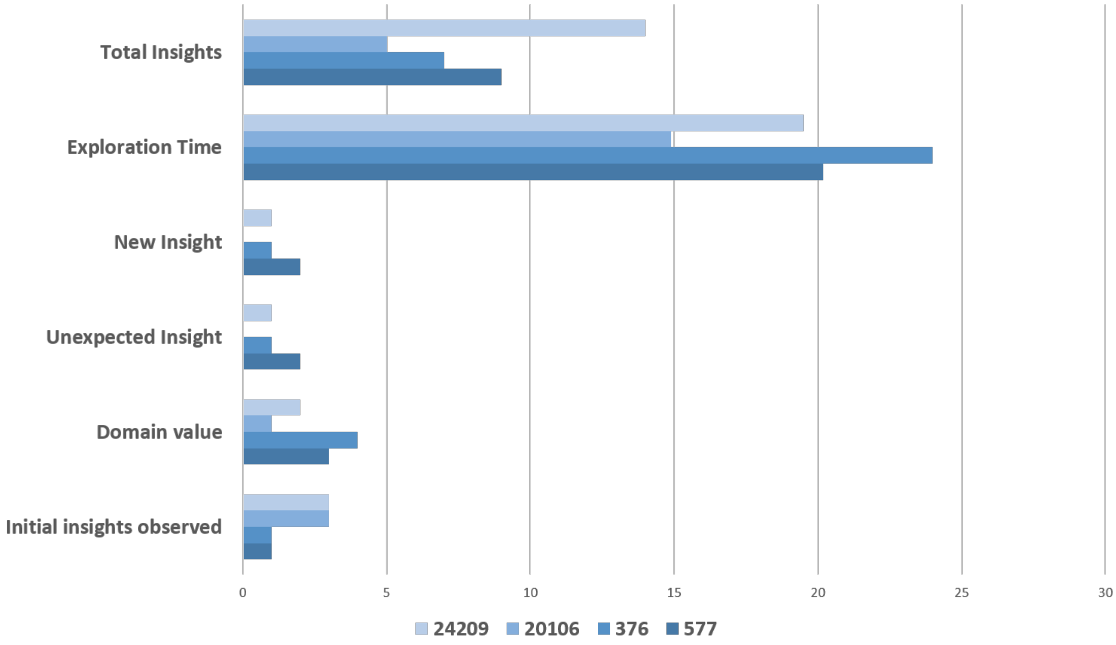


Figure 36: Bar chart showing the scores of the different insight criteria for each dataset.

10.3.3.4 Insights per expertise

In Figure 37, we have compared the insight scores on our tool based on users from PhD and Postdoctoral levels. By chance, out of eight evaluated users, four were PhD students (PhDs), and four were Postdoctoral researchers (Postdocs), giving us a 50% ratio. There are some intriguing elements to notice from Figure 37. Firstly, though PhDs have spent less time with the tool, they were able to get a higher Total Insights. This further could be the reason that they had more hypothetical inquiries and more domain value questions. We have observed that the PhDs were more motivated to do analyses based on their exploratory questions, and postdocs, in general, were more interested in doing exploration based on the tool characteristics. For example, they used more different types of visualization than PhDs (see Table 10.10). The second thing to notice here is that both groups have the same value for New insight and Unexpected insight. We were expecting these values higher for Postdocs because of their certain level of expertise in understanding the domain and the dataset. On the other hand, Postdocs were able to get more initial insights that they have planned from the exploration than PhDs. Overall from this analysis, we can say that this tool has been more helpful to the PhDs than Postdocs, which we had not expected. In general, people with more expertise will do more intense analysis than the other group. However, Postdocs were more efficient in exploring their Initial insights.

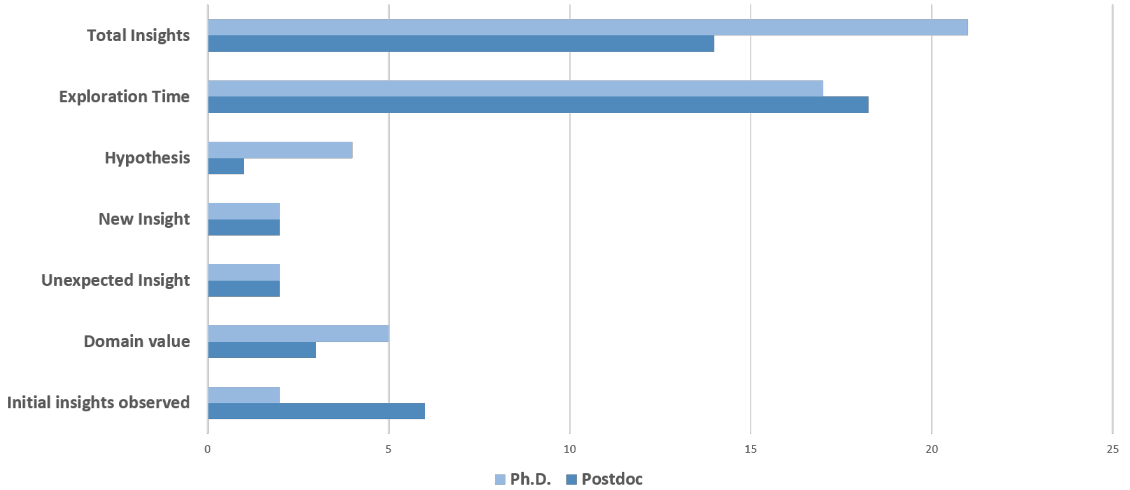


Figure 37: Bar chart showing the scores of the different insight criteria between PhDs and Postdocs.

10.3.3.5 Insights based on visualizations and visual goals

To analyze the results based on visualizations and visual goals, we have collected the data from the questionnaires, wherein they have mentioned the visualizations they have used to infer observations from the datasets (step 3 in Appendix G). These observations are also included in the Total Insight criteria mentioned in the previous section. Figure 38 shows that our tool has provided more distinctive visualizations for the users to explore various aspects of the dataset in lesser time than Excel (see Figure 35).

Furthermore, we have observed that the Postdocs have explored more variant visualizations and visual goals than PhDs (see Table 10.10). This could also be due to their expertise in the domain as well as with the visualizations. Moreover, in comparison to PhDs, during the evaluation, Postdocs had enquired more questions regarding the usefulness of the different visualizations. From Table 10.11, we know that, scientists have used scatterplot the most both with Excel and our tool. After the scatterplots, the same number of bar charts and histograms have been used.

Table 10.10: Total number of distinct visualizations used by PhDs and Postdocs for data exploration on our tool. Scores are shown based on the respective visual goals of the visualizations.

Visual Goals	Postdoc	PhD
Distribution	4	4
Comparison	1	2
Clustering	2	0
Network	1	0
Composition	1	0
Overview	1	0

Out of 49 available visualizations in Excel, only 5 distinct visualizations were helpful. Our tool had 20 visualizations, and 12 were able to provide insightful observations about the data. We can only understand the one reason behind this: ease of use of our tool. In our tool, the visualizations were already created for

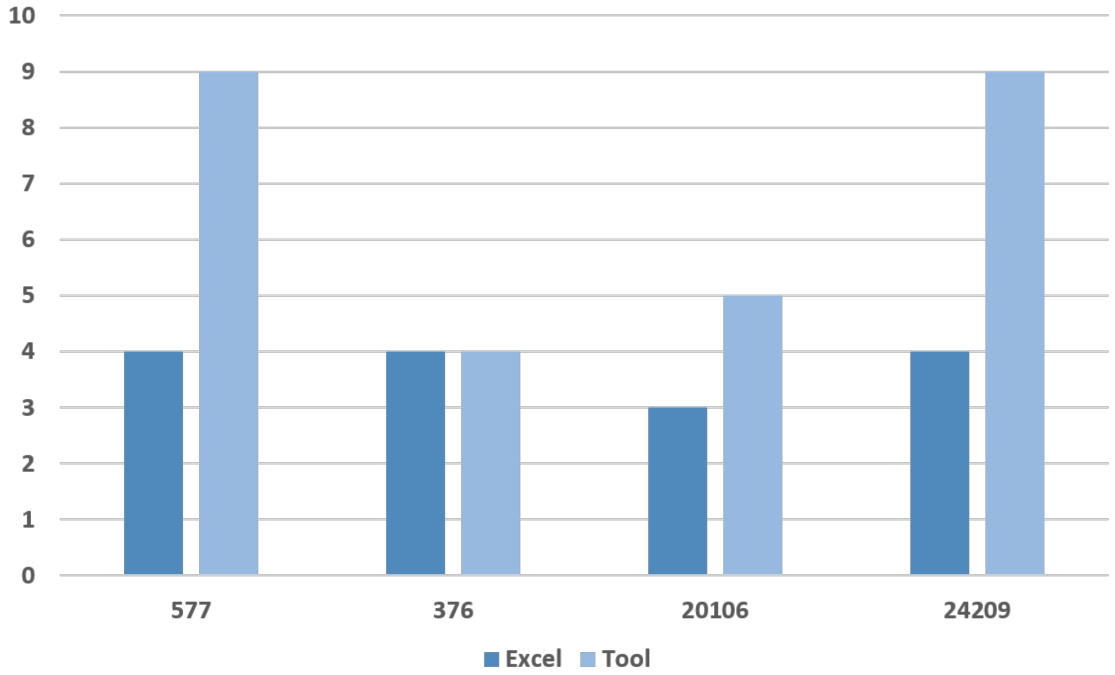


Figure 38: Total number of visualizations used by the participants to get insights from the datasets.

them, and they only had to change the dimensions. It was not the case with the manual charting tool like in Excel. In the manual creation of the visualization, if the variable datatype does not match the requirements of visualization type, then the visualization either cannot be created or will be of no use. For automatic creation of the visualisation, we have already supported this visual mapping functionality in our tool (see Table 5.1).

Table 10.11: Frequency distribution of different visualizations used in exploring the datasets on Excel and our tool.

Visualization	Excel	Our tool
Scatterplot	7	8
Bar Chart	3	3
Histogram	2	2
Boxplot	1	3
Grouped Bar Chart	1	0
Hexagonal Binning	0	2
Stacked Bar Chart	0	2
Pie Chart	0	1
Violin Plot	0	1
Line Chart	0	1
Multi Line Chart	0	1
Multi Boxplot	0	1

From Excel observations, we created a list of visual goals based on the visualization sequence mentioned by the participants in the questionnaire. The visual goals were derived based on our visualization taxonomy. These results are compared with the recommended visual goal list from our tool. Then, we have used the same Precision and Rank-biased Overlap (RBO) metrics from the Evaluation 1 (see Section 10.1). The only difference is, in Evaluation 1, we calculated the scores based on a visualization list and here we calculated them based on visual goals. The average Precision is 56%, and the average RBO score is 86%. The scores from Evaluation 1 is 62% for average Precision and 61% for average RBO. The new precision score is less in comparison to our Evaluation 1. The improved RBO score states that, the methodology of visual goals derivation has overall improved the performance of the system. The improved scores could be interpreted as though the visualizations are different, but their cumulative visual goals are similar, which matter more than the visualization itself. For example, Scatterplot, Histogram, Boxplot, Hexagonal Binning derive the common goal of data distribution.

10.3.3.6 Insights based on the variable selection algorithm

We have observed that overall all participants have mentioned that the recommended subset of variables is a prominent representative of the dataset. However, it lacks more profound variables of interest to the domain experts - for example, variables related to the causation of some ecological phenomena depicted in the dataset. All our participants felt that some of the variables of their interests are missing. Moreover, there is not much consensus on a particular missing variable. For example, for dataset number 577, the first participant said that because of the important ecological terminology "Jansen-conell effect" in the metadata, the variable "density" is significant. However, the second participant on the same dataset did not show any interest in that variable. He mentioned that he would like to see "distance" in the subset because it is important for exploring this dataset. This is a practical proof of fewer scores for the IRR or consensus metrics in evaluating the variable selection algorithm presented in Table 10.7. There are repeated examples of similar cases with the other datasets. However, one typical critic was that the recommended subset had missing variables that they would like to explore.

Therefore, we analyzed the session recordings in Excel and have tried to see which variables have been used for each dataset. We compared this result to our recommended subset, and the scores are presented in Table 10.2.

Table 10.12: Table shows for each evaluated dataset: Precision, Sensitivity and Accuracy scores.

Dataset	Precision	Sensitivity	Accuracy
376	0.2	0.29	0.48
577	0.6	0.49	0.70
24209	0.35	0.58	0.75
20106	0.05	0.30	0.26
Grand Mean	0.30	0.42	0.55

Compared to the evaluation results presented in Table 10.6, there is a little improvement over sensitivity or recall and accuracy scores. Because of the limited participants, this time, we did not do an inter-rater reliability check.

10.3.4 Overall feedback

At the end of each session, participants were requested to comment on their overall experience with our Biodiversity Visualization Recommendation Tool. The comments are provided in Appendix H.

- **Feedback on recommended variables**: Majority participants believed that the subset of the variables was interesting and showed the general project design. However, all mentioned that some of the needed variables were missing. Many have provided feedback on the improvement of the variable selection algorithm in the following ways:

  - For ecological datasets, the variable selection algorithm should be learned based on the ecological phenomenon (for example Horizontal Heterogeneity, Jensen Effect). As for each phenomenon, certain variables are significant. Missing such variables in the recommendation will be a loss of information.
  - The algorithm should include all Spatio-temporal variables as they show a specific trend. It should consist of all categorical variables as they show the experimental setup. For measurement or quantitative variables, apart from a context check, a collinearity check should be done. A better variable selection algorithm will avoid having too many collinear variables and would have more diversity.
  - Such algorithms should be trained on variable names in the biodiversity publications (better from similar projects) and the described biodiversity concepts within the captions.

  Our qualitative results are not adequate for the CVS algorithm; however, due to high total insight scores and good scores of hypothesis and domain value, we consider the current state of CVS is adequate to be used in the production environment. For future work, the CVS algorithm can use the solution mentioned above. Moreover, it can be improved by including the element of active and interactive learning. Wherein, an algorithm can fine-tune and train itself from the user's feedback at a run time.

- **Feedback on recommended goals**: After observing our participants' visualization usage pattern, we have found that every user has their sequence of data exploratory tasks. Some participants started their exploration through comparative analysis (Comparison goal), some through univariate analysis (Overview goal), and mostly from the data distribution (Distribution goal). Due to the quantitative nature of all the datasets (which is common in the biodiversity domain), all were interested to see the distribution and correlation among variables (see Table 10.10). Some users wanted to do more intensive correlation analysis and had reported that our tool needs more multivariate correlation and clustering visualizations. Some of the mentioned visualizations were Scatterplot Matrix, Coplots and Parallel Coordinates. Overall, the users found our tool's design, i.e., goal-based visualization exploration, very intriguing and helpful.

- **Feedback on visualizations**: In total, we had included 20 visualizations within the tool. Out of which, 12 visualizations were used by our participants, giving us a ratio of 60% of usage. There was some critical feedback regarding the features of the visualizations. For example: zooming out feature of an interactive treemap is not intuitive, size customization attribute for scatterplot or interactive colormap for scatterplot is missing and hover option for boxplot is missing. Many users found our system of recommending similar visualizations based on the visual goals very informative. One user said, *"though this data, if shown in the boxplot, would not make much sense; however, because of the violin plot (which shows kernel density on different values), I can read the data well. Had it not been suggested, I would not have thought about this plot."* Some said that *"I am not a great friend of the pie chart. However, as the pie chart shows the categorical distribution, I can see how balance the dataset is. I would not have used a pie chart to see the structure of the dataset"* (see pie chart created by a participant in Figure 39). One user said that the alluvial diagram shows the complete experimental setup for data (Figure 40). *"I can see which plot have what types of and how many species planted."*

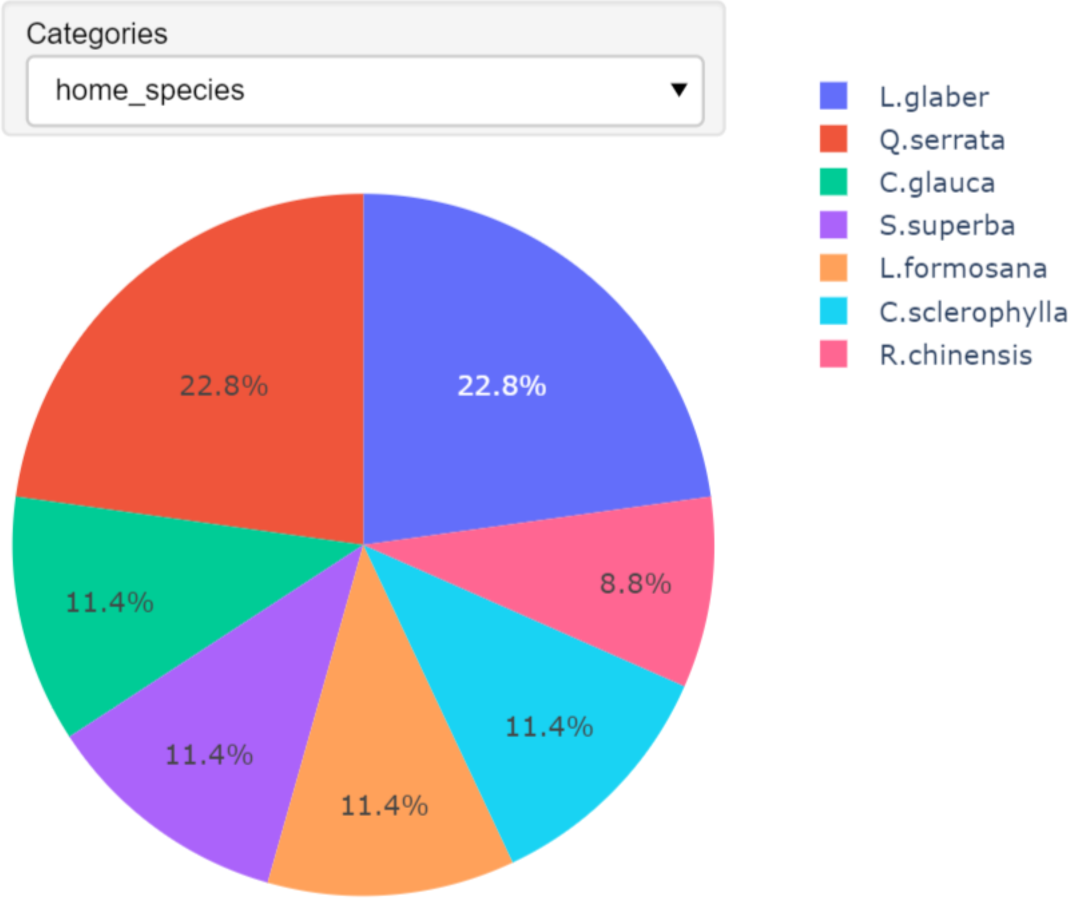


Figure 39: Pie Chart example from evaluation

- **Feedback on Tool**: Most users found the tool intuitive, easy to use, faster, and user-friendly. They liked the mouse-driven environment of the tool. One said, *"it identified outlier without doing deep exploration just by plotting with the mouse"*. They liked how multiple variables were easy to be selected by the drop-down list. They also liked that various visualizations were presented to

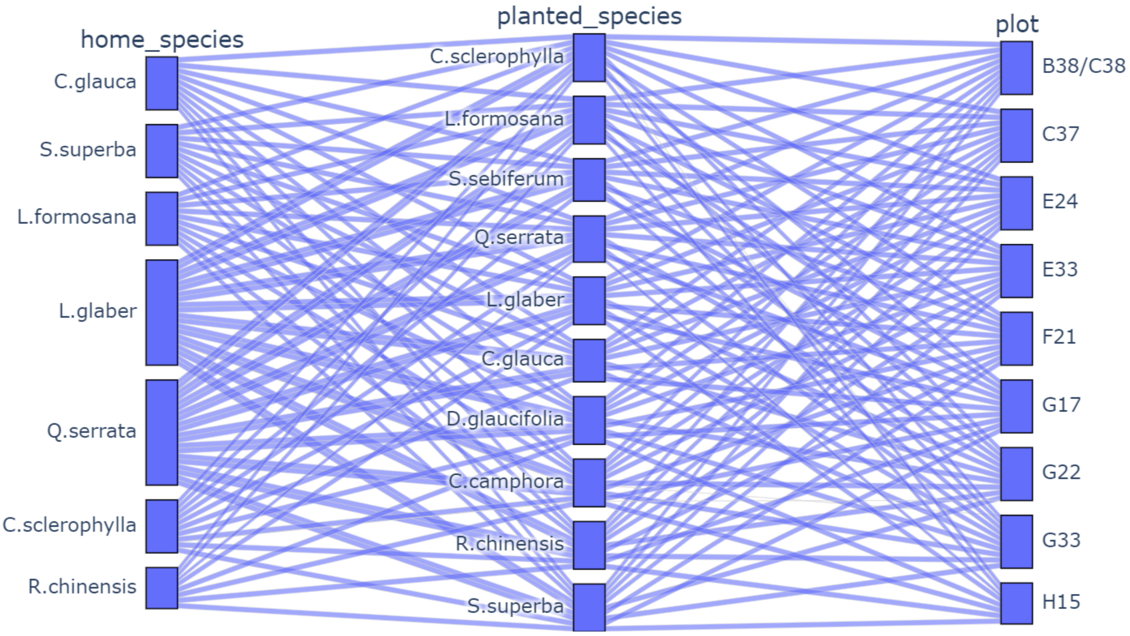


Figure 40: Alluvial Diagram example from evaluation

them with different roles. They acknowledge that most of the visualizations are new to them, and with regular use, they will get to know about it. One said, *"I liked how the tool gives ideas to visualize specific variables and data."* Others said it completely fulfills the question of *"What kind of graph to apply for the variables?"* For its improvement, the users have provided some feedback:

- It should also provide quantitative summary statistics like in R.
- The variable description should also be shown on the tool, along with the variable names. As some variables are shorthanded, one has to look to the variable names in the metadata files repeatedly.
- For each plot, it should also show how many observations the plot is based on?

10.3.5 Summary and discussion on the evaluation of the Knowledge-based Visualization Recommendation System

For the qualitative evaluation of our Knowledge-based Visualization Recommendation System, we selected 8 participants all from PhD level and above. Participants were asked to visually explore the data, both on Excel as well as on our tool. To evaluate and compare the results based on the gathered insights from the dataset, we used some of the insight criteria from the study by [Saraiya et al., 2004]. These insight criteria are: total number of observations, insights with some domain value, insights lead to the formation of some hypothesis, new and unexpected insights and total time spent. We found that our tool has outperformed Excel on almost all of the insight criteria. The only exception is with the Initial insights observed, wherein Excel has scored better (Figure 35). This is because only subset of the variables were available in our tool in comparison to Excel. We observed that in comparison to Postdocs, PhDs have spent less time on the tool and are able to get higher number of insights. Moreover, their insights had more domain value and contain more hypothetical inquiries (Figure 37). The reason we observed of this behaviour is that PhDs were more motivated to do analyses based on their exploratory questions, and postdocs, in general, were more interested in doing exploration based on the tool characteristics. That was the reason they have explored more variant visualizations and visual goals than PhDs (see Table 10.10). In comparison to Excel, our tool had presented more distinctive visualizations (Figure 38) to the users to explore various aspects of the dataset in lesser time. We have observed that overall all participants have mentioned that the recommended subset of variables is a prominent representative of the dataset. However, it lacks more profound variables of interest to the domain experts. All our participants felt that some of the variables of their interests are missing. Moreover, there is not much consensus on a particular missing variable. This is practical proof of fewer scores for the IRR or consensus metrics in evaluating the variable selection algorithm. However, domain experts provide different strategies from the improvement of this algorithm. They are presented in Section 10.3.4. Due to the quantitative nature of all the datasets (which is common in the biodiversity domain), all participants were interested to see the distribution and correlation among variables (Table 10.10). Overall the users found our tool's design, i.e., goal-based visualization exploration and its mouse-driven environment, very intuitive, intriguing and helpful. Apart from them there were some recommendation of improvement for the inclusion of more multidimensional visualization, better customization of some of the visualizations and overall tool configuration.

The big challenge we faced in doing the evaluation is the lack of studies to compare our results. This is due to this tool's very nature, our research, and limited insight-based qualitative studies. The study on which our qualitative metrics about insights have been inferred [Saraiya et al., 2005] is also an old study, and we could not find any recent advances in this work. However, through this study, we can fulfill some of the shortcomings of the domain-based visualization research area, which they have also mentioned in their paper [Saraiya et al., 2005]. Some of the issues that we have addressed are :

- **Higher-level domain-based interface:** Our system is a high-level visualization interface that is completely dependent on the biodiversity domain.
- **Interactive design that emphasizes consistent, usable interaction:** Our tool is interactive with the mouse-driven environment.
- **Clear visual feedback:** Our tool provides direct visual feedback for each user selection and interaction.
- **Multiple representations:** Our system provides multiple representations of the same dataset as well as alternative visualizations.

Furthermore, our tool and our research fulfill many of the visualization requirements gathered from the domain users at the requirement analysis phase in Chapter 2:

- It reduces the visualization selection dilemma by automatically selecting the variables based on data type and domain knowledge.

- It tries to bridge the gap of knowledge by offering different alternatives of similar visualizations.
- It helps visualize the large datasets by automatically selecting the contextual relevant and interesting variables.
- Instead of being prescriptive, it gives a wide range of visualizations and variables to explore.
- It is easy to use with an intuitive workflow that clusters the visualizations based on their representative goals.
- It efficiently helps in data exploration, which can further lead to data analyses.

10.4 Summary and Discussion

This chapter presents the evaluations of three core components or contributions of our research, i.e., Biodiversity Visualization Text Classifier, Context-aware Variable Selection Algorithm, and our comprehensive evaluation on the Knowledge-based Visualization Recommendation System. The first two components were evaluated by comparing the algorithms' results with the ground truth data collected from the biodiversity participants. The third component was evaluated by conducting a qualitative evaluation of the system based on the overall data insights generation.

The evaluation of the Biodiversity Visualization Classifier showed the score's dependency on the quality of the provided metadata files. The mean RBO and Precision scores vary from 48% to 80% based on the text's length and quality in the metadata files. The better the quality is, the good the scores are.

The Context-aware Variable Selection Algorithm evaluation showed neither good accuracy (54%) nor good inter-rater reliability or agreement scores (16%). To evaluate this algorithm, we have collected the ground truth data from 41 different biodiversity scientists. Then we compared the collective results from their responses to the ones from our algorithm. Doing so, we have realized that their mean agreement on one dataset is very low. This is also reflected in not so good precision-recall scores.

Our qualitative evaluation of the overall system with the biodiversity scientist had shown much better results than evaluating the first two components. Compared with Microsoft Excel graphic system, our tool has produced the most qualitative insights in lesser time. 60% of the participants were able to form a certain hypothesis which they would like to investigate further. Furthermore, they were able to create more visualizations and found more observations in comparison to Excel. Most users found the tool intuitive, easy to use, faster, and user-friendly. They liked the mouse-driven environment of the tool. Many users found our system of recommending similar visualizations based on the visual goals very informative.

From these evaluations, we conclude that our overall quantitative scores are not very impressive. However, our good qualitative results permit us to tune the software and use it in our BExIS system.

Chapter 11

Conclusion

In this thesis, we have discussed the problem of visualization selection, which occurs due to the availability of unlimited choices of visualizations or chart types, due to large and multidimensional datasets and due to the complexity involved in creating a visualization. We have discussed these issues in detail in Chapters 1 and 2. As a solution to these problems, visualization recommendation systems are used. A review of current literature revealed that, although there are plenty of visualization recommendation techniques available today, visualization science still lacks in providing domain integrated recommendation solutions. We have discussed the need for such solutions for specialized domains in Chapter 3. Understanding the issues our biodiversity users face in their visualization process and being aware of the shortcomings of current visualization recommendation technologies, we proposed to provide a domain knowledge-based visualization recommendation solution for the visual exploration of biodiversity datasets. Furthermore, this research increases awareness for using domain knowledge in visualization recommendation systems. Thus, this research contributed to visualization science by demonstrating various approaches to integrate specialized domain knowledge into the visualization recommendation process.

Our visualization recommendation model is based on the biodiversity domain knowledge and the context of the data. It is one of the core contributions of this thesis. Based on community feedback, we came to this conclusion that the visualization tool needed for this community should include an element of support in the visualization selection process. Moreover, we built this model based on the target community's domain knowledge to obtain visualization suggestions that closely corresponded to the user's knowledge. As it was impossible to collect needed mass knowledge directly from biodiversity scientists, we relied on knowledge extraction from publications. This resulted in the construction of the Biodiversity Visualization Text Classifier, which represents the second main contribution of this thesis. These are the first visualization classifiers that can suggest suitable visualizations by only reading the text from a particular domain. To provide a solution for visualizing high dimensional datasets, we have created our own ad-hoc Context-aware Variable Selection Algorithm. As current visualization recommendation studies do not provide any support to visualize high-dimensional datasets, we believe that our work will encourage future research to develop further techniques to visualize high dimensional data in visualization recommendation systems.

Our research is very user-centric and has tried to focus on domain users at the core of our research: we surveyed domain users at the time of the requirement gathering phase; each module of our system was thoroughly evaluated by domain users. We have also performed both qualitative and quantitative evaluations of our final complete system. For quantitative evaluation, we used current state-of-the-art evaluation techniques. In qualitative evaluation, we measured our system's efficiency based on users' perception and overall insight collection.

11.1 Challenges and Future Directions

This section summarizes two of our most important contributions, including the challenges we faced during their constructions and future directions for improving such work.

11.1.1 Biodiversity visualization text classifier

Summary: The Biodiversity Visualization Text Classifier was constructed to derive different chart type suggestions from a biodiversity text automatically. We also considered the charts' conceptual similarities for chart classification and the visual similarity of different chart types. We manually labeled the chart images and captions from biodiversity publications. We trained both the image and chart classifiers on this training data. From the best results of these two classifiers, we have incrementally trained our text classifiers. This resulted in an average F1-score of 92.2% from the assembly of binary chart classifiers. Next, our classifier was evaluated by the domain users. The text classifier performed significantly better on those questions with 80% scores, where the quality and quantity of the textual data was good. The high level of agreement between predicted and human results indicates that the classifier learned the concept that fits the human understanding of the data.

Captions have been considered for the first time in visualization research for chart classification. Before this work, chart type identification was done only based on the image and its pictorial elements. A comparison between our results and other studies (see Figure 20) demonstrates that a conceptual/semantic chart classifier can efficiently differentiate between chart types that are visually similar (e.g., column chart and histogram) and is as efficient as an image classifier. Such classifiers can be used for different purposes. One purpose that is described in our research is the creation of knowledge-based visualization systems.

Future research: Classifying chart types from caption data is still in a novice state. We identified different use cases in which research in caption analysis could be beneficial for visualization as well as for linguistics research:

- **Visualization research:** 1) A machine learning model trained on the visualization captions can be evolved and also used by other users for different domain knowledge-based visualization products, 2) a text classifier trained on different chart types can be used for tagging, indexing, and searching documents, 3) it could be a valuable source for future theoretical visualization research problems [Chen et al., 2017] like the creation of visualization ontologies based on classified visualization concepts.

- **Linguistics research:** Research on caption classification will help: 1) to better understand the requirements of classifications on concise and convoluted texts, 2) to study the influence of domain-specific languages on classification and possibly exploit domain-specific regularities to improve classification results, and 3) to find effective ways to integrate domain expert knowledge into the classification process.

Challenges: For an enhanced semantic chart recognition systems, several problems need to be solved:

- As our classifier was only trained on biodiversity texts, we do not know how well it will perform on general text or text from other domains. Studies are needed in which a text classifier trained in one domain can be generalized to other domains. The application of transfer learning[1] has already produced remarkable results in the field of computer vision. However, studies on the application of transfer learning on textual data and especially on decision trees and random forests are scarce. These questions are out of scope for our research and have to be addressed in future studies.

- Apart from only using captions in the training process, using text from the other parts of the publication referring to the chart images, could improve results. Moreover, if the data is enriched with more semantic knowledge like synonyms, concurrent words, domain-based ontological concepts then better classification accuracy can be achieved. As this research was the first of its kind in visualization caption classification, we did not apply these techniques in our experiments. Based on preliminary results, we are confident that further data enrichment will only improve the classification results.

- Real datasets are not always as homogeneous as the training data used for the construction of the classifier in our work. Therefore, the accuracy of the classifier might decrease when applying them to real datasets. More studies are required to understand the common variations found in visualization images and captions—for example, hybrid visualizations, multi-embed visualizations, grammatical irregularities or out of vocabulary text.

11.1.2 Context-aware variable selection algorithm

Summary: For the efficient visualization of high-dimensional datasets based on a few but relevant and salient features or variables, we have followed the approach of feature selection in machine learning models. However, unlike data-driven variable selection approaches, where the importance of variables is counted based on the data distribution or other statistical features, we applied the domain-specific variable selection approach. The approach uses the available domain information to retrieve important keywords from the current metadata and other metadata files from the same project. These keywords are used to filter out the relevant variables based on variable definition. The quantitative evaluation of this algorithm resulted in moderate scores, i.e., accuracy only 55 %. However, at the same time, the inter-rater agreement result was also very low, i.e., only 16%. It means that there is only

[1] `www.tensorflow.org/tutorials/images/transfer_learning`

16% of agreement between all 41 participants. The qualitative results and feedback from our participants yet were good, encouraging us to use this algorithm in the BExIS system.

Our literature review regarding previous visualization recommendation studies revealed that the data's domain knowledge and context are not directly considered to filter salient variables from the dataset to visualize. In these studies, either the complete variable set is allowed to visualize, or the users have to provide their interest at runtime interactively. Recent studies about machine-learning based visualization recommendation (built on rule-based trained models) select important variables for visualization. However, these studies hardly include domain knowledge or context in their approaches. Our research has provided a first step in this direction: the context of the data from the available domain knowledge is used to filter important variables to visualize. Currently, we are in the process of analyzing feedback from our domain users to improve this algorithm.

Future research: Some additional aspects could not be dealt within our timeframe and scope.

- **Machine learning-based context-aware variable selection algorithm:** Our current algorithm produces results based on the quality of the text in the metadata files, the project metadata files, and the known biodiversity terms. It does not use any pre-learned concept in calculating the results. Training the algorithms on the different biodiversity concepts and their related textual vocabulary (for example, which variables and terms are more often used to describe a beta diversity) might be an option to improve results.

- **Context-aware visual data summarization:** While working on this algorithm, we realized that apart from filtering the relevant variables from the dataset, we can also provide a visual summary of these variables. This summary shows which variables are highly connected based on the respective biodiversity concept. In Figure 41, we have presented an example from one of our experiments.

  Figure 41 illustrates the relationship between the filtered variables: there are two prominent entities in this dataset, i.e., *soil* and *plot.* The soil has a total *carbon* content (*c_t*) measured at a certain *depth* level. Each depth has a specific label (*depth_lb*) in the dataset. Furthermore, in this dataset, there is information about soil acidity (*ph_h2o*). Plots of a specific *size* have been assigned a special tag (*csp*). *Nitrogen* has been measured in two forms, i.e, *n_t* (nitrogen total) and (*c_n*) ratio of carbon and nitrogen. Compound *kci* (Potassium Chloride) has been measured in the form of *ph_kci.*

  This example indicates that, we can visually show the complicated relationship between the data variables in the dataset. This approach of data summarization could be very helpful to understand very high dimensional datasets.

- A domain-specific variable selection algorithm can also be trained at run time by using interactive machine learning techniques: if the user is not satisfied with the recommended variables or wants to include more variables, the algorithm can be trained based on the new variable selection from the user. This could provide better variable recommendations for the same datasets for future sessions. This technique of tagging variables to data and visualizations

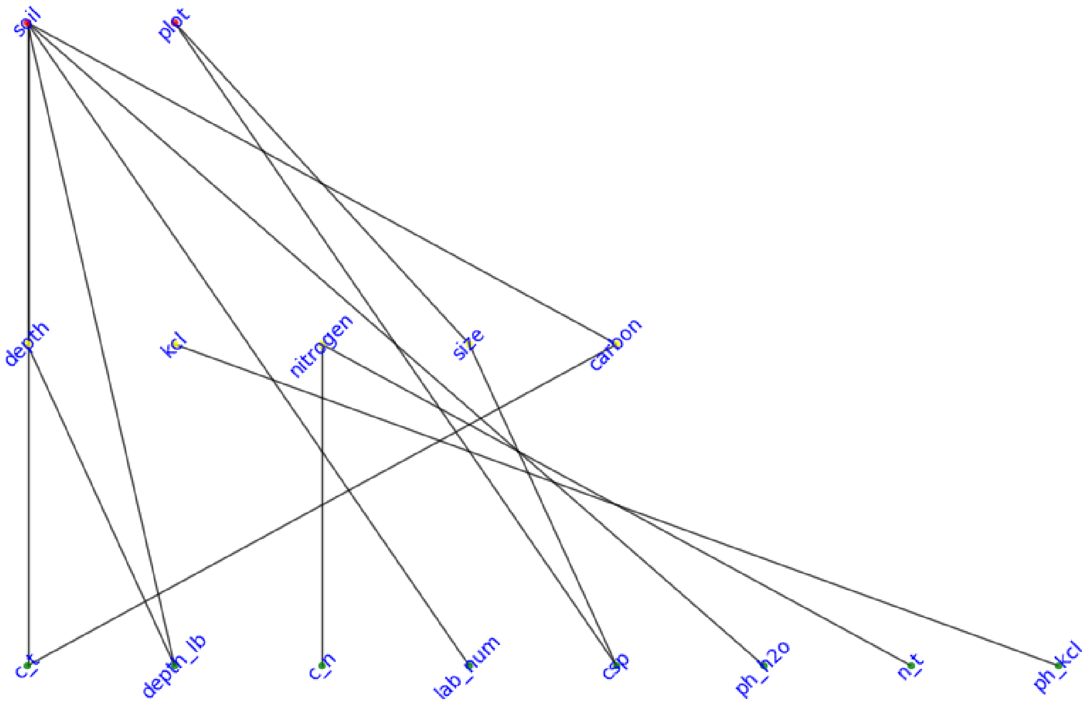


Figure 41: An example of context-aware visual data summarization.

has already been successfully used in previous machine learning-based visualization recommendation studies. However, these studies have only used offline training. Online training allows the efficient integration of user knowledge into the system's knowledge base at runtime.

Challenges: The biggest challenge we faced concerning this algorithm was linked with its evaluation. There are no benchmark techniques to evaluate domain-specific algorithms. In our study, we evaluated it with the same metrics used for feature selection algorithms in machine learning. Another challenge is related to the various description standards in the metadata of the biodiversity datasets. Use of more unconventional short-forms affects the overall understanding of the data.

11.2 Experience from User Study

We followed the principles of user-centric studies throughout this research. We contacted domain users through multiple surveys, meetings at conferences, interviews and informal chats. This knowledge helped us to construct the system according to the user's needs. This concept has also indirectly provided us with helpful cues in bridging the knowledge gap between the domain scientists and the computer scientist for future domain-specific software projects. We published results from this experience in a collaborative paper [Jänicke et al., 2020] with other scientists. Some of the findings are:

- **Understanding the domain:** Researchers in computer science and visualization, are often unaware of research interests and current workflows in the targeted domain. For successful construction and implementation of visualization systems, it is necessary to attain at least a basic understanding, ideally, fascination, for the respective domain field. We should also try to understand how domain experts use state-of-the-art tools, if present, for their daily work.

- **Early prototyping:** It can be beneficial to develop a basic functional prototype before gathering requirements from domain users. Being provided with a real system instead of an abstract concept, they are better able to give feedback, and they are better able to map their existing workflows and reflect on conceptual gaps. After completion of the system, they are also more indulged and provide input on how to amend it to make it more suitable for their requirements. This early prototyping approach has been proven successful, especially when targeting technically inexperienced scholars.
- **Engage in the domain:** A ready-to-use tool designed during an interdisciplinary project does not necessarily reach domain users beyond the project participants. It is useful to present applicable solutions at conferences of the targeted domain. This not only potentially increases the user base of a tool, but also reveals existing visualization gaps in the domain. Additionally, we can offer our expertise for related tasks leading to novel research directions in our field.

11.3 Work-in-progress

As evident from our good qualitative results presented in Chapter 10, the approach presented in the thesis about recommending visualizations based on the domain knowledge is convincing to a large extend. However, still significant further development is needed. Currently, we are in the process of the integrating recent aspects from the user feedback in the system and implementing it in our BExIS environment. We have also planned to work with other biodiversity projects and configure our recommendation system to visualize their datasets. Apart from that, we would like to use the current interface to not just support biodiversity users but also users of other domains. We would like to reduce the dependency of the recommendation system on domain specific aspects and would try to include other non-domain features into the recommendation logic.

Bibliography

[Abadi et al., 2016] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., and Zheng, X. (2016). Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16, page 265–283, USA. USENIX Association. doi: `10.5555/3026877.3026899`.

[Aggarwal and Zhai, 2012] Aggarwal, C. C. and Zhai, C. (2012). *Mining text data.* Springer Science & Business Media. doi: `10.1007/978-1-4614-3223-4`.

[Agrawal, 2013] Agrawal, R. (2013). Comparing ranked list. Available at: `http://ragrawal.wordpress.com/2013/01/18/comparing-ranked-list`.

[Amar et al., 2005] Amar, R., Eagan, J., and Stasko, J. (2005). Low-level components of analytic activity in information visualization. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, page 15. IEEE. doi: `10.1109/INFOVIS.2005.24`.

[Amar and Stasko, 2004] Amar, R. and Stasko, J. (2004). A knowledge task-based framework for design and evaluation of information visualizations. In *IEEE Symposium on Information Visualization*, pages 143–150. IEEE. doi: `10.1109/INFVIS.2004.10`.

[Auer et al., 2011] Auer, T., MacEachren, A. M., McCabe, C., Pezanowski, S., and Stryker, M. (2011). Herbariaviz: A web-based client–server interface for mapping and exploring flora observation data. *Ecological Informatics*, 6(2):93 – 110. Elsevier. doi : `10.1016/j.ecoinf.2010.09.001`.

[Balaji et al., 2018] Balaji, A., Ramanathan, T., and Sonathi, V. (2018). Chart-text: A fully automated chart image descriptor. *arXiv*, arXiv:1812.10636. Available at: `https://arxiv.org/ftp/arxiv/papers/1812/1812.10636.pdf`.

[Beilschmidt et al., 2017] Beilschmidt, C., Drönner, J., Mattig, M., Schmidt, M., Authmann, C., Niamir, A., Hickler, T., and Seeger, B. (2017). Vat: A scientific toolbox for interactive geodata exploration. *Datenbank-Spektrum*, 17(3):233–243. doi : `10.1007/s13222-017-0266-5`.

[Berry and Kogan, 2010] Berry, M. W. and Kogan, J. (2010). *Text mining: applications and theory.* John Wiley & Sons. isbn: 978-0-470-74982-1.

[Bertin, 1983] Bertin, J. (1983). *Semiology of graphics; diagrams networks maps.* Esri press. isbn: 9781589482616.

[Bochare et al., 2014] Bochare, A., Gangopadhyay, A., Yesha, Y., Joshi, A., Yesha, Y., Brady, M., Grasso, M. A., and Rishe, N. (2014). Integrating domain knowledge in supervised machine learning to assess the risk of breast cancer. *International Journal of Medical Engineering and Informatics*, 6(2):87–99. Inderscience Publishers. doi: 10.1504/IJMEI.2014.060245.

[Bolón-Canedo et al., 2015] Bolón-Canedo, V., Sánchez-Maroño, N., and Alonso-Betanzos, A. (2015). *Feature selection for high-dimensional data.* Springer. doi: 10.1007/978-3-319-21858-8.

[Boser et al., 1992] Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152. ACM. doi: 10.1145/130385.130401.

[Bouali et al., 2016] Bouali, F., Guettala, A., and Venturini, G. (2016). Vizassist: an interactive user assistant for visual data mining. *The Visual Computer*, 32(11):1447–1463. Springer. doi : 10.1007/s00371-015-1132-9.

[Boyle et al., 1993] Boyle, J., Eick, S. G., Hemmje, M., Keim, D. A., Lee, J. P., and Sumner, E. (1993). Database issues for data visualization: Interaction, user interfaces, and presentation. In *Workshop on Database Issues for Data Visualization*, pages 25–34. Elsevier. doi: 10.1007_2FBFb0021143.

[Brehmer and Munzner, 2013] Brehmer, M. and Munzner, T. (2013). A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–2385. IEEE. doi: 10.1109/TVCG.2013.124.

[Chen, 2005] Chen, C. (2005). Top 10 unsolved information visualization problems. *IEEE computer graphics and applications*, 25(4):12–16. IEEE. doi: 10.1109/MCG.2005.91.

[Chen et al., 2008] Chen, M., Ebert, D., Hagen, H., Laramee, R. S., Van Liere, R., Ma, K.-L., Ribarsky, W., Scheuermann, G., and Silver, D. (2008). Data, information, and knowledge in visualization. *IEEE computer graphics and applications*, 29(1):12–19. IEEE. doi: 10.1109/MCG.2009.6.

[Chen et al., 2017] Chen, M., Grinstein, G., Johnson, C. R., Kennedy, J., and Tory, M. (2017). Pathways for theoretical advances in visualization. *IEEE computer graphics and applications*, 37(4):103–112. IEEE. doi: 10.1109/MCG.2017.3271463.

[Cooper et al., 2002] Cooper, R. J., Schriger, D. L., and Close, R. J. (2002). Graphical literacy: the quality of graphs in a large-circulation journal. *Annals of emergency medicine*, 40(3):317–322. doi: 10.1067/mem.2002.127327.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. In *Machine Learning*, volume 20, pages 273–297, USA. Kluwer Academic Publishers. doi: 10.1023/A:1022627411411.

[Dasgupta et al., 2017] Dasgupta, A., Lee, J.-Y., Wilson, R., Lafrance, R. A., Cramer, N., Cook, K., and Payne, S. (2017). Familiarity vs trust: A comparative study of domain scientists' trust in visual analytics and conventional analysis methods. *IEEE transactions on visualization and computer graphics*, 23(1):271–280. doi: 10.1109/TVCG.2016.2598544.

[Eler and Garcia, 2013] Eler, D. M. and Garcia, R. E. (2013). Using otsu's threshold selection method for eliminating terms in vector space model computation. In *2013 17th International Conference on Information Visualisation*, pages 220–226. IEEE. doi: 10.1109/IV.2013.29.

[Emerson and Stoto, 1983] Emerson, J. D. and Stoto, M. A. (1983). John Wiley & sons. isbn: 978-0-471-38491-5.

[Federico et al., 2017] Federico, P., Wagner, M., Rind, A., Amor-Amorós, A., Miksch, S., and Aigner, W. (2017). The role of explicit knowledge: A conceptual model of knowledge-assisted visual analytics. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 92–103. IEEE. doi: 10.1109/VAST.2017.8585498.

[Ferreira de Oliveira and Levkowitz, 2003] Ferreira de Oliveira, M. C. and Levkowitz, H. (2003). From visual data exploration to visual data mining: a survey. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):378–394. IEEE. doi: 10.1109/TVCG.2003.1207445.

[Fischer et al., 2010] Fischer, M., Bossdorf, O., Gockel, S., Hänsel, F., Hemp, A., Hessenmöller, D., Korte, G., Nieschulze, J., Pfeiffer, S., Prati, D., et al. (2010). Implementing large-scale and long-term functional biodiversity research: The biodiversity exploratories. *Basic and Applied Ecology*, 11(6):473–485. Elsevier. doi: 10.1016/j.baae.2010.07.009.

[Flemons et al., 2007] Flemons, P., Guralnick, R., Krieger, J., Ranipeta, A., and Neufeld, D. (2007). A web-based gis tool for exploring the world's biodiversity: The global biodiversity information facility mapping and analysis portal application (gbif-mapa). *Ecological Informatics*, 2(1):49 – 60. doi: 10.1016/j.ecoinf.2007.03.004.

[Fowler, 1996] Fowler, D. (1996). The binomial coefficient function. *The American mathematical monthly*, 103(1):1–17. Taylor & Francis. doi: 10.2307/2975209.

[Gao et al., 2015] Gao, T., Dontcheva, M., Adar, E., Liu, Z., and Karahalios, K. G. (2015). Datatone: Managing ambiguity in natural language interfaces for data visualization. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software Technology*, page 489–500. ACM. doi: 10.1145/2807442.2807478.

[Georges et al., 2020] Georges, N., Mhiri, I., Rekik, I., Initiative, A. D. N., et al. (2020). Identifying the best data-driven feature selection method for boosting reproducibility in classification tasks. *Pattern Recognition*, 101:107183. Elsevier. doi: 10.1016/j.patcog.2019.107183.

[Gilson et al., 2008] Gilson, O., Silva, N., Grant, P. W., and Chen, M. (2008). From web data to visualization via ontology mapping. In *Computer Graphics Forum*, volume 27, pages 959–966. Wiley Online Library. doi: 10.1111/j.1467-8659.2008.01230.x.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. Available at: http://www.deeplearningbook.org.

[Gotz and Wen, 2009] Gotz, D. and Wen, Z. (2009). Behavior-driven visualization recommendation. In *Proceedings of the 14th international conference on Intelligent user interfaces*, IUI '09, pages 315–324. doi: 10.1145/1502650.1502695.

[Groves and Gini, 2013] Groves, W. and Gini, M. (2013). Optimal airline ticket purchasing using automated user-guided feature selection. In *Twenty-Third International Joint Conference on Artificial Intelligence*. Available at: https://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/7000.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182. JMLR.org. doi: 10.5555/944919.944968.

[Guyon et al., 2008] Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. (2008). *Feature extraction: foundations and applications*, volume 207. Springer. Inderscience Publishers Ltd. doi: 10.1504/IJMEI.2014.060245.

[Haber and McNabb, 1990] Haber, R. and McNabb, D. (1990). Visualization idioms: A conceptual model for visualization systems. *Visualization in Scientific Computing*, pages 74–93. IEEE, Available at: www.academia.edu/2205276/Visualization_idioms_A_conceptual_model_for_scientific_visualization_systems.

[Hanrahan, 2006] Hanrahan, P. (2006). Vizql: a language for query, analysis and visualization. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, SIGMOD '06, pages 721–721. ACM. doi : 10.1145/1142473.1142560.

[Harris, 2000] Harris, R. L. (2000). *Information graphics: A comprehensive illustrated reference*. Oxford University Press. isbn: 0195135326.

[Heath and Ramakrishnan, 2002] Heath, L. S. and Ramakrishnan, N. (2002). The emerging landscape of bioinformatics software systems. *Computer*, 35(7):41–45. doi: 10.1109/MC.2002.1016900.

[Heer and Agrawala, 2006] Heer, J. and Agrawala, M. (2006). Software design patterns for information visualization. *IEEE transactions on visualization and computer graphics*, 12(5):853–860. IEEE. doi: 10.1109/TVCG.2006.178.

[Hinrichs et al., 2017] Hinrichs, U., El-Assady, M., Bradely, A. J., Forlini, S., and Collins, C. (2017). Risk the drift! stretching disciplinary boundaries through critical collaborations between the humanities and visualization. Available at: https://scibib.dbvis.de/publications/view/738.

[Ho, 1995] Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE. doi: 10.1109/ICDAR.1995.598994.

[Hu et al., 2019] Hu, K., Bakker, M. A., Li, S., Kraska, T., and Hidalgo, C. (2019). Vizml: A machine learning approach to visualization recommendation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12. ACM. doi: 10.1145/3290605.3300358.

[Islam et al., 2018] Islam, M. S., Hasan, M. M., Wang, X., Germack, H. D., et al. (2018). A systematic review on healthcare analytics: application and theoretical perspective of data mining. In *Healthcare*, volume 6, page 54. MDPI. doi: 10.3390/healthcare6020054.

[Jänicke et al., 2020] Jänicke, S., Kaur, P., Kuzmicki, P., and Schmidt, J. (2020). Participatory visualization design as an approach to minimize the gap between research and application. *Gap bet. Vis. Res. Vis. Soft.(VisGap). The Eurographics Association.* The Eurographics Association. doi: 10.1145/2207676.2208570.

[Janicki et al., 2016] Janicki, J., Narula, N., Ziegler, M., Guénard, B., and Economo, E. P. (2016). Visualizing and interacting with large-volume biodiversity data using client–server web-mapping applications: The design and implementation of antmaps. org. *Ecological Informatics*, 32:185–193. Elsevier. doi : 10.1016/j.ecoinf.2016.02.006.

[Jannach et al., 2010] Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender systems: an introduction.* Cambridge University Press. doi: 10.1017/CBO9780511763113, isbn: 9780511763113.

[Joachims, 1998] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer Berlin Heidelberg. doi: 10.1007/BFb0026683.

[Jobin et al., 2019] Jobin, K., Mondal, A., and Jawahar, C. (2019). Docfigure: A dataset for scientific document figure classification. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 1. doi: 10.1109/ICDARW.2019.00018.

[Johnson, 2007] Johnson, J. (2007). *GUI bloopers 2.0: common user interface design don’ts and dos.* Elsevier. doi: 10.1016/B978-0-12-370643-0.X5001-X.

[Jung et al., 2017] Jung, D., Kim, W., Song, H., Hwang, J.-i., Lee, B., Kim, B., and Seo, J. (2017). Chartsense: Interactive data extraction from chart images. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI ’17, pages 6706–6717. ACM. doi: 10.1145/3025453.3025957.

[Kaur et al., 2016] Kaur, P., Gaikwad, J., and König-Ries, B. (2016). Towards recommending visualizations for biodiversity data. *Biodiversity and conservation*, 25(9):1801–1803. Springer. doi: 10.1007/s10531-016-1157-z.

[Kaur et al., 2018] Kaur, P., Klan, F., and König-Ries, B. (2018). Issues and Suggestions for the Development of a Biodiversity Data Visualization Support Tool. In Johansson, J., Sadlo, F., and Schreck, T., editors, *EuroVis 2018 - Short Papers*, pages 73–77. The Eurographics Association. doi: 10.2312/eurovisshort.20181081.

[Kerpedjiev et al., 1997] Kerpedjiev, S., Carenini, G., Roth, S. F., and Moore, J. D. (1997). Autobrief: a multimedia presentation system for assisting data analysis. *Computer Standards & Interfaces*, 18(6-7):583–593. Elsevier. doi: 10.1016/S0920-5489(97)00022-6.

[Kerracher and Kennedy, 2017] Kerracher, N. and Kennedy, J. (2017). Constructing and evaluating visualisation task classifications: Process and considerations. *Computer Graphics Forum*, 36(3):47–59. doi: 10.1111/cgf.13167.

[Key et al., 2012] Key, A., Howe, B., Perry, D., and Aragon, C. (2012). Vizdeck: self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 681–684. ACM. doi: 10.1145/2213836.2213931.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. Available at: http://arxiv.org/abs/1412.6980.

[Klein and Staab, 2017] Klein, A. M. and Staab, M. (2017). CSPs: Trap nest data CSPs (solitary cavity-nesting hymenoptera). Available at: https://china.befdata.biow.uni-leipzig.de/datasets/487.

[Klumpar et al., 1994] Klumpar, D., Anderson, K., and Simoudis, A. (1994). Rave: Rapid visualization environment. Available at: https://ntrs.nasa.gov/citations/19940030540.

[Kohavi et al., 1997] Kohavi, R., John, G. H., et al. (1997). Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324. Elsevier. doi: 10.1007/978-3-540-35488-8.

[Kühn et al., 2016] Kühn, P., Scholten, T., and Seitz, S. (2016). Main experiment: Soil CNS and pH analyses of depth increments on site B (2010-2011). Available at: https://china.befdata.biow.uni-leipzig.de/datasets/332.

[Kulyk et al., 2007] Kulyk, O., Kosara, R., Urquiza, J., and Wassink, I. (2007). *Human-Centered Aspects*. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-71949-6_2.

[Landis and Koch, 1977] Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33:165. JSTOR. doi: 10.2307/2529310.

[Lee et al., 2004] Lee, B., Parr, C. S., Campbell, D., and Bederson, B. B. (2004). How users interact with biodiversity information using taxontree. In *Proceedings of the working conference on Advanced visual interfaces*, AVI '04, pages 320–327. ACM. doi: 10.1145/989863.989918.

[Lee et al., 2006] Lee, B., Plaisant, C., Parr, C. S., Fekete, J.-D., and Henry, N. (2006). Task taxonomy for graph visualization. In *Proceedings of the 2006 AVI Workshop on Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, BELIV '06, pages 1–5. ACM. doi: 10.1145/1168149.1168168.

[Levenshtein, 1966] Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Available at: https://nymity.ch/sybilhunting/pdf/Levenshtein1966a.pdf.

[Li et al., 2018] Li, G., Zhang, S., Liang, J., Cao, Z., and Guo, C. (2018). Augmenting embedding with domain knowledge for oral disease diagnosis prediction. In *International Conference on Smart Computing and Communication*, pages 236–250. Springer. doi: 10.1007/978-3-030-05755-8_24.

[Lindsay et al., 2012] Lindsay, S., Jackson, D., Schofield, G., and Olivier, P. (2012). Engaging older people using participatory design. In *Proceedings of the SIGCHI conference on human factors in computing systems*, CHI '12, pages 1199–1208. ACM. doi: 10.1145/2207676.2208570.

[Löffler et al., 2020] Löffler, F., Abdelmageed, N., Babalou, S., Kaur, P., and König-Ries, B. (2020). Tag me if you can! semantic annotation of biodiversity metadata with the qemp corpus and the biodivtagger. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4557–4564. European Language Resources Association. isbn : 979-10-95546-34-4.

[Lotz et al., 2012] Lotz, T., Nieschulze, J., Bendix, J., Dobbermann, M., and König-Ries, B. (2012). Diverse or uniform?—intercomparison of two major german project databases for interdisciplinary collaborative functional biodiversity research. *Ecological informatics*, 8:10–19. Elsevier. doi: 10.1016/j.ecoinf.2011.11.004.

[Luo et al., 2018] Luo, Y., Qin, X., Tang, N., and Li, G. (2018). Deepeye: Towards automatic data visualization. In *2018 IEEE 34th International Conference on Data Engineering (ICDE)*, pages 101–112. IEEE. doi: 10.1109/ICDE.2018.00019.

[Macedo et al., 2019] Macedo, F., Oliveira, M. R., Pacheco, A., and Valadas, R. (2019). Theoretical foundations of forward feature selection methods based on mutual information. *Neurocomputing*, 325:67–89. Elsevier. doi: 10.1016/j.neucom.2018.09.077.

[Mackinlay, 1986] Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. *Acm Transactions On Graphics (Tog)*, 5(2):110–141. ACM. doi : 10.1145/22949.22950.

[Mackinlay et al., 2007] Mackinlay, J., Hanrahan, P., and Stolte, C. (2007). Show me: Automatic presentation for visual analysis. *IEEE transactions on visualization and computer graphics*, 13(6):1137–1144. IEEE. doi : 10.1109/TVCG.2007.70594.

[Marangunić and Granić, 2015] Marangunić, N. and Granić, A. (2015). Technology acceptance model: a literature review from 1986 to 2013. *Universal access in the information society*, 14(1):81–95. Springer. doi: 10.1007/s10209-014-0348-1.

[McHugh, 2012] McHugh, M. L. (2012). Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282. Biochemia medica. doi: 10.11613/BM.2012.031.

[McKenna et al., 2014] McKenna, S., Mazur, D., Agutter, J., and Meyer, M. (2014). Design activity framework for visualization design. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2191–2200. IEEE. doi: 10.1109/TVCG.2014.2346331.

[Mennis et al., 2000] Mennis, J. L., Peuquet, D. J., and Qian, L. (2000). A conceptual framework for incorporating cognitive principles into geographical database representation. *International Journal of Geographical Information Science*, 14(6):501–520. Taylor & Francis. doi: 10.1080/136588100415710.

[Moody and Jones, 2000] Moody, A. and Jones, J. A. (2000). Soil response to canopy position and feral pig disturbance beneath quercus agrifolia on santa cruz island, california. *Applied Soil Ecology*, 14(3):269 – 281. Elsevier. doi:10.1016/S0929-1393(00)00053-6.

[Munzner, 2009] Munzner, T. (2009). A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928. IEEE. doi: 10.1109/TVCG.2009.111.

[Munzner, 2014] Munzner, T. (2014). *Visualization analysis and design.* AK Peters/CRC press. isbn: 9781498759717.

[Mutlu et al., 2016] Mutlu, B., Veas, E., and Trattner, C. (2016). Vizrec: Recommending personalized visualizations. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 6(4):1–39. ACM. doi: 10.1145/2983923.

[Noll et al., 2016] Noll, L., Leonhardt, S., Arnstadt, T., Hoppe, B., Poll, C., Matzner, E., Hofrichter, M., and Kellner, H. (2016). Fungal biomass and extracellular enzyme activities in coarse woody debris of 13 tree species in the early phase of decomposition. *Forest Ecology and Management*, 378:181–192. Elsevier. doi: 10.1016/j.foreco.2016.07.035.

[Nusrat and Kobourov, 2015] Nusrat, S. and Kobourov, S. (2015). Visualizing cartograms: Goals and task taxonomy. *arXiv*, arXiv:1502.07792. Available at: https://ui.adsabs.harvard.edu/abs/2015arXiv150207792N.

[Parameswaran et al., 2013] Parameswaran, A., Polyzotis, N., and Garcia-Molina, H. (2013). Seedb: Visualizing database queries efficiently. *Proc. VLDB Endow.*, 7(4):325–328. VLDB Endowment. doi: 10.14778/2732240.2732250.

[Piantadosi, 2014] Piantadosi, S. T. (2014). Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic bulletin & review*, 21(5):1112–1130. Springer. doi: 10.3758/s13423-014-0585-6.

[Pike et al., 2009] Pike, W. A., Stasko, J., Chang, R., and O'connell, T. A. (2009). The science of interaction. *Information visualization*, 8(4):263–274. Palgrave Macmillan. doi: 10.1057/ivs.2009.22.

[Pu et al., 2012] Pu, P., Chen, L., and Hu, R. (2012). Evaluating recommender systems from the user's perspective: survey of the state of the art. *User Modeling and User-Adapted Interaction*, 22(4-5):317–355. Springer. doi: 10.1007/s11257-011-9115-7.

[Ramos et al., 2003] Ramos, J. et al. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 133–142. Piscataway, NJ. doi: 10.1.1.121.1424.

[Roth, 1994] Roth, P. L. (1994). Missing data: A conceptual review for applied psychologists. *Personnel psychology*, 47(3):537–560. Wiley Online Library. doi: 10.1111/j.1744-6570.1994.tb01736.x.

[Roth and Mattis, 1990] Roth, S. F. and Mattis, J. (1990). Data characterization for intelligent graphics presentation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 193–200. ACM. doi: 10.1145/97243.97273.

[Rougier et al., 2014] Rougier, N. P., Droettboom, M., and Bourne, P. E. (2014). Ten simple rules for better figures. *PLoS Comput Biol*, 10(9):e1003833. PLOS. doi: 10.1145/2807442.2807478.

[Rowley, 2007] Rowley, J. (2007). The wisdom hierarchy: representations of the dikw hierarchy. *Journal of information science*, 33(2):163–180. SAGE. doi: 10.1177/0165551506070706.

[Rui et al., 1998] Rui, Y., Huang, T. S., Ortega, M., and Mehrotra, S. (1998). Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on circuits and systems for video technology*, 8(5):644–655. doi: 10.1109/76.718510.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252. Springer. doi: 10.1007/s11263-015-0816-y.

[Sandler et al., 2018] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. (2018). Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *arXiv preprint arXiv:1801.04381*. Available at: https://www.arxiv-vanity.com/papers/1801.04381/.

[Saraiya et al., 2004] Saraiya, P., North, C., and Duca, K. (2004). An evaluation of microarray visualization tools for biological insight. In *IEEE Symposium on Information Visualization*, pages 1–8. IEEE. doi: 10.1109/INFVIS.2004.5.

[Saraiya et al., 2005] Saraiya, P., North, C., and Duca, K. (2005). An insight-based methodology for evaluating bioinformatics visualizations. *IEEE transactions on visualization and computer graphics*, 11(4):443–456. IEEE. doi: 10.1109/TVCG.2005.53.

[Satyanarayan et al., 2016] Satyanarayan, A., Moritz, D., Wongsuphasawat, K., and Heer, J. (2016). Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics*, 23(1):341–350. IEEE. doi: 10.1109/TVCG.2016.2599030.

[Savva et al., 2011] Savva, M., Kong, N., Chhajta, A., Fei-Fei, L., Agrawala, M., and Heer, J. (2011). Revision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST '11, pages 393–402. ACM. doi: 10.1145/2047196.2047247.

[Schall and Ammer, 2017] Schall, P. and Ammer, C. (2017). Forest ep - stand structural attributes – core ssa. v1.2.2. Available at: https://www.bexis.uni-jena.de/PublicData/PublicData.aspx?DatasetId=20106.

[Schriger et al., 2006] Schriger, D. L., Sinha, R., Schroter, S., Liu, P. Y., and Altman, D. G. (2006). From submission to publication: a retrospective review of the tables and figures in a cohort of randomized controlled trials submitted to the british medical journal. *Annals of emergency medicine*, 48(6):750–756. Elsevier. doi: 10.1016/j.annemergmed.2006.06.017.

[Schulz et al., 2013] Schulz, H.-J., Nocke, T., Heitzler, M., and Schumann, H. (2013). A design space of visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2366–2375. IEEE. doi: 10.1109/TVCG.2013.120.

[Schumann and Müller, 2013] Schumann, H. and Müller, W. (2013). *Visualisierung: Grundlagen und allgemeine methoden*. Springer-Verlag. doi: 10.1007/978-3-642-57193-0.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47. ACM. doi: 10.1145/505282.505283.

[Seitz, 2017] Seitz, S. (2017). Main experiment: Cover of biological soil crusts in runoff plots: bryophyte species. Available at: https://china.befdata.biow.uni-leipzig.de/datasets/510.

[Seitz et al., 2016] Seitz, S., Goebes, P., Song, Z., Bruelheide, H., Härdtle, W., Kühn, P., Li, Y., and Scholten, T. (2016). Tree species and functional traits but not species richness affect interrill erosion processes in young subtropical forests. *Soil*, 2(1):49. Copernicus GmbH. doi: 10.5194/soil-2-49-2016.

[Setlur et al., 2016] Setlur, V., Battersby, S. E., Tory, M., Gossweiler, R., and Chang, A. X. (2016). Eviza: A natural language interface for visual analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 365–377. ACM. doi: 10.1145/2984511.2984588.

[Shneiderman, 1996] Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE symposium on visual languages*, pages 336–343. IEEE. doi : 10.1109/VL.1996.545307.

[Shneiderman, 1999] Shneiderman, B. (1999). Dynamic queries, starfield displays, and the path to spotfire. Available at. http://www.cs.umd.edu/hcil/spotfire/.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. Available at: http://arxiv.org/abs/1409.1556.

[Spence, 2001] Spence, R. (2001). *Information visualization*, volume 1. Springer. isbn: 978-3-319-07340-8, doi: 10.1007/978-3-319-07341-5.

[Staab et al., 2016] Staab, M., Klein, A. M., and Peters, J. (2016). Main experiment: Visitors of extrafloral nectaries (2012). Available at: https://china.befdata.biow.uni-leipzig.de/datasets/498.

[Staab et al., 2019] Staab, M., Klein, A. M., and Peters, J. (2019). CSPs: Ant data CSPs complete. Available at: https://china.befdata.biow.uni-leipzig.de/datasets/502.

[Steichen et al., 2013] Steichen, B., Carenini, G., and Conati, C. (2013). User-adaptive information visualization: using eye gaze data to infer visualization tasks and user cognitive abilities. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, IUI '13, pages 317–328. ACM. doi: 10.1145/2449396.2449439.

[Stolte et al., 2002] Stolte, C., Tang, D., and Hanrahan, P. (2002). Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65. IEEE. doi : 10.1109/2945.981851.

[Thrash et al., 2019] Thrash, A., Arick, M., Barbato, R. A., Jones, R. M., Douglas, T. A., Esdale, J., Perkins, E. J., and Garcia-Reyero, N. (2019). Keanu: a novel visualization tool to explore biodiversity in metagenomes. *BMC bioinformatics*, 20(2):141–149. Springer Nature. doi: 10.1186/s12859-019-2629-4.

[Tory and Moller, 2004] Tory, M. and Moller, T. (2004). Human factors in visualization research. *IEEE Transactions on Visualization and Computer Graphics*, 10(1):72–84. IEEE. doi: 10.1109/TVCG.2004.1260759.

[Valdez et al., 2016] Valdez, A. C., Ziefle, M., Verbert, K., Felfernig, A., and Holzinger, A. (2016). Recommender systems for health informatics: state-of-the-art and future perspectives. In *Machine Learning for Health Informatics*, pages 391–414. Springer. doi: 10.1007/978-3-319-50478-0_20.

[Vartak et al., 2017] Vartak, M., Huang, S., Siddiqui, T., Madden, S., and Parameswaran, A. (2017). Towards visualization recommendation systems. *ACM SIGMOD Record*, 45(4):34–39. ACM. doi: 10.1145/3092931.3092937.

[Vartak et al., 2014] Vartak, M., Madden, S., Parameswaran, A., and Polyzotis, N. (2014). Seedb: Automatically generating query visualizations. *Proc. VLDB Endow.*, 7(13):1581–1584. VLDB Endowment. doi: 10.14778/2733004.2733035.

[Venkatesh and Anuradha, 2019] Venkatesh, B. and Anuradha, J. (2019). A review of feature selection and its methods. *Cybernetics and Information Technologies*, 19(1):3–26. Sciendo. doi: 10.2478/cait-2019-0001.

[Vergne, 2004] Vergne, J. (2004). Découverte locale des mots vides dans des corpus bruts de langues inconnues, sans aucune ressource. *Journées internationales d'Analyse statistique des données textuelles*, 7(8). Available at: https://lexicometrica.univ-paris3.fr/jadt/jadt2004/pdf/JADT_114.pdf.

[Viegas et al., 2007] Viegas, F. B., Wattenberg, M., Van Ham, F., Kriss, J., and McKeon, M. (2007). Manyeyes: a site for visualization at internet scale. *IEEE transactions on visualization and computer graphics*, 13(6):1121–1128. IEEE. doi: 10.1109/TVCG.2007.70577.

[Voigt et al., 2013] Voigt, M., Pietschmann, S., and Meißner, K. (2013). A semantics-based, end-user-centered information visualization process for semantic web data. In *Semantic Models for Adaptive Interactive Systems.* Springer. doi: 10.1007/978-1-4471-5301-6_5.

[Wagner, 2015] Wagner, M. (2015). Integrating explicit knowledge in the visual analytics process. *Doctoral Consortium on Computer Vision, Imaging and Computer Graphics Theory and Applications (DCVISIGRAPP 2015).* SCITEPRESS. doi: 10.13140/RG.2.2.19279.69283.

[Wagner et al., 2017] Wagner, M., Rind, A., Thür, N., and Aigner, W. (2017). A knowledge-assisted visual malware analysis system: Design, validation, and reflection of kamas. *Computers & Security*, 67:1–15. Elsevier. doi: 10.1016/j.cose.2017.02.003.

[Wagner et al., 2018] Wagner, M., Slijepcevic, D., Horsak, B., Rind, A., Zeppelzauer, M., and Aigner, W. (2018). Kavagait: Knowledge-assisted visual analytics for clinical gait analysis. *IEEE transactions on visualization and computer graphics*, 25(3):1528–1542. IEEE. doi: 10.1109/TVCG.2017.2785271.

[Ware, 2012] Ware, C., editor (2012). *Information Visualization: Perception for Design.* Morgan Kaufmann, San Francisco. CA, USA. isbn: 9780123814647.

[Webber et al., 2010] Webber, W., Moffat, A., and Zobel, J. (2010). A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38. ACM, doi: 10.1145/1852102.1852106.

[Wehrend and Lewis, 1990] Wehrend, S. and Lewis, C. (1990). A problem-oriented classification of visualization techniques. In *Proceedings of the First IEEE Conference on Visualization: Visualization90*, pages 139–143. IEEE. doi: 10.1109/VISUAL.1990.146375.

[Wijk, 2006] Wijk, J. J. V. (2006). Bridging the gaps. *IEEE Computer Graphics and Applications*, 26(6):6–9. IEEE. doi: 10.1109/MCG.2006.120.

[Winkler, 1999] Winkler, W. E. (1999). The state of record linkage and current research problems. In *Statistical Research Division, US Census Bureau*. Citeseer. Available at: `https://courses.cs.washington.edu/courses/cse590q/04au/papers/Winkler99.pdf`.

[Wongsuphasawat et al., 2015] Wongsuphasawat, K., Moritz, D., Anand, A., Mackinlay, J., Howe, B., and Heer, J. (2015). Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics*, 22(1):649–658. IEEE. doi: `10.1109/TVCG.2015.2467191`.

[Xu, 2019] Xu, L. (2019). User story based information visualization type recommendation system. *International Journal of Information Engineering & Electronic Business*, 11(3). MECS. doi: `10.5815/ijieeb.2019.03.01`.

[Zhao et al., 2020] Zhao, J., Fan, M., and Feng, M. (2020). Chartseer: Interactive steering exploratory visual analysis with machine intelligence. *IEEE Transactions on Visualization and Computer Graphics*. IEEE. doi: `10.1109/TVCG.2020.3018724`.

[Zhou and Feiner, 1998] Zhou, M. X. and Feiner, S. K. (1998). Visual task characterization for automated visual discourse synthesis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 392–399. ACM. doi: `10.1145/274644.274698`.

Appendices

Appendix A

Visualization requirement analysis survey

seit 1558

VISUALIZATION SURVEY

This survey is a part of my PhD work, which is focused on building a framework for recommending visualizations for biodiversity/ecology data, under the affiliation of Institute of Computer Science, Friedrich-Schiller-University Jena. It is meant to get visualization usage information from ecologists and biodiversity researchers. This data will help us to understand the current practices and patterns from the domain experts. All information will be held secure and will solely be used for the above mentioned research.

1. Name and Email address (optional)

____

2. What is your main research theme?

____

____

3. What visualization software do you use normally and for what purposes?

- **Visualization software**: □ Excel □ R □ SPSS □ Cytospace □ Tableau □ ArcGIS
  □ Others (Please specify) ____
- **Purposes**: □ Data Search □ Data Exploration □ Quality Assurance □ Data Analysis
  □ Result presentation in publications □ Others (Please specify) ______________________________
  ____

4. What factors do you consider when deciding if a particular visualization is good for your task? Also, please choose how prominent/important is that in the visualization selection process?

Factors	Most Prominent	Prominent	Less Prominent
(a) Data Type	□	□	□
(b) Data Variables	□	□	□
(c) Data Size	□	□	□
(d) It looks good (like colour or graphical icons)	□	□	□
(e) I only know these ones or literature shows the same	□	□	□
(f) Easy to use	□	□	□

5. Do you face difficulties in selecting a visualization for representing your research data (like in publications or in presentations)?

(Select one option)

○ Yes ○ Other (please specify) ______________________________

○ No

6. Would you consider to have a software tool that can guide you in the selection of suitable visualization for your data?

(Select one option)

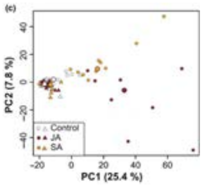
○ Yes ○ Other (please specify) ______________________________

○ No

From this point onwards, you will be shown some visualizations. For each visualization, you will be asked to mention some generic studies or analysis that you perform through that visualization. Then rate each on the preference of suitability of that visualization for that analysis (in the scale of 1 to 3).

Question 8 is done as an example for better understanding.

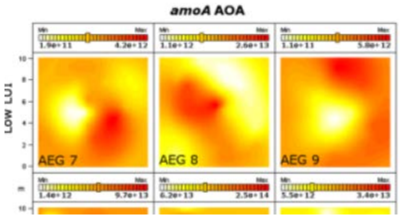
8. **SCATTERPLOT**



	Task	Preference	If you don't use this visualization, reason?
a.	Spatial Distribution	3 (Least Preferred)	□ Never needed □ Don't know about it
b.	PCA	1 (Most Preferred)	
c.	RDA	2 (Neutral)	

For example: You use Scatterplot to represent Spatial Distribution, for doing PCA and RDA. Your preference for using Scatterplot for PCA is more, in comparison to RDA and Spatial Distribution. Third column is not filled, as you use this visualization.

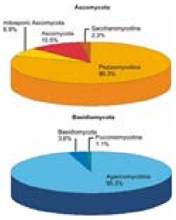
9. **HEATMAP**



	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

Any comments about the visualization:

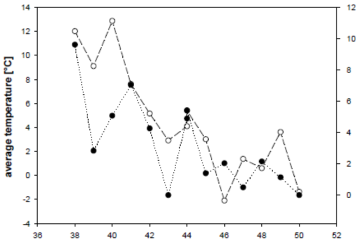
10. **PIE CHART**



	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

Any comments about the visualization:

11. **LINE GRAPH**



	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

seit 1558

VISUALIZATION SURVEY

This survey is a part of my PhD work, which is focused on building a framework for recommending visualizations for biodiversity/ecology data, under the affiliation of Institute of Computer Science, Friedrich-Schiller-University Jena. It is meant to get visualization usage information from ecologists and biodiversity researchers. This data will help us to understand the current practices and patterns from the domain experts. All information will be held secure and will solely be used for the above mentioned research.

1. Name and Email address (optional)

____

2. What is your main research theme?

____

____

3. What visualization software do you use normally and for what purposes?

- **Visualization software**: □ Excel □ R □ SPSS □ Cytospace □ Tableau □ ArcGIS

  □ Others (Please specify) ____

- **Purposes**: □ Data Search □ Data Exploration □ Quality Assurance □ Data Analysis

  □ Result presentation in publications □ Others (Please specify) ____

  ____

4. What factors do you consider when deciding if a particular visualization is good for your task? Also, please choose how prominent/important is that in the visualization selection process?

Factors	Most Prominent	Prominent	Less Prominent
(a) Data Type	□	□	□
(b) Data Variables	□	□	□
(c) Data Size	□	□	□
(d) It looks good (like colour or graphical icons)	□	□	□
(e) I only know these ones or literature shows the same	□	□	□
(f) Easy to use	□	□	□

5. Do you face difficulties in selecting a visualization for representing your research data (like in publications or in presentations)?

(Select one option)

○ Yes ○ Other (please specify) _______________________________

○ No

6. Would you consider to have a software tool that can guide you in the selection of suitable visualization for your data?

(Select one option)

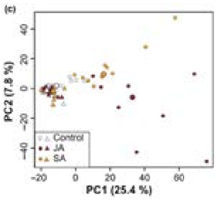
○ Yes ○ Other (please specify) _______________________________

○ No

From this point onwards, you will be shown some visualizations. For each visualization, you will be asked to mention some generic studies or analysis that you perform through that visualization. Then rate each on the preference of suitability of that visualization for that analysis (in the scale of 1 to 3).

Question 8 is done as an example for better understanding. Its explanation is also available in italics after the table

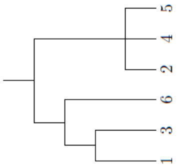
8. **SCATTERPLOT**



	Task	Preference	If you don't use this visualization, reason?
a.	Spatial Distribution	3 (Least Preferred)	□ Never needed □ Don't know about it
b.	PCA	1 (Most Preferred)	
c.	RDA	2 (Neutral)	

For example: You use Scatterplot to represent Spatial Distribution, for doing PCA and RDA. Your preference for using Scatterplot for PCA is more, in comparison to RDA and Spatial Distribution. Third column is not filled, as you use this visualization.

9. **TREE**



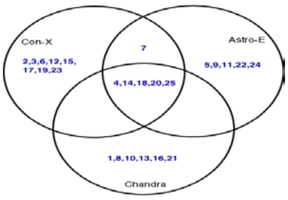
	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

Any comments about the visualization:

____

____

10. **VENN DIAGRAM**



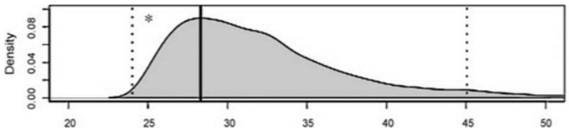
	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

Any comments about the visualization:

____

____

11. **DENSITY PLOT**



	Task	Preference	If you don't use this visualization, reason?
a.			□ Never needed □ Don't know about it
b.			
c.			

Appendix B

Raw data for Table 2.1

Scatterplot	**Dendrogra**	**Density Pl**	**Boxplot**	**Scatterplo**	**CoPlot**	**Heatmap**
PCA	Phylogene	Population	Carbon Stock	Covariatio	Regression	3D variable analysis
PCA	Cluster An	Spatial Dis	Statistical Analysis	Collinearit	2 way inte	Intensity analysis
Regression	Collinneari	Data Distri	Group Comparison	Regression	Population	3D variable analysis
Spatial Distribu	Phylogene	Bayesian E	Factorial Design	Data Explo	Correlatio	Spatial Distribution
CA	Cluster An	Data Distri	Outlier Presentation	Data Distri	Data Inves	2 way interation
Phylogenetic A	Classificati	Vegetatior	Factorial Design	Quality Co	Data Overv	Spatial Distribution
Cluster Analysis	Cluster An	Overview	Fatorial Design	Data Explo	Multivaria	Regression
PCA	Vegetatior	Trend	Statistical Comparison	Covariatio	Compariso	Spatial Distribution
RDA	Phylogene	Data Distri	Data Overview	Regression	Compariso	3D variable analysis
GLM	Cluster An	Density Di	Categorical Data Distril	Relationsh	Data Explo	Regression
Spatial Distribu	Categorica	Data Distri	Categorical Data Distril	Data Explo	Associatio	Spatial Distribution
RDA	Cluster An	Trend	Numerical Pattern	Data Distribution		Abundance
PCA	Cluster An	Landscape	Factorial Design	Correlation		Spatial Mapping
Spatial Distribu	Communit	Normality	Comparison Distributic	Data Overview		Comparison
RDA	Phylogene	Spatial Dis	Data Exploration	Correlation		Statistical Analysis
PCA	Kinship An	Data Distri	Experimental Data	Data Distribution		Spatial Statistics
RDA	Phylogene	Data Distri	Group Comparison	Data Exploration		Data Distribution
Cluster Analysis	Phylogene	Data Explo	Species Abundance Pre	Correlation		Heterogeneity
Phylogenetic A	Phylogeog	Normality	Species Richness Prese	Correlation		Spatial Distribution
RDA	Cluster An	Data Distri	Group Comparison	Linear Modelling		Correlation
Spatial Distribu	Workflow	Density Di	Group Comparison	Quality Control		Trend
Spatial Distribu	Species Re	Data Explo	Data Distribution	Spatial Correlation		Data Exploration
PCA	Cluster An	Altitudinal	Outlier Presentation	Correlation		Data Distribution
RDA	Phylogene	Data Distri	Group Comparison	Independence Test		
PCA	Cluster An	Bayesian E	Group Comparison	Data Investigation		
PCA	Phylogene	Temporal I	Data Exploration	Correlation		
Spatial Distribu	Species Re	Data Distri	Factorial Design	Regression		
RDA	Cluster An	Posterior D	Summary Statistics	Data Overview		
RDA	Phylogene	Data Distri	Outlier Presentation	Correlation		
PCA	Distance E	Data Inspe	Data Overview	Covariation Exploration		
RDA	Cluster An	Root Grow	Group Comparison	Data Exploration		
Spatial Distribu	Twinspan	Data Distri	Group Comparison	Correlation		
Correlation	Classificati	Compariso	Data Exploration	Correlation		
Cluster Analysis	Relatednes	Statistical	Summary Statistics	Data Exploration		
PCA	Phylogene	Statistical	Data Description	Temporal Distribution		
Cluster Analysis	Trait Alloc	Data Distri	Variance Analysis	Correlation		
PCA	Diversity	Data Distri	Statistical Comparison	Multivariate Statistics		
Ecological Grou	Ward's Dis	Data Distri	Statistical Comparison	Overview		
Spatial Distribu	Phylogene	Compariso	Statistical Comparison	Trend		
PCA	Evolutionary Analysis		Data Exploratiom	Trend		
Spatial Distribu	Species Description		Data Dispersion	Data Distribution		
Spatial Analysis	Phylogenetic Analysis		Group Comparison	Data Exploration		
Multivariate Sta	Nestedness		Overflow	Trend		
Correlation	Similarity		Significance Test	Multivariate Statistics		
Distribution	Phylogenetic Analysis		Factorial Design	Correlation		
Distribution	Species Relation		Summary Statistics	Multivariate Statistics		
Correlation	Hierarchy		Summary Statistics	Regression		
PCA	Data Exploration		HSD Test	Correlation		
Correlation	Clustering		T test	Correlation		

Data Exploration	Wilcoxon Test	Comparison
Association	Data Exploration	Outlier Identification
CCA	Data Exploration	Temporal Distribution
DCA	Significance Test	Comparison
	Group Comparison	ANOVA
	Factorial Design	Correlation
	Data Distribution	Data Exploration
	Group Comparison	
	Biomass Distribution	
	Species Abundance Presentation	
	Group Comparison	
	Group Comparison	
	Group Comparison	
	Summary Statistics	
	Culturability values	
	Culturability values	
	Group Comparison	
	Group Comparison	
	Comparison	
	Data Distribution	
	Data Exploration	

LineChart	**Pie Chart**	**Bar Chart**
2D variable analysis	Proportion Presentation	Data Distribution
Temporal Analysis	1D variable analysis	2D variable analysis
Correlation	Proportion Presentation	Factorial Design
Temporal Analysis	Proportion Presentation	Data Exploration
1D variable analysis	Comparison	Comparison
Trend	Proportion Presentation	Group Comparison
Correlation	Composition	Proportion
Regression	Proportion Presentation	Factorial Design
Temporal Analysis	Composition	Relative Abundance
Temporal Analysis	Composition	Temporal Analysis
Temporal Analysis	Distribution	Factorial Design
Temporal Analysis	Proportion Presentation	Phylogenetic Distribution
Comparison	Simple Statistics	Relative Abundance
Correlation	Composition	ANOVA
Growth Curves	Proportion Presentation	Statistical Analysis
3D Analysis		Taxonomic Richness
Temporal Analysis		Trends
Temporal Analysis		Comparison
Trend		Comparison
Trend		Distribution
Comparison		Association
Association		
Correlation		

Appendix C

Different visualization types found in biodiversity publications

1. 100% stacked area chart
2. Alluvial diagram
3. Area chart
4. Bar Chart
5. Bar Chart with error bars
6. Beanplot
7. Bifurcation diagram
8. Bubble chart
9. Bubble map
10. Boxplot
11. Chord diagram
12. Choropleth map
13. Circular dendrogram
14. Column chart
15. Column chart with error bars
16. Contour map
17. Contour plot
18. Correlogram
19. Dendrogram
20. Density chart
21. Dot map
22. Dot plot
23. Error plot
24. Flow map
25. Grid heatmap
26. Heatmap
27. Histogram
28. Line chart
29. Map
30. Mosiac bar chart
31. Mosiac column chart
32. Mosiac plot
33. Multiset bar chart
34. Multiset bar chart with error bars
35. Multiset column chart
36. Multiset column chart with error bars
37. Multiset stacked column chart
38. Multiset stacked column chart with error bars
39. Node-link diagram
40. Notched boxplot
41. Ordination scatterplot
42. Pie chart
43. Polar area chart
44. Population pyramid
45. Scatterplot
46. Scatterplot matrix
47. Scatterplot with regression line
48. Span chart
49. Spectogram
50. Stacked area chart
51. Stacked bar chart
52. Stacked column chart
53. Stacked column chart with error bars
54. Streamgraph
55. Taylor diagram
56. Timeseries
57. Triangle diagram
58. Violin plot
59. Waterfall diagram

Appendix D

Retained classes

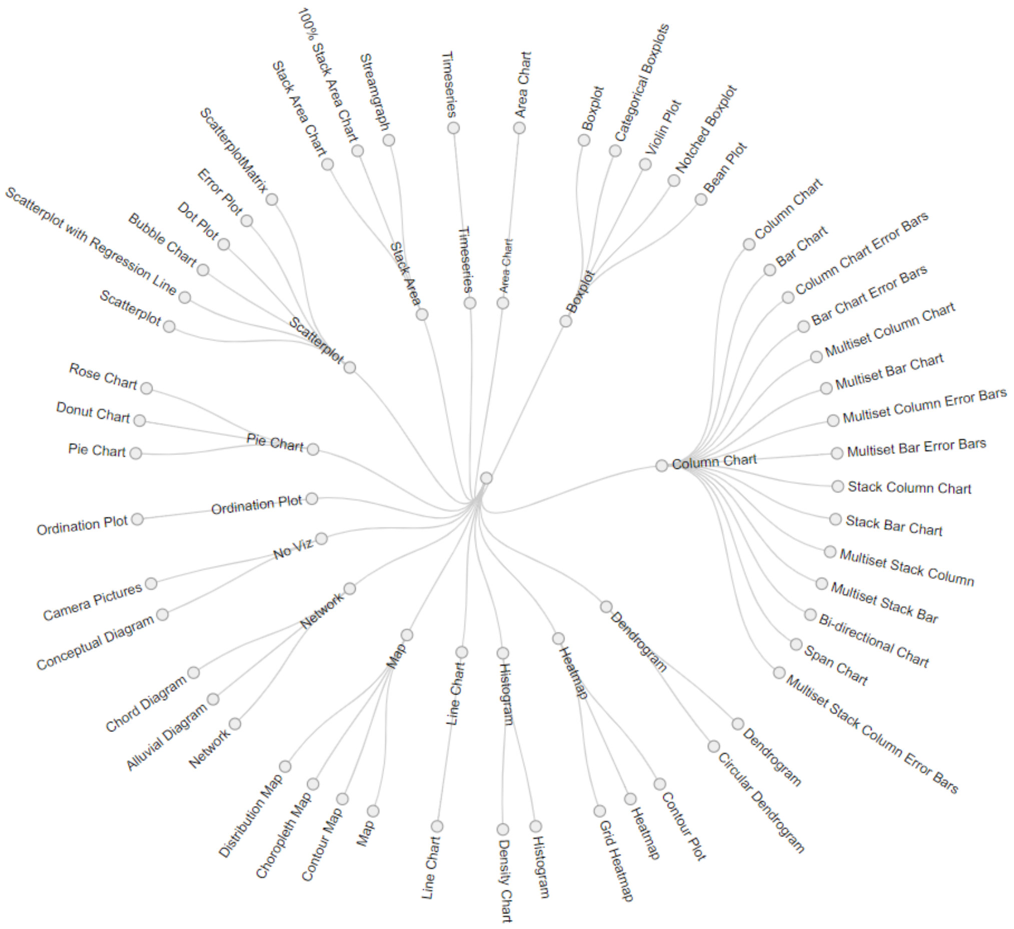
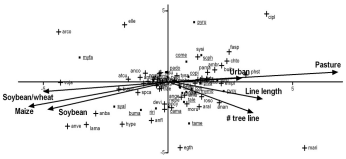
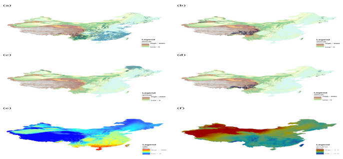
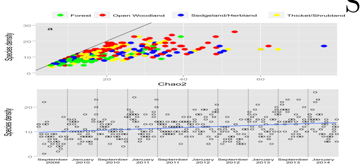
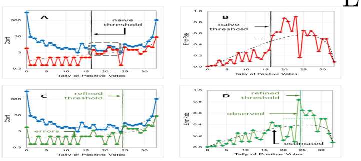
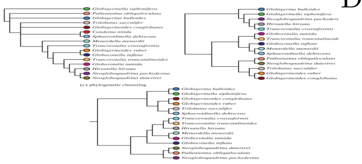
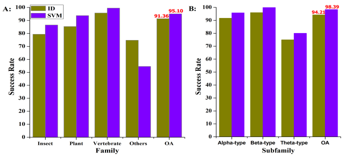
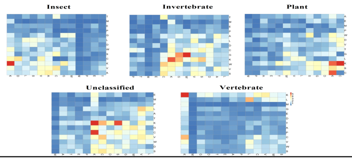
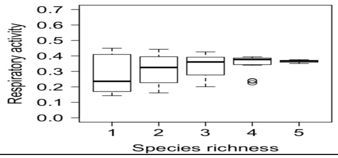
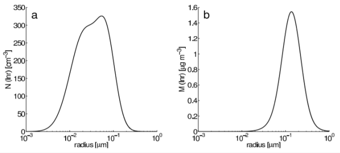
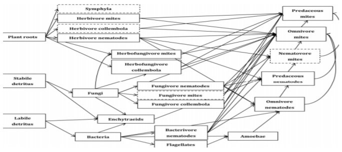
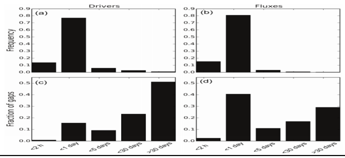
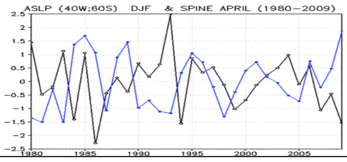
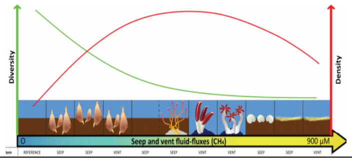
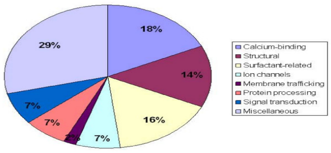
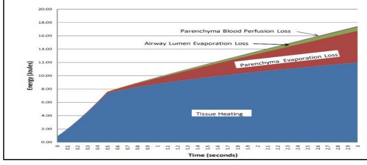


Figure 42: Grouping of visualization subclasses into superclasses

Appendix E

Example of visualization classes

Classes	Caption Classes
 Ordination Plot	Ordination plot for CCA at facet scale. Only variables with the highest interset correlations are drawn. Species are identified using four letter keys (see Additional file 1: Table S1). Species with underlined keys and square symbols are exotic to the Pampas. Ordination explains 2.8% of total variance.
 Map	Species distributions and environmental layers. Figure (a–d) are distribution maps of species pooled to genera Pinus, Abies, Larix and Picea, respectively, superimposed over background digital elevation model (DEM) maps with 50% transparence. Figure (e) and (f) are maps of the environmental factors gross degree days (GDD) and aridity respectively.
 Scatterplot	a) Scatterplots of observed species densities and Chao 2 estimates of species richness at replicates in the four habitat types along the transect and b) a scatterplots of species densities collected at the 11 sites as a function of the chronological order of surveys.
 Line Chart	Model logP2-2 uses three hidden neurons and 45 descriptors as input. The horizontal dotted lines running across the thresholds indicate where an error rate of 0.5 would fall. (A) Distribution of predictions (blue) and errors (red) for the external validation set. Dashed lines represent the fitted beta binomial distributions for the corresponding training pool results.
 Dendrogram	Ecological and phylogenetic clustering of the 15 macroperforate species that overlap between the Tohoku University dataset and the coretop/exemplar dataset. (a) Consensus cluster dendrogram of the full-3D Tohoku University specimen morphospace (same as figure 10 c). (b) Ecological cluster dendrogram built using Jaccard distances calculated from three ecological traits (table 1). (c) The phylogenetic relationships between the 15 macroperforate species, as pruned and redrawn from Aze et al. 's [22] stratophenetic phylogeny. Dendrogram tip label colours correspond to morphospace species colours from figure 9.
 Column Chart	Comparing the performance of the proposed method with our previous methods. A: indicated the prediction results of defensin family; B: indicated the prediction results of vertebrate defensin subfamily.
 Heatmap	The heatmap shows the adjacent correlation of 13 reduced amino acids for five different defensin families.

Boxplot  	Boxplots of the effects of species richness on respiratory activity under constant temperature conditions.
Area Chart  	Initial aerosol (a) number density and (b) mass density size distributions for all simulations.
Network  Symphyla Herbivore mites Herbivore collembola Herbivore nematodes Plant roots Herbofungivore mites Herbofungivore collembola Predaceous mites Omnivore mites Nematovore mites Predaceous nematodes Omnivore nematodes Stabile detritus Fungi Fungivore nematodes Fungivore mites Fungivore collembola Labile detritus Enchytraeids Bacteria Bacterivore nematodes Flagellates Amoebae	Soil food web diagram representative for all three land use types in the Koiliaris Critical Zone Observatory (Crete, GR). Boxes represent the presence of trophic groups in the soil food web, arrows represent feeding interactions based on diet information (the arrow points from the group eaten to the group that eats). Groups with drawn boxes were present at all sites, groups with dashed boxes were only present at some sites.
Histogram  	Histograms of gap frequency by duration for (a) drivers and (b) fluxes and histograms of the fraction of total gap length for (c) drivers and (d) fluxes.
Timeseries  	Timeseries of anomalous SLP at 40 0 W; 60 0 S (black line) and timeseries of NE SPI (blue line).
No-viz  Diversity Density Seep and vent fluid-fluxes (CH₄) 900 μM	Conceptual diagram of macrofaunal diversity, density and composition patterns along a fluid-flux gradient in the chemosynthetic ecosystems of the Guaymas Basin.
Pie Chart  	Pie chart showing the functional classifications. The 44 identified proteins were categorized into 8 different functional categories. The pie chart represents the percentage of identified proteins under each category. The percentages are shown within the pie chart.
Stack Area Chart  	Model Energy Budget . Area chart showing breakdown of energy usage stacked to show contribution to total predicted energy delivered.

Appendix F

Evaluation form for the variable selection algorithm

CHOOSE THE INTERESTING VARIABLES FROM THIS DATASET

About our research: We have created a system that reads the metadata from the biodiversity domain and then automatically selects the input variables for the dataset visualizations. The goal of these 2-D visualizations is to provide meaningful insight into the data and provide initial data exploration without any model hypothesis formation.

About the task: To evaluate this algorithm, we would like the participants to go through the description of the dataset provided in the section below. Then select the interesting variables from the next page. Please choose those variables that can best fit to visualize this dataset.

Dataset Abstract

In soil erosion research, it is widely accepted that vegetation is a key factor for the type and intensity of erosion. Thus, scientists have long recognized the importance of forests for erosion control and afforestation is a common measure of soil protection. However, the mechanisms of how forests protect the soil remain debated, and especially the role of biodiversity is unclear.
In this experiment, we quantified the initial soil erosion under forest using micro-scale runoff plots (ROPs, 40 cm x 40 cm). 70 study plots have been equipped with 5 ROPs each (350 ROPs in total). The study plots represent different levels of tree diversity ranging from 1 to 24 tree species mixtures and bare ground. The measurements took place during the rainy season from May to June 2013, with rainfall events showing intensities up to 85 mm h-1. We measured sediment discharge, runoff volume, soil surface cover and canopy cover in the field. In addition, organic carbon and nitrogen contents in eroded sediments were analysed.

Dataset Design

350 runoff plots on 70 VIPs covering tree diversity levels from 0 to 24.

Dataset Analysis

GLM, LME

Pawandeep Kaur, pawandeep.kaur@uni-jena.de

Tick(X)	Variable Id.	Variable definition
	PTAG	BEF China plot identification
	site	Experimental Site
	plot	VIP number
	rop	runoff plot identification
	timestep	timestep
	slope_rop	slope angle at every ROP
	slope	slope angle VIP
	asp	exposition
	asp_com	exposition class (N, S, W, E)
	altitude	VIP altitude
	surface_cov_tot	surface cover total
	surface_cov_crust	surface cover biocrusts
	surface_cov_stone	surface cover stones
	soil_dens	soil density
	spec_numbers	number of tree species within the respective experimental plot (also "plot diversity level")
	tree_cmp	tree composition surrounding the runoff plots. Species names in Latin annotated in a consecutive way with genus abbreviated
	ground_cover	leaf canopy cover
	lai	leaf area index
	start_date	start measurement
	sample_date	date of sampling
	days	days between start and sampling date
	precip	rainfall amount climate station A
	precip_hours	hours of rainfall
	intensity_mean	mean intensity
	intensity_peak	peak intensity
	precip_eros	rainfall amount classified erosive
	precip_eros_hours	hours of classified rainfall
	intensity_eros_mean	mean intensity (erosive event)
	intensity_eros_peak	peak intensity (erosive event)
	precip_events	Rainfall event no. (erosive)
	precip_rop	rainfall on ROP
	precip_rop_eros	precip_rop_eros
	runoff	surface runoff (1600 cm2)
	infiltration	infiltration (1600cm2)
	sed_dis	sediment discharge
	sed_dis_stan	sediment discharge standardized
	N_dis_t	N total in sediment
	N_dis_pc	N percentage in sediment
	C_dis_t	C total in sediment
	C_dis_pc	C percentage in sediment

Appendix G

Evaluation form for the visualization recommendation system

VISUAL EXPLORATION THROUGH BIODIV VIS TOOL

As a part of the evaluation of the thesis „Knowledge-Assisted visualization recommendation.“

Prerequisite: Biodiversity domain knowledge at Ph.D. level or above.

Step 1: Please read the metadata for dataset no……. provided by the evaluator.

Step 2: After you have read the metadata, list below the various insights that you would like to get from the corresponding dataset

Insights:

____

____

____

____

____

Step 3: Once done, enlisting the insights, visually explore the corresponding dataset in our visualization recommendation tool. For each visualization that you use produce, please fill the information below.

Visualization Name	**Observation**
____________________	______________________________
____________________	______________________________
____________________	______________________________
____________________	______________________________
____________________	______________________________
____________________	______________________________
____________________	______________________________
____________________	______________________________

Step 4: Once finished with the visual exploration, answer the questions below based on the insights you have gained.

A. Were you able to find all insights that you initially wanted to gain after reading the metadata?

B. Did you find any unexpected insight while exploring the dataset?

C. Did you find any new insight?

D. Visualizations that you find helpful in understanding the data?

E. Do these visualizations and subset of the data trigger further analytical questions/queries or understanding?

Any information you would like to provide that does not fall into any of the above questions?

Thank you very much for participating in this survey.

Appendix H

Notes from the qualitative evaluation

Excel Chart Recommendation tool - ERT

Participant 1

577 Excel

- Participant used ERT but it did not show anything to them, then he used manual charting option.
- Boxplot was not readable by Participant. It needs transformation to show better boxplots.
- Switching between metadata and dataset for better understanding.

24209 Our tool

- There is no hover functionality for boxplot.
- He asked about why only the subset was shown. He wanted more variables.

Overall

- Some needed variables were missing.
- Mostly interesting variables were shows by the tool.
- Great for exploration.
- Ways to include colour manually.

Participant 2

577 Our Tool

- Data shown by boxplot was not much helpful. However, he liked the violin plot. He might have not thought about it if it was not included in the list of recommended viz. Adding alternative visualization aid in better data recommendation.
- Size in scatterplot should also change by integer and not only by other variables.
- Currently scatterplot is showing only for categories and not for discrete data or linear measurement values.
- Participant said it is a difficult and complex data.

24209 Excel

- Helped Participant to select X and Y categories as he was not able to find the option.
- Raw data does not provide the complete information as it is mentioned in the metadata.
- He used data transformation and other Excel analytic function to understand the data best.

Overall

- Tool did better as compare to Excel as Excel is more for business or financial domain and does not suit to the demand of biodiversity analysis.
- Jensel effect needs density which was missing in the recommended subset and BefChina has less dataset related to density.
- Variable subset algorithm should be learned based on the phenomenon like spatial heterogeneity and horizontal heterogeneity.
- Selected variables show general project design and project based interesting variables but not in-hand dataset based.

Participant 3

577 Our tool

- Participant was navigating between tool /raw data / metadata while exploration.
- Participant could not find distance variable which he wanted to explore in the data. Told that we have only taken subset of variables suggested by our algorithm.

24209 Excel

- Participant said it is easier dataset than 577.

Overall

- Liked environment of tool.
- There should be option to select more explanatory variables.
- After using both Excel and the biodiversity visualization tool, I would prefer the second one due to the inclusion of multiple explanatory variables at once in an easy way.

Participant 4

577 Excel

- The screen recording stopped twice. Paused the evaluation and had to start again.
- Participant used ERT twice but it did not show anything. Then he moved to MCT and transformation and Excel functions.
- Participant prefer first to ask the data owner to describe the dataset and then Participant will start exploring the data by its own.

24209 Our tool

- Participant asked about why only the subset was shown. Participant would want more variables.
- When asked, the difference between Hexagonal Binning and Scatterplot was told.
- He said that zoom out in in treemap is not very intuitive.
- Node-link dataset not showing all values of edges with water_content.
- When asked told that no transformation of the raw data is done before visualizing.
- Participant was doing more of tool driven exploration than data driven exploration.
- Alluvial diagram shows experimental setup and not analysis and exploration.

Overall

- Participant found tool to be easy and intuitive.
- When told how the tool will be used in future, Participant said that it is better to use the tool and plot randomly rather than downloading each dataset.
- Participant wanted to see more multidimensional plots for correlation like Scatterplot Matrix and Parallel Coordinates
- Certain columns are not enough and need whole dataset.
- The recommended subset of variables were reasonable but not enough.

- Liked freedom of selection of variables with drop down buttons.

Participant 5

20106 Our tool

- Switching between tool to metadata to understand the variable definition better.
- Participant asked about why only the subset was shown. Participant would want more variables.
- When asked, the Hexagonal Binning was explained.

376 Excel

- Participant was not able to understand some of the variables in the metadata like precip_eros and ground_cover. Coordinator being not biodiversity person could not help much in that.
- Participant was more interested in exploratory through Excel viz thumbnails rather than enlarged viz.
- Build in functions for grouping and filtering were used.

Overall

- Convenient as with Excel too many variables need too much scrolling and then viz construction. In tool, it is all in one list.
- New plots and new experience with tool.
- Some variables like mean_dbh was not there due to which important viz cannot be created.
- Variable definition should also be included in the plotted area.
- She recommended to create variable selection algorithm trained on variable names in the visualizations of the biodiversity publication and the described concept within the caption. Better to use publications from the same project for better subsetting.
- Variable subset algorithm should be learned based on the phenomenon like spatial heterogeneity and horizontal heterogeneity.
- People can get more merits of software when they know it much better.

Participant 6

20106 Our tool

- Participant would prefer to have variable definition in the plotting area.
- Participant prefer first to ask the data owner to describe the dataset and then will start exploring the data by its own

376 Excel

- Participant did exploration on its own Mac as Participant was more convenient in it. Therefor no recording for this session is available.
- Participant was creating lots of composite variables and using lots of transformation.

- Participant was not able to use build in functions for grouping as she is not much use to Excel.

Overall

- It shows lots of relationships but it is not possible to know from tool the cause of these relation. Need corrections (statistical correction of estimates) for that.
- It should also show how many observations (n) the plot is based on.

Participant 7

376 Our tool

- Participant asked about why only the subset was shown. Participant would want more variables like species_number.

Overall

- Liked how tool gives ideas to visualize specific variables and data.
- User friendly
- It had eliminated the variables which could be useful.
- It is good to explore all the variables.
- Hover option are good to see the specific values on the chart.
- Hover option should also be available for scatterplot.
- It identified outlier without doing deep exploration just by plotting with mouse.
- Liked clustering visualizations based on goals. Like goal-based visualization exploration technique.
- It should also give statistical summary of a dataset when one clicks on the dataset number.

Participant 8

376 Our tool

- Participant asked about why only the subset was shown. Participant would want more variables like altitude, tree_cover and slope.
- Recommended variable sets tells obvious things about the dataset. Available variables are common or general variables. They are output related variables but they don't give deeper understanding that could give more hint about the reasons or causation.
- Participant would prefer to have variable definition in the plotting area.

20106 Excel

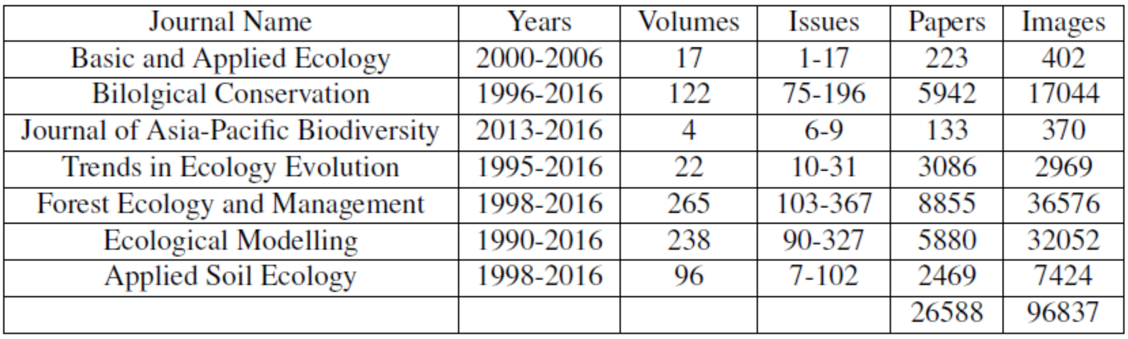
- All variables are available there is also lots of redundancy. Still it is better to have more than less.
- Some variables like no. of tree, basal area and volume are important for forest inventory and they are missing in the dataset.

Overall

- A better variable selection algorithm will avoid having too many collinear variables like erosion_mean and erosion_peak. It could do collinearity check for that. It should include spatial variables like altitude, aspect(N,W,E,S) and temporal variables. It should include all categorical variables and do collinearity check for measurement variables.
- For general users the recommended variables are fine but for experts it is not.
- Easy to use
- Good for overall goal of the research like what kind of graph can I apply? It answers this question very well.
- Spatial dimension is missing. Should have provisions of maps.
- Temporal dimension is missing. It should include that.
- Distribution to understand data is good goal for this dataset.

Appendix I

Breakdown of the downloaded publications



Journal Name	Years	Volumes	Issues	Papers	Images
Basic and Applied Ecology	2000-2006	17	1-17	223	402
Bilolgical Conservation	1996-2016	122	75-196	5942	17044
Journal of Asia-Pacific Biodiversity	2013-2016	4	6-9	133	370
Trends in Ecology Evolution	1995-2016	22	10-31	3086	2969
Forest Ecology and Management	1998-2016	265	103-367	8855	36576
Ecological Modelling	1990-2016	238	90-327	5880	32052
Applied Soil Ecology	1998-2016	96	7-102	2469	7424
				26588	96837

Figure 43: Breakdown of the downloaded publications from different journals