TOWARDS COMBINING SEARCH AND EXPLORATION

Escaping the Filter Bubble through Map-Based Exploration

DISSERTATION

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

angenommen durch die Fakultät für Informatik
der Otto-von-Guericke-Universität Magdeburg

von Dipl.-Inf. Thomas Low
geb. am 09.11.1986 in Altenburg

Gutachterinner/Gutachter

Prof. Dr. Andreas Nürnberger
Ass.-Prof. Dr. Wolfgang Hürst
Prof. Dr. Stéphane Marchand-Maillet

Magdeburg, den 21.01.2022

Towards Combining Search and Exploration – Escaping the Filter Bubble through Map-Based Exploration by Thomas Low, Dissertation, 21.01.2022

ABSTRACT

Exploration is an integral method to learn about unfamiliar environments such as digital information spaces. Unfortunately, state-of-the-art search engines and search user interfaces do not provide sufficient support for exploratory user behaviour, even though information spaces have reached an incomprehensible scale such that it is impossible for users to be familiar with every aspect of them. This thesis combines and evaluates methods from the research areas of Human Computer Interaction, Information Retrieval and Machine Learning to support users during exploration and exploratory search tasks. A first approach augments classic list-based search result visualizations by annotating results with concepts from a domain ontology. Users benefit by being able to easily identify search results that are related to already known concepts from a personal ontology that can be interactively extended during a search process. The main contribution of this thesis is a novel map-based approach called visual berrypicking, which allows to iteratively explore local neighbourhoods of similar information objects. Results of a k-nearest neighbour search are visualized in a two-dimensional similarity-based map and subsequent maps are aligned to create the impression of navigating a global map of the information space. A detailed analysis of dimensionality reduction methods has revealed that multidimensional scaling in combination with Procrustes analysis achieves a good balance between projection accuracy and map consistency. Several incremental user studies have demonstrated that the proposed interaction design can be used to efficiently retrieve images, find interesting links between scientific papers, and discover novel movies from a large movie collection. In addition, since exploration is considered a highly user-specific process, which depends on many factors like a user's prior knowledge or individual user abilities, several approaches for personalization are investigated. For example, users are enabled to interactively adapt feature weights. A user study evaluates the possibility to use gaze and pupil tracking for personalization. And a final prototype demonstrates how similarity-based maps can be adapted based on direct manipulations of map items.

ZUSAMMENFASSUNG

Die Exploration von noch unbekannten Umgebungen ist eine elementare Methodik neue Erkenntnisse zu gewinnen. Leider bieten aktuelle Suchmaschinen bisher keine hinreichende Unterstützung digitale Informationsräume zu explorieren, trotz dass es aufgrund deren unvorstellbaren Größe inzwischen unmöglich geworden ist, dass Nutzer mit allen Facetten eines Informationsraumes vertraut sein können. Diese Dissertation verknüpft Methoden aus den Bereichen der Mensch-Computer-Interaktion, des Information Retrievals und des Maschinellen Lernens, um Nutzer bei der Exploration und explorativen Suche zu unterstützen. Ein erster Ansatz erweitert die klassische Listendarstellung von Suchergebnissen und annotiert Suchergebnisse mit Konzepten einer Ontologie. Die Nutzer profitieren, indem sie Suchergebnisse einfacher mit bereits bekannten Konzepten der Ontologie in Verbindung bringen können, sowie indem sie neu erlernte Konzepte während des Suchprozesses zur Ontologie hinzufügen können. Der Hauptbeitrag der Dissertation ist ein neuartiger interaktiver Ansatz zur Exploration von lokalen Nachbarschaften mit Hilfe von Kartendarstellungen, genannt "Visual Berrypicking". Die Ergebnisse einer k-nächsten Nachbarschaftssuche werden basierend auf deren paarweisen Ähnlichkeiten als zweidimensionale Karte dargestellt. Karten von angrenzenden Nachbarschaften werden aneinander ausgerichtet, sodass für die Nutzer der Eindruck entsteht, sie würden eine große, globale Karte des gesamten Informationsraumes erkunden. Eine umfangreiche Analyse aktueller Verfahren zur Dimensionsreduktion hat ergeben, dass die Kombination von Multidimensionaler Skalierung und Procrustes Analyse eine gute Balance zwischen Projektionsgenauigkeit und Kartenstabilität erreicht. Mehrere aufeinander aufbauende Nutzerstudien haben gezeigt, dass der vorgeschlagene Ansatz dazu genutzt werden kann, effizient Bilder zu finden, aufschlussreiche Zusammenhänge zwischen wissenschaftliche Veröffentlichungen zu erkennen, und interessante Filme in einer großen Filmdatenbank zu entdecken. Des Weiteren werden verschiedene Methoden zur Personalisierung untersucht. Zum einen ist es möglich die Kartendarstellung über eine Einstellung der Gewichte von Objektmerkmalen an die Präferenzen eines Nutzers anzupassen. Zum anderen wurde eine Nutzerstudie durchgeführt, die untersucht, inwiefern Blick- und Pupillenparameter genutzt werden können, um einen Explorationsprozess zu personalisieren. Abschließend demonstriert ein finaler Prototyp, wie Kartendarstellungen mit Hilfe von Verschiebeoperationen an die individuelle Vorstellung eines Nutzers angepasst werden können.

RELATED PUBLICATIONS

A list of my publications ordered by publication date:

[pub:1] T. Gossen, T. Low and A. Nürnberger. 'What are the real differences of children's and adults' web search'. In: *Proc. of the 34th Int. Conf. on Research and Development in Information Retrieval*. SIGIR. 2011, pp. 1115–1116 (cit. on p. 55).

[pub:2] A. Hein, T. Low, M. Hensch, T. Kirste and A. Nürnberger. 'Gesture Spotting for Controlling a Mobile Assistance System for Service and Maintenance.' In: *GI-Jahrestagung*. Vol. 208. LNI. 2012, pp. 549–560 (cit. on p. 168).

[pub:3] T. Low, C. Borgelt, S. Stober and A. Nürnberger. 'The Hubness Phenomenon: Fact or Artifact?' In: *Towards Advanced Data Analysis by Combining Soft Computing and Statistics*. Vol. 285. Studies in Fuzziness and Soft Computing. 2013, pp. 267–278 (cit. on pp. 23, 105).

[pub:4] S. Stober, T. Low, T. Gossen and A. Nürnberger. 'Incremental Visualization of Growing Music Collections'. In: *Proc. of the 14th Int. Society on Music Information Retrieval Conference*. ISMIR. 2013, pp. 433–438 (cit. on p. 89).

[pub:5] T. Low, C. Hentschel, S. Stober, H. Sack and A. Nürnberger. 'Visual Berrypicking in Large Image Collections'. In: *Proc. of the 8th Nordic Conf. on Human-Computer Interaction: Fun, Fast, Foundational*. NordiCHI. 2014, pp. 1043–1046 (cit. on pp. 99, 114).

[pub:6] P. Butka, T. Low, M. Kotzyba, S. Haun and A. Nürnberger. 'A Framework for FCA-based Exploratory Web Search'. In: *Proc. of the 1st Int. Symposium on Companion-Technology*. ISCT 1. 2015, pp. 131–136 (cit. on p. 69).

[pub:7] M. Kotzyba, D. Ponomaryov, T. Low, M. Thiel, B. Glimm and A. Nürnberger. 'Ontology-supported Exploratory Search for Physical Training Exercises'. In: *Demonstrations Track at the 14th Int. Semantic Web Conference*. Vol. 1486. ISWC. 2015 (cit. on pp. 61, 64).

[pub:8] S. Stober, T. Low, C. Hentschel, H. Sack and A. Nürnberger. 'The ISMIR Paper Explorer: A Map-Based Interface for MIR Literature Research'. In: *Extended Abstracts for the Late-Breaking Demo Session of the 16th Int. Society for Music Information Retrieval Conference* (2015) (cit. on p. 126).

[pub:9] T. Low, N. Bubalo, T. Gossen, M. Kotzyba, A. Brechmann, A. Huckauf and A. Nürnberger. 'Towards Identifying User Intentions in Exploratory Search Using Gaze and Pupil Tracking'. In: *Proc. of the Conf. on Human Information Interaction and Retrieval*. CHIIR. 2017, pp. 273–276 (cit. on p. 144).

[pub:10] T. Low, C. Hentschel, S. Stober, H. Sack and A. Nürnberger. 'Exploring Large Movie Collections: Comparing Visual Berrypicking and Traditional Browsing'. In: *Proc. of the 23rd Int. Conf. on MultiMedia Modeling*. MMM. 2017, pp. 198–208 (cit. on p. 133).

[pub:11] P. Butka, T. Low, M. Kotzyba, S. Haun and A. Nürnberger. 'Exploration of Web Search Results Based on the Formal Concept Analysis'. In: *Proc. of the 3rd Int. Conf. on Semantic Keyword-Based Search on Structured Data Sources*. IKC. 2018, pp. 123–135 (cit. on p. 69).

[pub:12] T. Low, C. Hentschel, S. Polley, A. Das, H. Sack, A. Nürnberger and S. Stober. 'The ISMIR Explorer - A Visual Interface for Exploring 20 Years of ISMIR Publications'. In: *Proc. of the 20th Int. Society for Music Information Retrieval Conference*. ISMIR. 2019, pp. 392–399 (cit. on p. 126).

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

This thesis is structured into four parts. Part i will recap important fundamentals related to the main contributions discussed in parts ii, iii and iv. Part ii proposes methods to enrich traditional list-based keyword search systems. Part iii presents and evaluates a novel approach to exploratory search in text and media collections. And Part iv will give an outlook to future research opportunities and summarize the thesis. Before going into the details, the following Sections will introduce the general topic. Section 1.1 will motivate the importance of this research. Section 1.2 provides concise research goals that will be the focus of this thesis. Finally, Section 1.3 gives a detailed outline.

1.1 BACKGROUND & MOTIVATION

Today, search engines provide the most important digital door to the ever-growing vast amount of digital information. Without search engines most information would not be efficiently accessible due to its sheer incomprehensible scale, whether it being all web pages of the whole web, the collection of all scientific literature, a company's library of business regulations and contracts, or just one's own email storage. Search engines promise to provide all available information at one's fingertips: every day, instantaneously. But do they actually keep this promise?

Ranking Bias and Inaccessible Search Results

Search engines are used in a large variety of ways, from a simple search intended to quickly navigate to a website to complex business or research scenarios. These diverse usage cases provide a major challenge both for ranking algorithms, which determine the order in which results are presented, as well as the design of search user interfaces. Exploratory search intentions, e.g., when researching a previously unknown topic, are usually less well supported than more common search tasks, e.g., searching for a specific fact, entity or website. When researching general topics, search engines usually provide thousands or millions of matching search results, which are not practically accessible to users due to their list-based presentation and huge quantity. Users mostly only inspect the first few search results, see [134], missing out on potentially relevant, or even eye-opening new information. This highlights the importance of the implemented ranking algorithm. The exact ranking calculation directly influence which information is accessible, and what information disappears in the vast space of ignored search results.

Filter Bubbles and Personalization

In addition, in-transparent personalization methods hide important factors that determine which results are presented to users. Users are left without control and are limited to an unseizable subspace of all available information depending on shallow behaviour analysis that is based on, e.g., prior search queries, or the results a user clicked on. This explicit information confinement based on alleged personal preference is known as a filter bubble. In simple search scenarios, e.g., when repeatedly searching for the same entity, an adaptation of the ranking calculation based on prior user behaviour may even improve user satisfaction. However, for complex research scenarios ranking personalization will most likely lead to users being isolated from information about certain concepts, ideas, or opinions. Even worse, users will likely confirm and reinforce their own filter bubble since only information that conforms to current personalization parameters is presented.

Breaking out of Filter Bubbles

Exploration methods aim to provide means to efficiently discover new information. By integrating exploration methods in state-of-the-art search systems, users would benefit from a more diverse search experience, effectively breaking out of filter bubbles. In this thesis, a special focus is given to methods that provide an overview over a data collection, e.g., search results. Provided with an overview over search results, users would be enabled to discover search results they would otherwise not have noticed.

1.2 RESEARCH GOALS

Primary goal of this thesis is to develop and evaluate methods that integrate classic keyword-based search and novel exploration concepts for common use cases, e.g., web search or media exploration.

In order to achieve this goal, methods from three major research areas are adapted and combined. Figure 1 illustrates how each of the following research questions (RQ) is related to the research areas of Human Computer Interaction (HCI), Machine Learning (ML) and Information Retrieval (IR). Since exploration is a highly dynamic process, it is considered necessary to integrate methods from all three research areas in order to develop an effective approach for supporting exploratory search.

Two general approaches are studied. In the beginning of this thesis, classic list-based search result presentations are enriched with additional domain-specific information in order to improve the user's efficiency of evaluating search results. The main contribution are methods that utilize similarity-based maps as a means to provide an overview during exploration and exploratory search tasks. In the

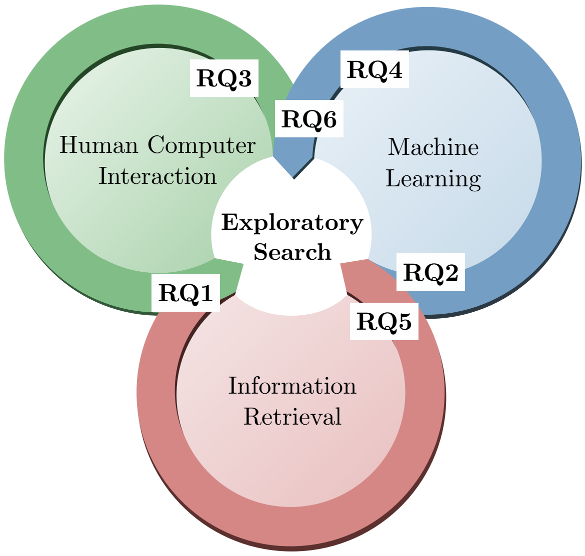


Figure 1: Exploratory search is enabled by combining methods from the research areas of Human Computer Interaction (HCI), Machine Learning (ML) and Information Retrieval (IR). Each research question (RQ) addresses challenges that can be linked to at least two of all three research areas.

following, the main research goals of this thesis are presented and motivated in terms of research questions (RQ).

(RQ1) IDENTIFY OPPORTUNITIES FOR SUPPORTING EXPLORATORY SEARCH

What are promising approaches for supporting exploration and exploratory search tasks?

In order to develop effective new methods, first promising approaches have to be identified. This is achieved through studying and analysing state-of-the-art methods, observing and evaluating user behaviour, identifying problems and open challenges as well rapid prototyping of various ideas.

Section 2.3 and 2.4 discusses various challenges related to supporting exploration and developing interactive information systems. In Section 3.3 state-of-the-art methods for supporting exploratory search are presented. Chapter 5 investigates the search behaviour of users for state-of-the-art web search engines by performing a log file analysis. And Chapters 6 and 7 introduce two promising approaches and their respective prototypical implementations. Chapter 6 proposes the augmentation of search results lists using a domain ontology. Chapter 7 describes a method to structure search results in a concept graph.

(RQ2) LARGE SCALE MAP PROJECTIONS FOR EXPLORATORY SEARCH

Is there an efficient map-based projection approach that can be utilized to augment search and exploration systems given the computational complexity and accuracy of state-of-the-art projection methods when projecting large scale real world data sets?

Considering that in many cases of using state-of-the-art search methods on real world data sets, users are often overburdened with hundreds or even millions of relevant documents presented as a simple list, a map-based approach could visualize many or even all results at the same time, and thus, provide meaningful structural information about the result set distribution. For exploratory search scenarios, users would greatly benefit from a map-based visualization by being able to observe the full variety of search results, and by being able to infer information about the relevancy of clusters of search results from inspecting a few representative objects of that cluster.

In Chapter 8, challenges and limitations of recent map-based projection methods are discussed in detail. Unfortunately, computational complexity of state-of-the-art projection methods poses a significant challenge when generating and visualizing large maps of dynamic data subsets such as keyword-based query results.

(RQ3) EFFECTIVE MAP-BASED EXPLORATORY SEARCH

Given that computational complexity of state-of-the-art projection methods limits the practicality of large scale map projections, is there an approach that avoids high computational costs when generating maps but still allows for effective exploration of large data sets?

Users would still benefit from a map-based exploration approach, even if only partial projections of the data set can be efficiently generated. Partial projections would still provide partial information about the structure of the data and result set, which then can be utilized to, e.g., observe local cluster structures and infer information about the relevancy of local search result clusters.

In Chapter 10 and in particular in Section 10.4, a method and prototype for similarity-based map exploration is proposed, which avoids high computational costs by interactively combining overlapping small maps in order to create the impression of exploring a large continuous map of information objects. In Sections 11.1, 11.2 and 11.3, multiple incremental user studies are conducted and described, which aim at validating the proposed method and allow to conclude that it is in fact an effective approach for supporting exploration and exploratory search.

(RQ4) MAP CONSISTENCY UNDER DATA CHANGES

How can similarity-based maps be consistently visualized to the user considering that real world data sets often change over time?

State-of-the-art projection methods usually suffer from either degrading projection accuracy when the underlying data changes, or they generate a completely new and structurally different map, which confuses users that were already familiar to the structural design of the previous map. An approach that balances consistency and projection accuracy would enable users to transfer knowledge from one map to the next, and thus, improve the overall effectiveness of map-based exploration methods.

In Section 9.1, state-of-the-art projection methods are discussed with respect to their advantages and disadvantages for map-based exploration. In Section 9.2, a selection of projection methods are compared in the scenario of a growing data collections. Also, a user study for evaluating which method provides a good compromise between consistency and projection accuracy is presented. In Section 9.3, the problem of overlapping thumbnails is discussed, and a grid regularization method is proposed, which helps to avoid overlappings when visualizing objects via thumbnails in a similarity-based map.

(RQ5) LIMITS OF MAP-BASED EXPLORATION

Are there algorithmic or structural limits restricting a user's ability to explore an information space using similarity-based maps?

Without any doubt, users will be faced with biased visualizations of the data, e.g., favouring certain features or objects, or even omitting certain information. Since not all information can be presented to users in full detail at the same time, some bias is necessary and desired. However, depending on the specific algorithms involved, there might be non-desirable bias that will limit the user's ability to effectively explore an information space, e.g., information that is not accessible to the user due to algorithm limitations.

In Section 2.4 and 10.3, challenges in high-dimensional spaces are discussed, and in particular, the effects of the Hubness phenomenon is investigated when navigating information spaces via k-nearest neighbour search.

(RQ6) PERSONALIZED MAP EXPLORATION

Are there methods allowing to personalize the process of exploring information spaces using similarity-based maps with and without invasive user behaviour analysis?

Since real world data sets often consist of high dimensional feature spaces, and not all information is relevant to every user in every scenario, methods for personalizing the process of exploration can improve its effectiveness. User would benefit from a tailored exploration experience, which provides only the most relevant information given a user's current information need. Considering that data privacy and data minimization is a fundamental interest of all users,

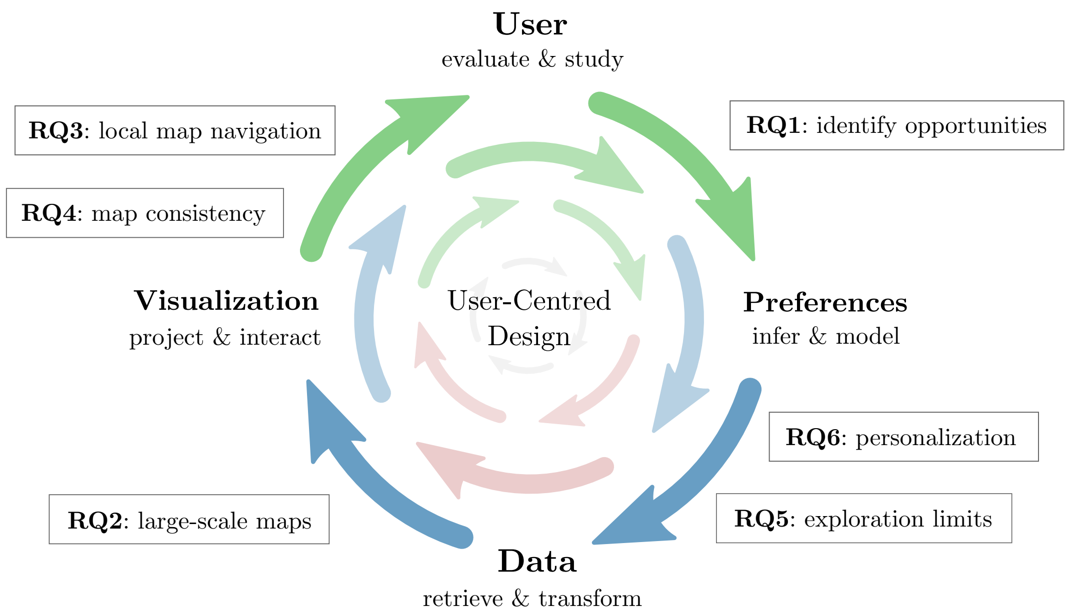


Figure 2: Illustration of user centred design method: research questions (RQ) are answered by performing user studies and evaluating user behaviour in order to refine methods in all areas supporting exploratory search tasks, including methods for modelling user preferences, for retrieving and transforming data, and for interacting with visualizations.

both invasive and non-invasive methods should be researched, i.e., methods that are based and are not based on user observation or user behaviour analysis.

In Section 10.2, a non-invasive personalization method is presented, where users may adapt the exploration process by interactively changing feature weights of the k-nearest neighbour search. In Section 11.4, a preliminary user study is conducted, which investigates the possibility of invasive personalization by tracking the user's gaze and pupil dilation while interacting with a map-based exploration system.

These six goals form the basis for this thesis and will be discussed in detail throughout the following chapters. In order to answer these research questions, a user centred design approach is followed, see Figure 2. Since exploration is a highly dynamic and user-specific process, the developed approaches are evaluated by performing various user studies and analysing user behaviour. Therefore, multiple incremental prototypes are implemented, which allow to evaluate different aspects of the overall system design in context of the research questions posed above. Based on the results gained by prior user studies, a final comprehensive prototype for Movie exploration is implemented, which integrates selected methods. A comprehensive comparative user study is performed to judge the overall effectiveness of the map-based approach for supporting exploration and exploratory search tasks.

1.3 THESIS OUTLINE

This thesis consists of four main parts: a revisit of relevant fundamental methods and algorithms, the two sections presenting main contributions of this thesis, and a discussion of conclusions and future research ideas. First, Chapter 1 provides a short introduction to the topic of search and exploration. Section 1.1 motivates the goal of supporting exploratory search scenarios. Section 1.2 establishes six main research goals that are the basis of the following chapters. And Section 1.3 provides this outline of the thesis.

In Part i, a number of fundamental terms, methods and algorithms are introduced, which will be referenced to throughout the thesis. Chapter 2 introduces the concepts of exploration and exploratory search by providing a definition (Sections 2.1 and 2.2) and discussing current challenges (Sections 2.3 and 2.4). Chapter 3 discusses several state-of-the-art approaches for exploration and exploratory search: graph-based exploration systems (Section 3.1), map-based exploration systems (Section 3.2) and exploratory search approaches (Section 3.3). Then, in Chapter 4, a selection of state-of-the-art algorithms are introduced. Section 4.1 describes the fundamentals of formal concept analysis. Section 4.2 focuses on various algorithms for dimensionality reduction. Sections 4.3, 4.4 and 4.5 describe the methods Procrustes analysis, nearest neighbour search and metric learning, respectively.

Part ii discusses several novel approaches to enrich keyword-based search in order to support users during exploratory search. In Chapter 5, current search behaviour is first analysed based on a log analysis of state-of-the-art list-based search engines. Then, in Chapter 6 a novel method for enhancing search result list presentations based on a domain-specific ontology is presented. Chapter 7 proposes to structure search results in a concept hierarchy using formal concept analysis.

Part iii focuses on map-based visualization approaches to support users during exploratory search. First, Chapter 8 reviews the challenges of map-based exploration. Then, in Chapter 9 several projection methods are compared and evaluated for their suitability. Chapter 10 proposes the main contribution of this thesis: a novel interaction design for map-based exploration, called visual berrypicking. Chapter 11 evaluates various aspects of the proposed methods in several user studies. First, in Section 11.1 a study on artificial data is used to validate the effectiveness of the visual berrypicking method. Then, in Sections 11.2 and 11.3 two studies are described, which evaluate the method using two real world datasets: a set of scientific papers, and a collection of movies. In Section 11.4, preliminary results of a user study are presented, which aims at personalizing the exploration process through the means of gaze and pupil tracking. Finally, in

Chapter 12 an additional method for personalization is presented on the basis of direct manipulations of map items.

Part iv discusses final conclusions and presents two future research ideas. Chapter 13 revisits all six research goals and provide a summary of the main contributions of this thesis. Finally, Chapter 14 presents future research ideas. Section 14.1 discusses how advanced map interactions could further improve usability. And Section 14.2 proposes to generalize exploration methods in order to be able to explore the space of views on the data.

Part I

FUNDAMENTALS

2 EXPLORATION OF INFORMATION SPACES

The continuous growth of digital data prompts the advancement of efficient tools to interact with increasingly complex digital information spaces. The tool set for exploring information spaces - as one type of possible interaction - may be perceived as lacking in contrast to the vast abundance of modern search, analytical, expert, and administrative tools. In the following, an introduction to basic terminology, related concepts and relevant technical challenges for exploring information spaces is given. Section 2.1 derives a definition for information exploration. Due to the strong relationship between exploration and search, Section 2.2 gives an introduction to the concept of digital search and its exploratory perspective. In Section 2.3, important challenges related to the development of efficient exploration tools are discussed. Finally, Section 2.4 focuses on specific technical challenges that arise for high-dimensional information spaces, which will be further discussed later throughout the thesis.

2.1 EXPLORATION — A DEFINITION

Exploration is a elementary process of life itself. When considering the development process of children, exploratory behaviour is deemed a central objective for successfully managing life, i.e., re-evaluating knowledge based on new observations [37]. The Oxford dictionary [book:10] defines the term exploring as travelling through an unfamiliar area in order to learn about it, e.g., a child exploring its surroundings. Another meaning refers to the comprehensive investigation or discussion of a subject, e.g., when exploring a topic or hypothesis. Exploration may also co-occur with a search goal, e.g., when exploring an area for resources such as mineral deposits. Exploration and search processes can be highly intertwined and need to be discussed further. In the following, a definition of information exploration is developed. Later in Section 2.2, search is discussed as a tool that, unfortunately, only inadequately supports exploration.

Properties of Exploration

Typical exploration settings share a number of common properties. There is an entity that performs the act of exploration, e.g., a child, person, animal or robot. There is a space or environment that is unfamiliar to the entity, e.g., a physical space or information space. And the entity is moving through, observing and interacting with this unfamiliar environment with the goal of learning about it, e.g., its structure, properties and limits.

Information Exploration

In this thesis, the process of information exploration is defined as a sequence of interactions performed by a user with the goal of learning about an unfamiliar information space. An information space is described as a collection of digital information, e.g., information about objects, their properties and relationships, which is stored as data. There can be many diverse information spaces of different size and scope. Typical information spaces are, e.g., the set of globally accessible web pages, the combined collection of all scientific literature, or merely the set of personal emails. In addition, the results of learning about an information exploration can be highly diverse as well, e.g., when exploring the web for clothing web shops, users may discover new favourite shops or special deals; when exploring a book collection, users may discover new authors or interesting topics; or when exploring a collection of emails, investigators may discover new evidence as part of a criminal investigation. In this thesis, several information spaces are used as part of prototype experiments and user studies: the web, a collection of scientific publications, an image data set, and a collection of movies.

Moving Through Information Spaces

In contrast to exploring physical spaces, the act of moving through, observing and interacting with an information space is not well defined. Information spaces often do not conform with the 3-dimensional spatiality of physical spaces, and thus, principals of physical exploration can not be easily applied to digital spaces. Thus, novel methods for exploring information spaces need to be developed. In the most trivial form, moving through an information spaces can be implemented as a linear process of iteratively inspecting one information object after another. However, similar to unguided physical exploration, such a procedure can not be considered effective. Instead, digital equivalents for tools that support exploration are needed, e.g., cartography tools like a compass or map. Given effective and easy to use digital exploration tools, large and complex information spaces can be made accessible to a wide audience of users. Hence, the goal of this thesis is to advance research in methods that support exploration of information spaces. Throughout the remainder of this thesis, the term exploration refers to information exploration except where otherwise noted.

Exploration of Media Collections

A modern day example for information exploration can be found in web platforms that provide access to vast and continuously growing movie or music collections. While there are other use cases of these platforms, exploration is a common method for users to discover new music to listen to, or movies to watch. Since a one-by-one iterative exploration of these huge media collections is practically infeasible, most platforms do not even offer the ability to browse through their databases one item at a time. Instead, they rely on a combination of tools that support exploration to a varying degree. Media objects are grouped by categories and sorted by various user metrics, e.g., most popular, most liked, and so on. Item recommendations suggest related media based on similar user behaviour. And search tools allow to find matching results based on user defined keywords. While these methods provide some insight into an information space, in this thesis, it is argued that they are not sufficient to effectively support exploration. Instead, tools are needed which provide a intuitive perspective on the structure, properties and limits of an information space. Grouping media by categories and sorting them by user metrics only provides a very narrow view on a small selection of objects. Even worse, some sorting criteria, e.g., by popularity, can be considered self reinforcing, such that popular items remain popular simply because they are more easily accessible to users. Recommendation systems suffer from the same disadvantage. And search tools usually require a specific information need that can be expressed as keywords, although many users have learned to use them in various ways that also facilitate exploration.

Exploratory User Behaviour

Often, there already exists partial knowledge about an information space. Then, other user behaviour can be mixed with exploration, in particular classic search, filtering, navigation or browsing behaviour. For example, when exploring a movie collection, most users will already know some movies, and thus, may decide to start searching for their favourites, and continue exploring from there. Some users may not like certain movie genres, and wish to only explore movies from their favourite genres, which requires a prior filtering step. Given existing knowledge about an information space, usually many complex and intertwined scenarios are conceivable. In this thesis, user behaviour is considered exploratory if it serves the purpose of learning about an information space. Therefore, any kind of interaction, including search, navigation and browsing, can be considered exploratory behaviour if it serves the purpose of learning about an information space. However, if an interaction follows another purpose, e.g., a mere look-up of a fact or item, it is not considered exploratory behaviour.

Related Work

Waterworth and Chignell define information exploration as the process of "travelling to a region to find what (information) is located there" [157]. Amongst other aspects, they characterize information exploration by an "absence of a definite target in the mind of the user" [157]. They also highlight that users rely on an awareness of the immediate surrounding in order to be able to "decide where to go next within an information structure" [157]. However, they do not involve the goal of learning about an information space. Instead, they describe exploration through several scenarios in the context of browsing, navigation and querying.

In the following Section, the process of searching and its relation to exploration is discussed in more detail.

2.2 SEARCH AND EXPLORATORY SEARCH

Today, when working with large information spaces, classic query-based search is often the most prominent or sometimes even the only tool implemented by an information system. A prominent example are modern day web search engines, which solely rely on keyword-based search. However, in this thesis, classic search is considered an ineffective tool for information exploration. Further research is required in order to develop novel methods that combine state-of-the-art search systems and exploration components. In the following, the method of searching and its relation to exploration is discussed in more detail.

A Definition of Search

In comparison to the definition of exploration from Section 2.1, the process of searching can be viewed as the act of scanning an environment with the purpose of finding a selection of items that match well defined search criteria. Retrieving the set of matched items is considered the only goal, and thus, the result of a search process. Search criteria as well as search strategies can be manifold and depend on properties of the environment. A traditional example is the search for a word definition in a dictionary. The set of words, the environment, can be ordered lexicographically, which then facilitates an effective look-up search strategy.

Search in Information Spaces

Again, this thesis is only concerned about digital information spaces. In this case, users are often provided with a search system that implements a specific search strategy, such that users do not have to perform the actual search operations themselves. Depending on the kind of information space to be searched and depending on the kind of search intention, many search strategies and matching criteria are the focus of ongoing research. A comprehensive overview of information retrieval methods can be found in [book:1]. For example, when searching a collection of documents, e.g., web pages, a common search strategy is to use a list of query terms or keywords as search criteria in order to find documents that contain one or all of these terms. Throughout this thesis, in particular in Chapter 6 and Sections 11.2, 11.3 and 12, various prototypes are presented that integrate classic keyword-based search and additional exploratory methods. Another search scenario called near duplicate image retrieval aims at finding images in an image database that were derived from a given query image by common image operations, e.g., cropping, rotation, scaling, and more. Such a search system can be used to detect, e.g., copyright infringements, but also support users in finding similar images. In Section 10.4 a related prototype is presented, which allows to search and explore an image collection based on a more general sense of image similarity, focusing on colour and colour distribution in images.

Information Need and Result Relevance

Unfortunately, users often are not able to fully specify search criteria that exactly match their actual information need. This might happen either because it may not be possible to specify the required search criteria given all available options of the search system, or users simply fail to fully express their information need for a variety of reasons. The information need describes a user's true retrieval intent, and is defined as a subset of relevant items from the information space. When search criteria do not match a user's information need, the set of search results may lack some relevant items or contain irrelevant results.

Ranking

Besides providing a set of results, search systems also often implement a ranking of results. A ranking defines an order in which results are presented to users. It is often necessary, since many information spaces contain a huge number of items, and many search scenarios lead to a vast number of results, which can not all be examined by a user. Then, an ordering of results can help to prioritize results that are more likely to be relevant to the user's information need. In similarity-based image search, the ranking usually follows the definition of similarity, whereas more similar images are presented first. However, in the scenario of detecting copyright infringement, a ranking by similarity might not benefit the search goal since many legitimate uses of an image could be highly similar to the query image, and would be ranked before cleverly modified less similar illegitimate uses of an image. Therefore, ranking of search results constitutes a selection bias that may or may not be beneficial to the search process. In many scenarios, additional information is used for ranking, e.g., query statistics like the number of term matches in a document, see the Vector Space Model in Section 2.5.3 of [book:1], or user statistics like the number of times a result was clicked by other users.

Evaluation

If the set of relevant items is known for a given query, the effectiveness of a search systems can be objectively evaluated using various measures. Precision describes the fraction of retrieved items that are relevant to the query. Recall describes the fraction of relevant items that are retrieved by a search system. Together, they allow to measure the effectiveness of a search system to retrieve only relevant items given a query. In order to be able to compare various search strategies and implementations, a number of reference datasets for various retrieval scenarios have been developed, e.g., the Text Retrieval Conference (TREC) data sets [46]. A detail introduction to evaluation approaches can be found in Chapter 3 of [book:1].

Keyword-Based Search as a Tool for Exploration

When looking at classic keyword based search as a stand-alone tool for exploration, many problems arise. First, a search system alone does not provide a structured overview over the information space. Instead, the information space is only accessible via keywords that have to be provided by users. However, if users are fully unfamiliar with the information space, they might not be able to express any search intent, and thus, might have difficulties learning fundamental facts about the information space, e.g., what kind of information is contained

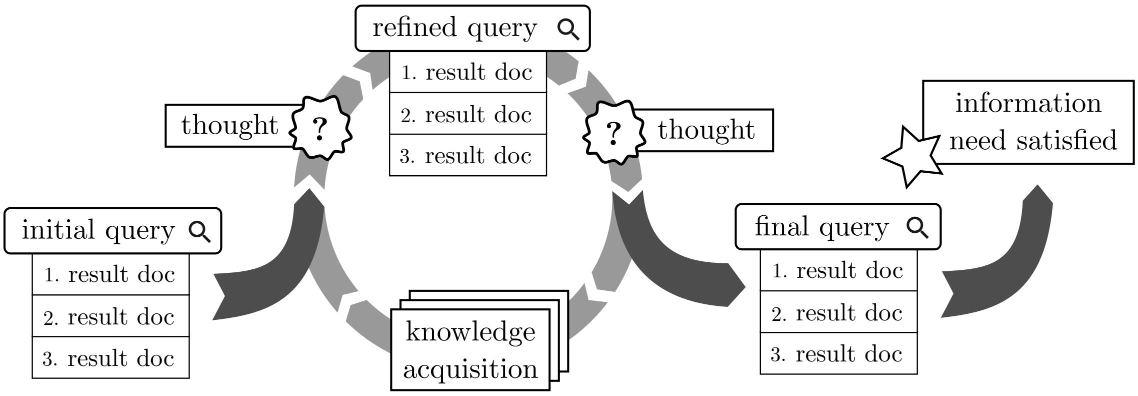


Figure 3: Illustration of berrypicking search behaviour: Search queries are iteratively refined after acquiring new knowledge from studying result documents until final information need is satisfied.

in it. This scenario can be viewed as a cold-start problem, i.e., user do not have sufficient information in order to effectively formulate queries. Additionally, if a search is performed, results are usually presented as an ranked list of items. While a list-based representation is beneficial for other search scenarios, e.g., a simple look-up of facts, in case of exploration it significantly restricts users in learning about the information space, i.e., learning about its structure, properties and limits. Therefore, most approaches discussed throughout this thesis will focus on enhancing or reinventing the presentation of and interaction with search results. For example in Chapter 6, the list-based presentation of search results is augmented by additional information from a domain ontology, which connects search results with ontology concepts in case a reference to these concepts is found inside the full-text document. In Chapters 7, 10 and 12 novel search result visualization and interaction methods are proposed. Furthermore, a classic search does not provide a meaningful way of moving through the information space, which is a fundamental concept of exploration. While a set of search results can be interpreted as an sub-area of the information space, there is no intuitive way to move to a neighbouring area, or get a sense of any navigation directions.

Berrypicking

Despite these disadvantages, in some cases, e.g., web search, there are no alternatives yet to classic keyword-based search. As a consequence, users established search behaviour patterns that exhibit exploratory components. In 1989, Bates [6] identified the pattern he coined *berrypicking*. Bates describes an evolving search, where users incrementally refine their information need based on new knowledge acquired during the search process. Over multiple iterations a search is performed, search results are analysed for new information (or "berries"), and a new refined query is deduced. At some point, the user's information need is satisfied and the incremental search process stops. Figure 3 illustrates this process. This evolving search can be viewed as moving through an information space by the means of query modification. At the same time, knowledge about the information space is gained, which conforms with the goal of exploration. Later in this thesis, in Chapter 10, this idea of an evolving search will be extended to a map-based exploration approach.

Exploratory Search

In 2006, Marchionini [87] establishes the notion of exploratory search, which describes the need to extend or adapt classic lookup search in order to support exploratory user behaviour. Marchionini defines exploratory components as aspects of a search system that support learning, knowledge acquisition and investigative activities. This thesis follows this idea of exploratory search. However, in comparison to Marchionini's broad definition, this thesis focuses on two facets of exploratory search in particular: learning about the structure of an information space, and serendipitous discoveries. Over the course of this thesis, several methods are developed that extend traditional keyword-based search systems in order to support users in learning about relations between items of an information space and to facilitate serendipitous discoveries of interesting items. In contrast, Marchionini views the aspects of learning and investigating in a broader sense, also including aspects of life-long learning by comprehending new concepts, or investigative analysis for planning and forecasting. In this thesis, these facets of exploratory search will not be discussed, since they would entail many other requirements and approaches that are out of the scope of this thesis.

In the following, various challenges related to the task of supporting exploration and exploratory search are discussed.

2.3 CHALLENGES IN SUPPORTING EXPLORATION

When designing systems that support users in exploring information spaces, many different aspects need to be considered. For example, users may have a mixture of different goals when using an exploration system, or they might have different prior knowledge about an information space. In the following, a short overview over various challenges is given, including challenges related to the wide variety of users, the diversity of information spaces, algorithmic challenges, and challenges when visualizing and interacting with complex information spaces. Some of these challenges are unique for exploration systems, others are universal and apply to interactive information systems in general.

Usage Diversity Challenges

As discussed in previous Sections, exploration is rarely done without any context. Users often follow a mixture of goals, whereas exploration is a first step in order to get familiar with an information space. For example, when exploring a collection of books, users most likely also have other goals in mind. They might try to find a new interesting book to read, or try to find a gift for a friend. When exploring the web for information on a specific topic, some users might want to educate themselves, others collect information for the purpose of writing a scientific article. Depending on the context, which might imply certain preferences for a subset of features or preferences for subspaces of the information space, the exploration process can be optimized. For example, when exploring the web with the goal of writing a scientific article, websites and resources with academic background might be more useful than general purpose ones. Users could benefit from an exploration system that allows to adapt to these contexts, e.g., by focusing the exploration process on the subspace of academic information. However, it is not obvious how information systems can obtain clues about the contexts they are used in. Apart from explicit information provided by users directly, the information system could try to infer contexts by analysing user behaviour. In [73], the authors analyse how users interact with state-of-the-art web search engines in order to be able to characterize user behaviour live during a search session and distinguish between the search behaviour pattern of fact finding and exploratory search. Given an effective online classification approach, search systems could be adapted during a search session, e.g., by showing longer result excerpts in case of fact finding.

Integrating Prior Knowledge

In addition, users may also have different prior knowledge about the contents and properties of the information space. Depending on prior knowledge, certain information might be new to some users, but already known to others. Accordingly, an exploration system needs to be able to adapt to different levels of prior knowledge about an information space. For example, a frequent reader might already know many books and authors, and is only interested in recent releases when exploring a book collection. In this case, only a certain subspace of not yet discovered information is of interest for an exploration process. Unfortunately, similar as before, it is not clear how users can effectively describe their prior knowledge, or how systems can effectively infer a user's prior knowledge about an information space. Some aspects of prior knowledge might be representable by simple filters, e.g., when books are annotated with their genre, and a frequent reader of horror novels would like to explore other genres, the set of horror novels could be excluded from the exploration process. However, in many cases, there exists no such convenient mapping. In case of recommender systems, a similar problem is known as the serendipity problem [90]. Users are unsatisfied for receiving recommendations of already known items, since recommendations are deduced from their own user profile, which in turn is based on past interactions with the same or highly similar items. In [23], the authors propose a graph-based approach that dampens the influence of highly similar items and promotes the generation of non-obvious recommendations. However, a user's prior knowledge is not directly modelled.

User Abilities, Disabilities and Impairments

Finally, users may exhibit different levels of abilities when interacting with an exploration system. Abilities, disabilities or impairments can be situational, temporary or permanent. For example, some users may require the possibility for one-handed interaction, which can be the result of a permanent loss of a limb or a mother carrying her child. Trouble reading text from a display showing information can be caused by bad contrast of the display when used outside, or by various temporary or permanent eye deficiencies. In any case, the exploration system should be able to support all users as best as possible. Due to the high variety of abilities, disabilities and impairments, many different interaction methods need to be combined in order to provide an effective exploration system for all users. For example, in addition to classic mouse and touch support, an exploration system could implement one-handed touch gestures in order to support one-handed interactions. However, many other scenarios are conceivable. These and similar challenges that aim at finding effective interaction methods between all kinds of digital computer systems and their users are the focus of a research area called *Human Computer Interaction* (HCI). An comprehensive introduction to HCI can be found in [book:6]. Recent work defines principles for so called *inclusive design*, which focuses on developing interactive systems suitable for all users [book:5].

Visualization and User Interface Design Challenges

Another sub-area of HCI deals with the challenge of information visualization. Since exploration systems are supposed to help users in learning about an information space, they need to provide suitable visualizations that convey meaningful information about it, its items and their relations. Depending on the type of information space and its properties, different kind of visualizations can be appropriate. For example, binary relations between items can be visualized as graphs, a connectivity matrix or simple list. Distance relations can be visualized as a similarity-based map or heat matrix. Items themselves can be visualized as dots, textual labels, pictograms or thumbnails. Unfortunately, it is unclear which visualization technique should be used at which point during an exploration process in order to convey useful information. Several general interface design patterns have been identified in the past, and may be applicable to exploration systems as well, e.g., the idea to implement a parallel view providing both overview and detail information, zoom interfaces, or focus and context interfaces [20].

Evaluation Challenges

Another important challenge is to evaluate the effectiveness of an exploration system, whether it actually supports users in exploring an information space. For example, when considering exploratory search and berrypicking behaviour, where users incrementally refine their own information need by studying search results, at some point, users stop issuing new queries. Unfortunately, there is no simple criteria to determine whether users successfully explored and satisfied their information need, or whether users simply failed to do so. Exploration is concerned with learning about an information space. Learning is an highly dynamic and user-specific process influence by prior knowledge and user abilities. Traditionally, learning progress is evaluated via progress tests, e.g., as applied in education. However, the learning process as part of an exploration session is considered highly sensitive, such that even the simple expectation of progression tests may have a significant influence on user behaviour. A common strategy to evaluate information systems in general, is to measure user satisfaction [36]. It is assumed that successful exploration will positively impact user attitudes towards the exploration system. Thus, measuring user satisfaction may allow to draw partial conclusions about the effectiveness of an exploration system. However, user satisfaction is influenced by many additional factors not directly related to exploration effectiveness, e.g., the overall usability of the interface, but also user characteristics like expectations or mood. White et al. [158] propose to evaluate exploration tools by conducting a comparative user study that indirectly measures the time required to learn through posing a task that can be solved more efficiently by users that simultaneously learn about the information space. It is assumed that, when comparing information systems that provide different levels of support for exploration, a significant variation in user performance can be measured. In [82], a news exploration system based on clustering analysis is evaluated by comparing multiple variations of the system. Multiple indicators are measured through questionnaires, interviews and interaction logs, e.g., the cognitive load of users, or the user performance when solving a retrieval task.

Algorithmic Challenges

Finally, there are a number of algorithmic challenges related to developing exploration systems. Information spaces may contain various kinds of data, from raw data to various media, documents, transactions, and more. Often, data consists only of low-level features, e.g., pixels or words, but users require high level information in order to learn about the information space, e.g., patterns or relations. Depending on the type of data, many different algorithms and methods are required to extract meaningful high-level information for users, e.g., statistics, image processing, object classification, document summarization, clustering, dimensionality reduction, and more. Since most methods are only approximations, they may introduce errors, which might lead to users being confused. Information spaces may contain many millions of objects and, at the same time, the system needs to be able to respond quickly to new user input. Thus, efficient algorithms are required that do not exceed certain computational performance limits.

In summary, due to the diverse challenges and requirements for exploration systems, it is considered highly unlikely that there exists a universal approach for effective exploration of all kinds of information spaces suitable for all kinds of users and scenarios. In the remainder of this thesis, exploration approaches are always considered in context of specific scenarios, e.g., exploration of a movie collection to find interesting movies to watch. In Chapter 3, a number of existing exploration approaches are discussed. Most of the previously mentioned challenges are revisited in later sections of this thesis. Chapter 8 discusses additional challenges related to map-based exploration approaches.

2.4 CHALLENGES IN HIGH-DIMENSIONAL SPACES

Information spaces often consist of highly complex objects and relations. When working with such detailed information spaces, various problems arise. The following two sections shortly introduce two of these problems.

2.4.1 *Curse of Dimensionality*

In order to find patterns and relations between objects of an information space, many algorithms require that data is represented as mathematical vectors. For simple information spaces, e.g., the space of all visible colours, a concise vector representation consisting of only a few components can be found. However, for rich information spaces, e.g., a collection of documents, each object, i.e., a document, consists of many information and, thus, can only be fully represented by a high-dimensional vector of many hundred or thousands of components.

Unfortunately, high-dimensional spaces exhibit properties that do not conform with human intuition developed from 3-dimensional physical spaces, and are cause for a variety of problems. In order to reference these problems the term *Curse of Dimensionality* was coined, see [book:2].

A simple example of a high-dimensional oddity can be found when considering the vastness of high-dimensional space. Lets imagine that a user would like to explore an n-dimensional cube of length ten by iteratively moving one unit at a time in a regular grid, otherwise known as grid sampling. For a 1-dimensional cube, i.e., a line, only 10 points have to be visited one after another. However, for a 2-dimensional cube, i.e., a square, already 100 points have to be iteratively observed. In case of an n-dimensional cube, a total of 10^n points needs to be visited. For $n = 100$, this corresponds to 10^{100} points, which is already much larger than estimated number atoms in the universe, which is believed to be no more than 10^{80}. Unfortunately, many information spaces easily exceed this number of dimensions, e.g., images, which consist of millions of dimensions or pixels.

Another example showing properties of high-dimensional space concerns the relative volume of a high-dimensional cube and ball. The

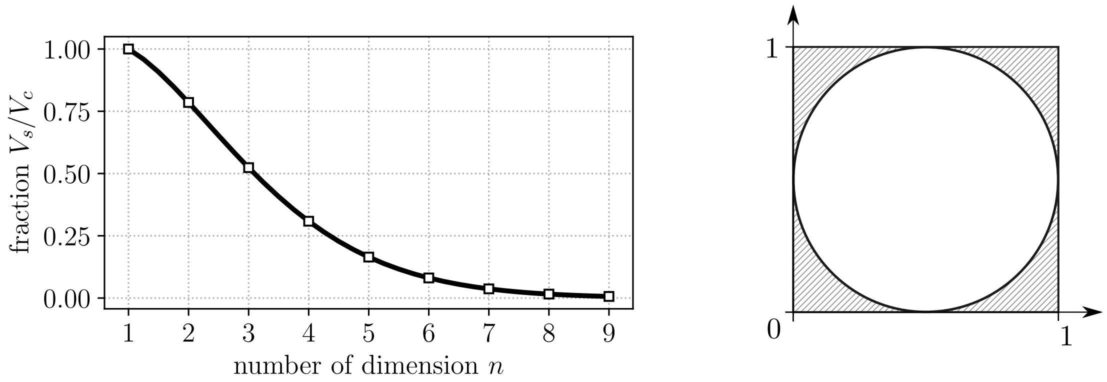


Figure 4: Fraction V_s/V_c of volumes of an n-dimensional unit balls and unit cube (left). A high-dimensional unit ball becomes increasingly small in relation to a cube of the same dimension. Two-dimensional unit cube and unit ball (right). The relative volume of the space near corners of the unit cube (marked with grey stripes) increases with higher dimensions in comparison to a unit ball.

volume of an n-dimensional unit cube is $V_c = 2^n$. The volume of an n-dimensional unit ball is

$$V_s = \frac{2\pi^{n/2}}{n\Gamma(n/2)}$$

whereas Γ references the Gamma function. Thus, the fraction of both volumes V_s/V_c is not constant with increasing dimensions. Figure 4 (left) illustrates this for up to nine dimensions. It shows that the relative volume of a unit ball and unit cube decreases with higher dimensions. As a consequence, the space near the 2^d corners of a high-dimensional unit cube, i.e., the volume of the unit cube that is not covered by its contained unit ball, takes up most of its space. An illustration of this corner space in two-dimensional space is given in Figure 4 (right).

A more relevant problem of high dimensional space was described by [8]. The authors show that with increasing dimensionality the relative difference of distances from a query point to its nearest and farthest neighbour converges to zero under common conditions. As a result, the authors question the meaningfulness of nearest neighbour search in high-dimensional spaces, since the relative contrast of distances vanishes with increasing dimensions. This especially applies to the euclidean distance or L^2-norm, but not all distance metrics in general. The authors of [1] show that for so called fractional L^p distance metrics with $0 < p < 1$, the relative contrast does not converge to zero. They experiment with multiple high-dimensional data sets and demonstrate that more nearest neighbours with correct class labels can be retrieved with fractional distance metrics in comparison to L^p metrics with $p \geqslant 1$.

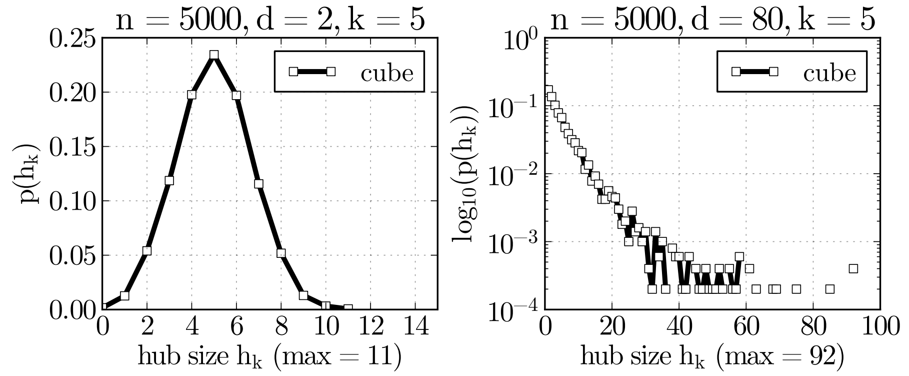


Figure 5: Distribution of the size of hubs for data uniformly distributed in a d-dimensional (hyper-)cube for $d = 2$ and $d = 80$ (right in logarithmic scale).

2.4.2 *Hubness Phenomenon*

Another curiosity that is often linked to the curse of dimensionality is called the *Hubness Phenomenon* [111, 112]. In high dimensional spaces, some points, called *hubs*, are observed to occur much more often than expected amongst the k-nearest neighbours of other points. For example, when considering uniformly distributed random points, one would expect that all points appear on average k-times amongst the k-nearest neighbours of other points. Instead, in high-dimensional space, this distribution is skewed to the right. In the following, the number of times a points occurs in the list of k-nearest neighbours is denoted by the *hubness* of a point, or sometimes also called the *size of a hub*. Hubs can be imagined to be similar to network hubs, which have many incoming nearest neighbour connections.

The Hubness Phenomenon has been shown to be present in many high-dimensional data sets, e.g., in a lot of data sets of the UCI Machine Learning Repository [112]. Large hubs can also be found in music collections [127], where a piece of music that is listed as most similar to many others is considered as a hub. In previous literature, the Hubness Phenomenon has only been observed in high-dimensional space, and thus, is often considered as a curse of dimensionality. In a later Section 10.3 or as published in [pub:3], hubs are demonstrated to exist in low-dimension spaces as well.

The first work that described the Hubness Phenomenon was published by Radovanovic et al. [111, 112]. They showed that the Hubness Phenomenon occurs in uniformly distributed data sampled from a high-dimensional (hyper-)cube. An illustration of the hubness distribution for low- and high-dimensional uniformly distributed random data can be found in Figure 5. For 2-dimensional data, the distribution of the number of times a point occurs in k-nearest neighbour lists follows as Gaussian distribution with mean $k = 5$. However, for 80-dimensional data, the distribution is significantly skewed to the right. A few points occur much more often amongst the k-nearest neighbours of other points. Radovanovic et al. [112] propose that points closer to the data mean tend to become hubs. In Section 10.3, the origin of hubs and their influence on k-nearest neighbour-based navigation is discussed further.

Mitigating the Hubness Phenomenon

In recent work, the hubness phenomenon is tried to be mitigated, e.g., by adapting the distance measure or nearest neighbour search. In [112] it is shown that fractional L^p distance metrics as proposed by [1] result in smaller hubs for $p = 0.5$ in comparison to $p = 2$. In [59], the distance between two points is computed based on the number or rank of mutual nearest neighbours deduced from a primary distance measure. These shared nearest-neighbour distances are shown to be more robust for high-dimensional data and distance-based tasks like clustering. Schnitzer et al. [127] propose an unsupervised method, called Mutual Proximity, which transforms distance matrices such that symmetric nearest neighbour relations are rewarded and asymmetric relations are penalized. Since nearest neighbour relations become more symmetric, fewer hubs occur.

However in general, it can be questioned whether the goal of removing hubs from the data is desirable at all, since hubs may induce the very structures that users seek to discover. In [139] a map-based user interface for exploring music collections is presented, which uses a multi-focus zoom in order to highlight high-dimensional nearest neighbours even if they are not projected close to each other. In this case, hubs are not visible in the two-dimensional projection, but can still be discovered when using the proposed multi-focus zoom.

In later sections of this thesis, e.g., Chapter 10, nearest neighbour search is suggested as a method for navigating information spaces. Therefore, various phenomena of the curse of dimensionality are relevant and need to be considered. However, the hubness phenomenon in particular introduces tangible limitations to the proposed exploration approach, and thus, is further discussed in Section 10.3.

3 EXPLORATION APPROACHES

In this chapter, various exploration approaches, tools and user interface designs are presented. Section 3.1 focuses on exploration tools based on graph visualizations. Section 3.2 discusses map-based exploration approaches. And Section 3.3 describes different interaction concepts for exploratory search.

Besides graph-based, map-based and exploratory search systems, there are many other approaches designed to support information exploration. Due to the large amount of research, only a small selection can be briefly mentioned. Early approaches include Waterworth [156], which proposes to make information space accessible by using the metaphor of virtual islands consisting of buildings of information. In [40], documents of an information space are suggested to be presented similar to articles in a newspaper. Wittenburg et al. [160] describe the concept of so called *bargrams*, which resemble histograms, i.e., they visualize the distribution of feature values, but at the same time allow to interactively filter objects of an information space.

Further approaches focus on collaborative information exploration. Isenberg et al. [61] propose a system that enables multiple users to interact with the same information space via different views on the surface of a touch-enabled tabletop. Other approaches suggest to combine multiple views in a single exploration interface. In [28] search results are shown as a sorted list together with additional widgets illustrating the result sets distribution with respect to time, geographic location and associated tags. Langner et al. [77] propose to combine multiple views on multiple devices, and align visualizations based on their physical arrangement on a table. In [42], documents can be filtered through a concept graph visualized in a list view, they can be clustered based on similarity in a cluster view, and inspected using word clouds in a document view.

Many more approaches exist. In the following three sections, special consideration is given to graph-based, map-based and approaches designed to support exploratory search.

3.1 GRAPH-BASED EXPLORATION SYSTEMS

There are many methods for visualizing graphs, e.g., using different layouts, highlighting clusters of nodes, and methods suitable for huge graphs. A survey of visualization methods can be found in [50]. While graph visualizations can convey important information about the structure of an information space, the visualization alone can not be considered an effective technique for information exploration. Instead, graph-based exploration systems support users in interactively learning about the overall graph structure, individual relations and their connected entities.

Exploring Local Graph Structure

In [63] a radial graph visualization is proposed, which allows to explore websites connected by hyperlinks. The graph structure is not visualized to its full extend, but a local sub-tree can be navigated iteratively through a focus and context approach. Each web page is presented as a thumbnail. The currently selected web page is shown in more detail in the centre. Navigation paths and linked web pages are arranged around the centre web page in a radial tree visualization. The user is enabled to interactively explore the hyperlink graph by selecting neighbouring web pages. In Chapter 10, the proposed map-based exploration approach also follows the focus and context strategy to support exploration.

Lohmann et al. [83] present an interactive graph exploration tool that allows to discover links between multiple entities of an ontology, e.g., DBpedia. Details of entities and related objects can be inspected in a side panel. Similar to [63], only the currently relevant sub-graph is visualized. However, both [63] and [83] do not integrate full-text search.

Exploratory Search and Graph Visualizations

Both Haun et al. [53] and Falke et al. [32] propose a graph visualization in context of exploratory search. In [53] search results are visualized by an extendible graph. Both results and queries are represented as connected nodes. Subsequent keyword queries are added to the same graph visualization, which all allows to discover novel relationships in case search results of different queries are connected through common neighbours. Additionally, clicking on a node will dynamically add new related neighbours, which allows to extend the graph depending on user interests. In Chapter 7, the same graph exploration framework is used for exploring web search results based on Formal Concept Analysis.

Recently, [32] suggest to combine a classic list-based search result visualization and a graph exploration interface. Results are connected if automatically extracted entities co-occur. However, the authors propose that more research is required to find suitable methods to extract graphs from documents or search results in order to support exploratory search and offer their tool as an evaluation system.

Multi-View Exploration

Other exploration systems combine graph visualizations with additional exploration components in multiple views. For example, Yimam et al. [163] propose an exploration system that allows to search documents via keywords, explore extracted entities in a graph visualization, view the geographic distribution of documents on a map, and filter documents based on a time-line histogram view.

3.2 MAP-BASED EXPLORATION APPROACHES

Maps have been proven to be an effective tool to convey spatial information for many centuries. Map-based exploration systems try to build on this foundation and utilize maps for the representation of digital information spaces. In the past, maps have been primarily used to visualize geographic information. However, with the emergence of advanced dimensional reduction methods, see Section 4.2, non-geographic information can be mapped as well.

Over the past years, many map-based exploration systems have been proposed. All approaches share the common strategy of representing similar information at nearby positions on the map. Users are enabled to explore the information space by zooming and panning the map. However, there are differences in how maps are structured, how they are visualized, and how users can interact with them.

Early Approaches using SOMs

Early approaches include map visualizations based on Self-Organizing Maps (SOMs) [71]. An introduction to SOMs can be found in Section 4.2. Kohonen [72] proposes to use SOMs to generate a similarity-based map of documents. Documents are represented as histograms of word categories, which were deduced from the full-text of the document. The resulting map can be explored by clicking on a cell of the map, which contains multiple similar documents. Regions of high density are visualized by white cells, and regions of lower density by darker cells. Search queries are possible by treating a query as a new document, projecting it to a cell of the map, and suggesting similar documents from the same or nearby cells. In contrast to later approaches, the map visualization of [72] does not provide a sense of orientation, i.e., there are no topic labels or representative items shown in the map. Users have to identify cluster semantics by inspecting individual cells and their documents.

Knees et al. [70] apply SOMs for exploring music collections. The map is generated by extracting Mel-frequency cepstral coefficients (MFCCs) [86] as audio features from 30 second segments in the middle of each song. Instead of using cell colours to represent data density as in [72], Knees et al. visualize the map as a landscape of islands separated by water in three dimensional space. Then, the height of the landscape illustrates data density, which amplifies the visual impression of separated clusters. In order for users to get a sense of orientation, for each cell of the map, song titles are shown and region labels were extracted, e.g., keywords describing the genre or mood. Individual songs can be discovered by flying through the 3d environment. Audio samples of the respective closest songs are played while moving through the landscape. Unfortunately, this strong analogue to physical environments and physical movement through a three-dimensional space can also be seen as limiting in comparison to two-dimensional visualizations, where map cells can be selected very efficiently via simple mouse or touch interactions.

Advanced Map Interactions

The Song Explorer proposed in [65] also uses SOMs for exploring music collections. Instead of a 3d landscape, songs are projected on the two-dimensional circular surface of a touch-enabled tabletop. The map can be explored by one- and two-handed touch gestures for selecting an item, as well as panning and zooming the map. Tangible objects can be placed on the surface of the tabletop to perform additional interactions, e.g., changing how map cells are coloured based on various audio features or showing textual information of a particular song. Such advanced interaction strategies are not subject of this thesis, but will be discussed as possible future work in Section 14.1.

Tailored Maps

In [147] music can be explored through a visualization that maps songs to a disc. Instead of learning a similarity-based map from the data, the authors suggest a fixed mapping based on the genre, artist and release year of each song. Genres and artists are represented as sections of the disc. The size of each section corresponds to the amount of songs that is available for that particular genre or artist. A song is represented by a coloured dot. The release year is encoded as the distance of a dot to the centre of the disc. The colouring scheme for songs can be changed according to user input. Users benefit from a comprehensive and easy to understand overview visualization for a music collection that both illustrates common music features and density information.

Personalized Map Generation

Highly complex information spaces such as music collections often consist of many diverse features that may or may not all be relevant to certain users in different scenarios. For example, a radio broadcaster that tries to find new songs may benefit from a different map structure than a private music enthusiast. Accordingly, map personalization approaches have been studied. Stober et al. allow to personalize maps of both music [139] and image collections [136] by adaptable feature weights. Users can interactively change the weighting for high-level feature facets, which starts an immediate recalculation of the map projection. Unfortunately, due to the high sensitivity of the employed multidimensional scaling projection algorithm, see Section 4.2, even small weight adjustments lead to structurally different maps. Thus, users can not easily transfer knowledge from maps of different weight configurations, and are required to invest additional time and effort to identify new clusters and patterns. Besides, the user interface provides a multi-focus zoom lens, which highlights high-dimensional neighbours even if they have not been projected close to each other on the map. As discussed later in Section 4.2, the effectiveness of how well neighbourhood relations are preserved by a projection algorithm can be used as an optimization goal. Thus, such an interactive multi-focus zoom lens represents an interesting alternative approach to this problem.

Reinert et al. [116] propose a method that allows users to manipulate a map visualization by moving individual items. The position of moved items on the map is then used to deduce the best matching 2d projection. However, in order to limit the search space, only axis-parallel projections are considered. Therefore, the problem is reduced to finding two features that best correlated to these user-specified item positions. They combine this approach with a novel layouting algorithm, which arranges images of various shapes and sizes without overlappings on the map. In Section 9.3, the problem of overlappings is discussed in more detail. In addition, Chapter 12 presents a prototype that follows the same idea of personalizing maps through direct manipulation.

Zooming through Relevance Feedback

Camargo et al. [15] propose a different approach for navigating large image collections through relevance feedback. Instead of generating a global similarity-based map, which would be computational infeasible for sufficiently large collections, only a representative sample of the collection is illustrated at all times. Users can incrementally refine the sampling or filtering mechanism by selecting a number of relevant images. Based on the relevance feedback, a refined set of images is retrieved and visualized. This dynamic filtering can be seen as a alternative approach to zooming a global map.

Multi-Stage Maps

Paul et al. [105] propose a hierarchical map visualization of documents. At the lowest zoom-level, i.e., the zoom level that contains all data, individual documents are not shown but represented by clusters. Each cluster is described by a set of keywords. When zooming into a cluster, more and more relevant keywords are revealed. Users can pin keywords as relevant, which boosts related keywords and makes them visible at lower zoom-levels. At the highest zoom-level, short document snippets are presented, which then can be clicked to retrieve the source document. Clusters are positioned via a principal component analysis (PCA) on the cluster centres. Keywords are positioned inside of a cluster based on a graph layouting algorithm. This two-step approach allows to minimize the number of items that have to be projected to a small number of clusters or visible keywords, which helps to reduce computational costs of map projection algorithms for large collections of documents.

Large-Scale Maps

In recent years, methods have been developed that further push the limit of how many items can be efficiently projected to a single map.

In [126], a collection of 100.000 songs is projected using the t-student distributed stochastic neighbour embedding (t-SNE) algorithm [149]. However, given reasonable hardware a mapping of a few million items is achievable. An introduction to the stochastic neighbour embedding (SNE) algorithm can be found in Section 4.2. Songs in [126] are visualized in a 3d environment as stacked blocks. Stacks are created by discretization of the projected coordinates. The height of a stack illustrates the data density similar to Kohonen et al. [72] and Knees et al. [70]. Popular songs are placed at the top of each stack. Users can navigate the environment by a virtual camera that allows zooming.

Mixing Map and Graph Visualizations

Lastly, map visualizations can be combined with other types of visualizations. This can be achieved either side by side, i.e., with multiple parallel views as described before in Section 3.1, or by combining them into a single view. Repke et al. [118] present a novel map projection algorithm, which combines map and graph visualizations by fusing optimization criteria such that similar documents are positioned close to each other and at the same time documents and their connected entities are positioned close to each other, e.g., document authors connected through co-authorship relations. The resulting map is overlayed with a graph visualization showing both documents, connected entities and their relations.

In this thesis, similarity-based map visualizations are further investigated for their effectiveness in supporting exploratory search in media collections like music, images or movies. An overview of various challenges for creating effective map-based exploration systems can be found in Chapter 8. A novel map-based exploration approach is proposed in Chapter 10.

3.3 EXPLORATORY SEARCH SYSTEMS

There are many different approaches to supporting exploration in the context of keyword-based search. An overview over various search user interfaces can be found in Chapter 4 of [159]. In the following, a selection of approaches are discussed in the context of this thesis.

Faceted Search

A popular extension to a simple list-based search result visualization is the introduction of search facets. Search facets can be viewed as either static or query-specific filters, which represent topics, categories or properties of the information space. They allow to interactively navigate the space of search results by applying individual filters or filter combinations. Facets are typically visualized in parallel to search results as a list or hierarchy of terms or concepts. Sometimes, facets are annotated with the number of search results that match a facet. Assuming an exploratory search scenario, users benefit from search facets in multiple ways. First, they allow to explore the result list more efficiently in comparison to a non-faceted list of results, which can help to get a better understanding of the information space. Second, if facets are generated dynamically from the current set of results, e.g., by extracting topics or categories, they can provide an overview over the set of search results. Lastly, if facets are annotated with match counts, they can give insights into the distribution of the set of search results.

Early research includes Hearst et al. [55], which propose a search system that allows to filter search results based on meta data structured in a hierarchy. Users can interactively combine keyword queries and filters on meta data. Since then, faceted search has been applied in many scenarios. Research has mostly focused on how to generate search facets for unstructured data, or how to find appropriate facets for complex information spaces, e.g., ontologies. In [141], methods are discussed to find facets suitable for end-users based on highly complex relations in ontologies. Capra et al. [16] evaluate search facets in the context of exploratory search by comparing user interface variations, e.g., facets visualized as a list in comparison to a word cloud. Hoque et al. [58] propose a faceted image search and exploration system. Image results can be filtered based on a concept hierarchy deduced from Wikipedia. In [24], hierarchical facets are extracted from documents by a Formal Concept Analysis (FCA). An introduction to FCA can be found in Section 4.1.

Search Result Space Visualization

Another way to support exploration during search is to provide advanced visualizations of the set of search results. In [99], the set of results is visualized as a two-dimensional embedding, called the VIBE display. In contrast to map-based exploration systems as described in Section 3.2, documents are not arranged in a way that reflects their pairwise similarity, but according to their respective relevance to individual query terms. Users are enabled to find documents that, e.g., are only relevant to a subset of query terms, or are particular relevant to one term, but less relevant to others. In comparison to a list visualization, where term relevance is combined to a single rank, the VIBE display illustrates multidimensional term relevance in a two-dimensional linear embedding. Singh et al. [131] build on the idea of a newspaper visualization as proposed by [40]. They suggest to combine query term relevance and measures of topical and temporal diversity in order to arrange search results similar to articles in a newspaper, which is supposed to facilitate exploration. In [140], document results of a search are grouped based on co-occurring terms and visualized as a graph. The graph illustrates a set of user-selected terms and their associated documents. Each node contains the set of documents for a particular term or set of co-occurring terms. Nodes are connected based on their term subset relation. Users benefit from an overview visualization of the set of search results with respect to their topics and topic overlappings.

Query Recommendations

Since the information need of users evolves during an exploratory search session, approaches have been developed to support users by modelling this continuous process. The authors of [52] analysed search sessions of a state-of-the-art search system in order to derive a graph-based task model for suggesting queries tailored to exploratory search tasks. In contrast to traditional query recommendation, which focuses on refining queries by suggesting common co-occurring query terms, the approach described in [52] suggests related queries that are rated to be novel and diverse given previous queries in the current search session. Users are supported by receiving more relevant query suggestions for typical exploratory search tasks previously experienced by other users, e.g., vacation planning.

Personalized Ranking

In [2], the authors suggest to re-rank search results based on a short-term task model to support users during exploratory search. Following the berrypicking model, the search system allows users to collect notes during subsequent queries in a search session. Simultaneously, a task model is deduced, which consists of weighted terms extracted from these notes. Then, search results can be re-ranked by mixing query relevance and task model relevance for scoring documents. Users benefit from this re-ranking by finding documents that are more relevant to their evolving information need. However, as the authors point out, the proposed method does not consider the aspect of information novelty and favours retrieving redundant documents containing information that has already been noted.

Query Modelling

The Quest system proposed by Nitsche et al. [96] allows to visually model a user's information need by moving individual query terms in a radial visualization. Users may add multiple query terms to a radial layout. Search results for the combined query of all active terms are shown in a separate panel as a traditional list. Depending on the position of each query term in the radial layout, terms are weighted higher or lower, which influences the ranking of search results. Users can explore the space of search results by interactively adjusting term weights and observing changes to the result list. Terms that are not relevant to the current information need can be stored outside of the radial layout and reused later. In addition, search results are embedded in the radial layout in a similar way as done by the VIBE display [99], i.e., results that are more relevant to individual terms are positioned close to these query terms. Ruotsalo et al. [124] suggest a similar radial visualization as the Quest system, consisting of an inner circle of currently active query terms and outer circle containing search results and additional keyword recommendations. In [19], queries are modelled as a combination of semantic groups of keywords called *lenses*. Each lens contains one or multiple keywords describing one aspect of the user's information need. Lenses and keywords have to be defined by a user manually, which may be perceived as an initial burden. Additionally, keywords can be weighted and are visualized in coloured boxes of different size representing their individual weight. Users benefit from separate visualizations that illustrate the relevance of search results with respect to each lens and keyword.

Supporting Multiple Search Streams

Since exploratory search sessions may consist of multiple subsequent queries, there are approaches that directly represent multiple search streams. Nitsche et al. [97] present a user interface prototype that visualizes multiple search result lists. The user starts with an initial query. Search results are illustrated as thumbnails and arranged in a horizontal list that can be scrolled to the right of the screen. Additional search streams can be added by clicking on a search result, which will add new results in a vertical list for automatically extracted keywords. Subsequent result lists can be branched out and will be visualized in alternating horizontal and vertical direction. Overlapping results are stacked. Over multiple interactions, a tree-like visualization of multiple search streams is generated. A disadvantage of this approach is that individual results can only be represented by small thumbnails. Thus, the relevance of results can not easily be judged by users without opening each web page. Klouche et al. [69] follows a more textual representation of search streams by showing both extracted keywords, extracted entities and document titles in a vertical result list. User can drag keywords and entities to create new search streams. Relevant documents can be saved in a separate readings list. Users are supported by having access to past search result lists in a single view, which may help to get a sense of their own evolving information need and formulate new queries.

Search History Visualization

A more detailed representation of the search history is proposed in [41]. Past queries are related to the currently active search results by visualizing the their relevance with respect to prior queries in a histogram. This allows to discover results that are novel in comparison to prior queries and prior result lists, or discover results that have been consistently relevant over multiple queries. Additionally, a separate history view of past queries provides an overview over all search results, whether they have been already visited, marked as useful or marked as irrelevant.

This thesis investigates several approaches to supporting exploratory search tasks. In Chapter 6, exploratory search is supported by annotating search results with entities of an extendible ontology, such that users can more easily judge the relevance of each result. Chapter 7 proposes to structure search results by applying Formal Concept Analysis (FCA) and visualizing the resulting concept lattice. In Chapter 10, exploratory search is supported using a novel interaction approach that visualizes search results inside a similarity-based map and allows to explore neighbouring information objects through nearest neighbour search.

4 STATE-OF-THE-ART ALGORITHMS

Designing exploration systems involves many state-of-the-art technologies and algorithms. Information may need to be analysed to extract relevant concepts, patterns or clusters, involving methods from research areas like Natural Language Processing, Image Processing, Machine Learning or Formal Concept Analysis. Information may need to be indexed in order to support efficient retrieval through different approaches like inverted indices or fast nearest neighbour search. And information may need to be presented by novel visualization methods involving methods from research areas like Graph Theory, Dimensionality Reduction or Artificial Neural Networks. In this chapter, a selection of relevant algorithms of various research areas are shortly introduced. Section 4.1 provides an introduction to Formal Concept Analysis. Section 4.2 describes the approach of dimensionality reduction, including relevant algorithms that will be discussed later throughout the thesis. Section 4.3 shortly introduces the method of Procrustes analysis. In Section 4.4, the topic of nearest neighbour search is outlined. Finally, Section 4.5 introduces the concept of metric learning.

4.1 FORMAL CONCEPT ANALYSIS

Formal Concept Analysis (FCA) describes an approach to derive a hierarchy of concepts from a set of objects and their attributes. It can be used in a wide variety of scenarios, e.g., as a tool for generating and visualizing concept hierarchies for exploratory literature analysis [107], to support matching of biomedical ontologies [164], or to provide hierarchical categories for a faceted search system [24]. A comprehensive introduction to Formal Concept Analysis can be found in [book:7]. Formal concepts are sets of objects and their corresponding shared attributes. For example, the formal concept *bird* would include all bird species and their common properties, e.g., species like hummingbird and starling, and properties like "have feathers" and "lay eggs". However, the property of flying would not be part of the concept, since some birds, e.g., penguins, can not fly. Similarly, there can be more specific and more general concepts, e.g., the concept of ducks or animals, respectively.

Formal Context

Formal concepts are derived from a formal context $\mathbb{C} = (\mathbb{O}, \mathbb{A}, \mathbb{I})$, which is a triple of the sets of objects $\mathbb{O}$, attributes $\mathbb{A}$, and, in its most simple form, a binary relation between objects and attributes $\mathbb{I} \subseteq \mathbb{O} \times \mathbb{A}$. In many applications, higher-order relations are needed, e.g., attributes with multiple values, or even fuzzy attributes, see [14]. Table 1 illustrates the formal context for a simple example consisting of seven hypothetical documents and five topics. A document and topic is linked by the binary relation $\mathbb{I}$, if the corresponding row and column in Table 1 is marked with an x, which symbolizes that a document covers that particular topic.

Derivation Operators

In order to deduce formal concepts, two derivation operators need to be defined, which describe the set of shared attributes given a subset of objects, and the set of shared objects given a subset of attributes. Let $O \subseteq \mathbb{O}$ be a subset of objects $\mathbb{O}$, than O^* is defined as the set of attributes that is shared amongst all objects of O:

$$O^* = \{a \in \mathbb{A} \mid \forall o \in O : (o, a) \in \mathbb{I}\}$$

Similarly, let $A \subseteq \mathbb{A}$ be a subset of attributes $\mathbb{A}$, than A^* is defined as the set of objects that is shared amongst all attributes of A:

$$A^* = \{o \in \mathbb{O} \mid \forall a \in A : (o, a) \in \mathbb{I}\}$$

Both derivation operators are inversely connected through $\mathbb{I}$, such that certain properties hold, called *Galois connection*. In particular, both derivation operators will only extend the subset of objects O or the subset of attributes A when applying them twice, i.e., $O \subseteq O^{**}$ and $A \subseteq A^{**}$.

Formal Concept

Then, a formal concept is defined as a maximal subset of objects and attributes $(O, A) \subseteq \mathbb{O} \times \mathbb{A}$ that can not be extended through derivation operators, i.e., $O^* = A$ and $A^* = O$. In other words, a subset of objects and attributes is considered maximal, and thus, a formal concept, if there are no other objects that share the same attributes of objects in O, and there are no additional attributes not yet included in A that are common to all objects in O. For a formal concept (O, A), the set of objects O is called *extend*, the set of attributes is called the *intent*.

Going back to the example shown in Table 1, the pair of subsets ({document 3, document 5}, {science, health}) defines a formal concept, since there are no other documents that share both attributes, and document 3 and 5 do not have any other topics both in common. In total, eight formal concepts can be deduced from the example in Table 1.

Concept Hierarchy

In order to structure and visualize these formal concepts in a hierarchy, first a notion of a sub-concept relationship needs to be defined. A formal concept $C_1 = (O_1, A_1)$ is a sub-concept of another formal concept $C_2 = (O_2, A_2)$, if and only if $A_1 \subseteq A_2$ or, equivalently, if $B_2 \subseteq B1$. This follows the intuitive definition as discussed before for natural concepts, e.g., animal species, i.e., ducks being a sub-concept of birds, all ducks are birds, but not all birds are ducks, and ducks sharing more common properties than birds in general.

Concept Lattice

	art	economy	politics	science	health
document 1			x	x	
document 2	x				
document 3		x		x	x
document 4	x		x		
document 5				x	x
document 6		x		x	
document 7				x	

Table 1: Binary object to attribute relationship describing a formal context. Rows correspond to documents, columns correspond to topics. An x marks the fact that a document covers a certain topic.

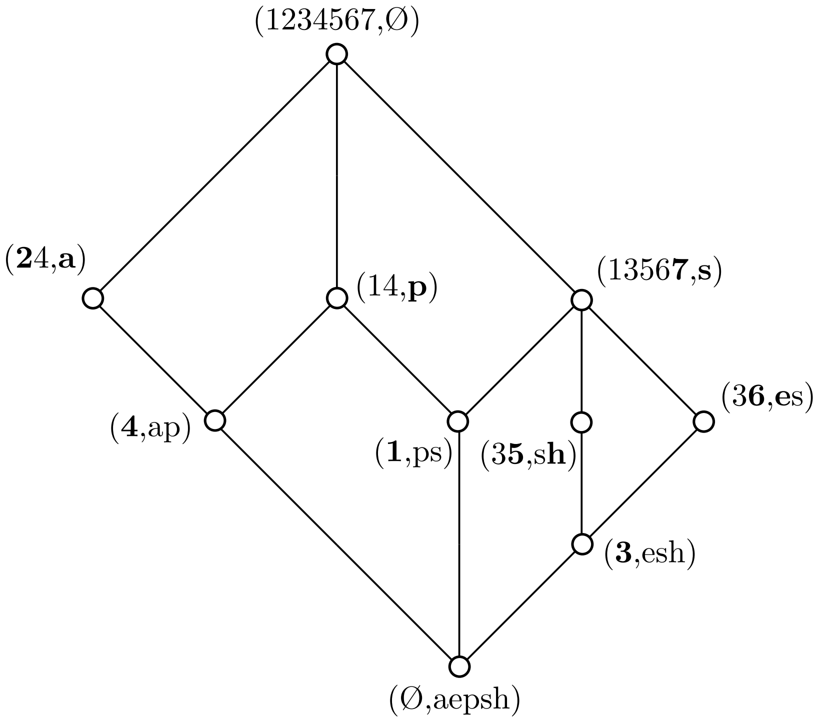


Figure 6: Lattice diagram of the formal concept hierarchy derived from the formal context defined in Table 1. Each circle or node represents a formal concept, digits reference documents, and letters reference topics.

Given this sub-concept relationship, a hierarchy of formal concepts can be deduced. Figure 6 illustrates this hierarchy as a concept lattice based on the formal context of Table 1. Each node or circle corresponds to a formal concept. The objects and attributes of a concept are represented by digits and letters, referencing the corresponding document and topic. A descending line between two nodes depicts a sub-concept relationship. From top to bottom, nodes will represent more specialized concepts that contain fewer objects, but share more attributes. Attributes that are added for a sub-concept, and objects that are not included in any further sub-concepts are marked in bold. Unfortunately, visualizing large concept hierarchies as a lattice diagram can quickly become overwhelming.

In Chapter 7, search results are analysed for formal concepts and visualized as a lattice using a graph exploration toolkit. It is hypothesized that users would benefit from a structured visualization of search results in comparison to simple lists.

4.2 DIMENSIONALITY REDUCTION

The purpose of dimensionality reduction is to find a projection that maps data points from a high-dimensional representation to a lower-dimensional representation while preserving certain relevant information of the original data space. In this thesis, dimensionality reduction is used to map data to a 2-dimensional space, which is visualized to the user for the purpose of search and exploration. Apart from that, it is often used as a pre-processing step aiming to reduce the feature space by removing irrelevant information, e.g., noise, or compressing redundant information, e.g., when features are highly correlated. Further analysis of the lower dimensional data, e.g., when applying classification or clustering algorithms, often yields better results than applying the same algorithm to the data in original space. Dimensionality reduction methods are widely applied in the area of data analysis, including pattern recognition in Bioinformatics, or Latent Semantic Indexing for document retrieval, amongst others.

There are many methods for dimensionality reduction: classic principle component analysis (PCA) [book:9], multidimensional scaling (MDS) [75], local linear embedding (LLE) [122], Isomap [144], and many more. Methods can be divided into linear and non-linear approaches. A detailed overview can be found in [151]. A narrow selection of them is discussed in further detail in Section 4.2.2 and 4.2.5.

Formally, a projection p can be defined as a simple mapping $p : \mathbb{R}^d \mapsto \mathbb{R}^l$, which describes an arbitrary transformation of d-dimensional points to an l-dimensional space with $l < d$. In this thesis, l usually equals 2, since the aim is to provide a 2-dimensional visualization to users.

4.2.1 *Optimization Criteria*

The goal of preserving information can be expressed by various optimization criteria, e.g., maximizing the variance of the projected points as in PCA, or minimizing a reconstruction error as done in autoencoders. Unfortunately, there is no universal optimization criteria. Depending on the task, different criteria are more suitable.

In many cases, not all relevant information can be preserved. In the following, the information that is lost due to projection p is considered as projection error. However, depending on the application, there are different aspects of information that is aimed to be preserved, which results in different approaches for formalizing a projection error.

Reconstruction Error

In case of a reconstruction hypothesis, the projection error can be viewed as the error that is made when reconstructing a high-dimensional representation from a projected point. The corresponding reconstruction projection can be defined a mapping $p^r : \mathbb{R}^l \mapsto \mathbb{R}^d$. The reconstruction error e^r is then calculated as

$$e^r = \sum_{x \in X} \|x - p^r(p(x))\|$$

for $X \subseteq \mathbb{R}^d$. In case of PCA, finding a linear embedding that maximizes variances can also be interpreted as finding a linear embedding that minimizes the reconstructing error under the euclidean norm [145]. In case of autoencoders, the neural network is specifically designed to model both a projection and reconstruction mapping, which is then optimized for a minimal error.

Pairwise Distance Error

Instead of focusing on object features, the projection error can also be modelled as a measure of how well pairwise relations between objects are preserved by the projection. A common strategy aims at measuring the difference in pairwise distances between objects in the original space compared to objects in the projected space.

Given a matrix of pairwise distances or dissimilarities $D \in \mathbb{R}^{n \times n}$ and its components $d_{i,j}$ in original space, a matrix $\hat{D}$ of pairwise distances and its components $\hat{d}_{i,j}$ in projected space, the projection error e^d can be defined as the sum of squared differences between both matrices:

$$e^d = \sum_{i,j} (d_{i,j} - \hat{d}_{i,j})^2$$

In Section 4.2.2 below, a project algorithm called multidimensional scaling (MDS) is presented, which is based on this error hypothesis.

Neighbourhood-Based Error

Alternatively, the projection error can be viewed as the ability to preserve neighbourhood relations. Of course, considering only local neighbourhood relations will not preserve global structure and neglect pairwise distance relations. In [67] the measures of trustworthiness and continuity are introduced. Intuitively, trustworthiness and continuity describe how well neighbourhood relations are preserved after a projection. The duality between trustworthiness and continuity roughly correspond to the duality of false positive and false negative error rates in a two class classification problem.

In the following, k-nearest neighbours of a data point $x_i \in X$ with respect to a distance function δ are denoted by $N_{\delta,k}(x_i)$. Nearest neighbours of x_i in the original d-dimensional space are referred to as $N_{\theta,k}(x_i)$, with θ being a distance function in the original space. Nearest neighbours of x_i in the projected space are referred to as $N_{\rho,k}(x_i)$, with ρ being a distance function in the projected space. Furthermore, the rank of point x_i in the sequence of all points $(x_1, \dots, x_n)$ ordered by distance θ from x_j in the original d-dimensional space is called $R^o_{x_j}(x_i)$. Equivalently, the projected rank $R^p_{x_j}(x_i)$ is considered the rank of a point x_i in the sequence of all points ordered by distance ρ from x_j in the projected space.

False positive nearest neighbours, meaning, neighbours in the projected space that are not neighbours in the original space, are denoted by:

$$N_k^{fp}(x_i) = N_{\rho,k}(x_i) \setminus N_{\theta,k}(x_i)$$

False negative nearest neighbours, meaning, neighbours in the original space that are not neighbours in the projected space, are denoted by:

$$N_k^{fn}(x_i) = N_{\theta,k}(x_i) \setminus N_{\rho,k}(x_i)$$

Trustworthiness is a measure of the nearest neighbour rank in original space evaluated for false positive nearest neighbours. It is defined as

$$\text{trustworthiness} = 1 - C(k) \sum_{x_i \in X} \sum_{x_j \in N_k^{fp}(x_i)} (R^\theta_{x_i}(x_j) - k)$$

with $C(k) = 2(nk(2n - 3k - 1))^{-1}$ being a constant for normalization to the range of $[0, 1]$. Trustworthiness is maximal if all neighbours in the projected space are also neighbours in the original space. If there is a neighbour in the projected space, which is not neighbour in the original space, the measure of trustworthiness is reduced relative to the rank of that neighbour in original space being higher than k.

Correspondingly, continuity is a measure of the nearest neighbour rank in projected space evaluated for false negative nearest neighbours. It is defined as:

$$\text{continuity} = 1 - C(k) \sum_{x_i \in X} \sum_{x_j \in N_k^{fn}(x_i)} (R^\rho_{x_i}(x_j) - k)$$

In Section 9.2, the measures of trustworthiness and continuity are used to evaluate several projection algorithms when new items are being added to the data collection.

4.2.2 *Multidimensional Scaling*

Multidimensional Scaling (MDS) follows a non-linear approach and aims at finding a lower-dimensional embedding that preserves pairwise dissimilarities between all data points. It only requires a matrix $D \in \mathbb{R}^{n \times n}$ of pairwise dissimilarities as input, as well as the dimensionality l of the lower-dimensional embedding space. The resulting embedding is a matrix $X \in \mathbb{R}^{n \times l}$ of l-dimensional vectors whose pairwise distances matches D as best as possible. A comprehensive introduction can be found in [book:4].

Classical MDS

In the past, multiple approaches to MDS have been developed. Classical MDS as introduced by Torgerson [146] and Gower [44] provides a closed form solution assuming that pairwise dissimilarities are distances. Given a matrix of dissimilarities D, an l-dimensional embedding can be obtained by the following procedure [book:4]:

1. Calculate the matrix of squared dissimilarities D^2.
2. Apply the so called *double centring* to D^2:
$$B = -\frac{1}{2} J D^2 J$$
with
$$J = \mathbb{I} - \frac{1}{n} \mathbb{1}\mathbb{1}^T$$
and $\mathbb{I}$ being the identity matrix of size n, $\mathbb{1}$ being an $n \times 1$ vector of ones, and n the number of rows or columns of matrix D.
3. Calculate the eigenvalue decomposition of B to $Q\Delta Q^T$.
4. Consider the first l largest eigenvalues of Δ as Δ_l and the first l columns of Q as Q_l.
5. Then, the embedding is obtained by $Q_l \Delta_l^{1/2}$.

The resulting pairwise distances of the embedding will match D perfectly, if D contains euclidean distances with intrinsic dimension of at most l. Otherwise, the embedding will exhibit a projection error and only represent an approximate solution. The computational complexity of classical MDS is bound by the computational complexity of eigenvalue decomposition, which can be calculated in $\mathcal{O}(n^3)$. Thus, dimensionality reduction of very large data sets is computationally infeasible. Another downside of this approach is that classical MDS is sensitive to changes in the data, which will at least lead to solutions that are arbitrarily rotated or reflected, and in the worst case lead to embeddings that are structurally different. Also, classical MDS can not be incrementally refined in case more data is available. In practice, both problems can be mitigated by applying Procrustes analysis, see Section 4.3, which can be used to transform the embedding by the means of translation, rotation, scaling and reflection in order to approximate a previous solution.

MDS variants

Other MDS variants are based on iteratively optimizing a stress function, see Metric-MDS in [book:4], or do only require ordinal relationships between pairwise objects, see Non-Metric MDS in [book:4]. The problem of high computational complexity has been approached by applying MDS to a representative subset only, called landmarks, and projecting the remaining objects as a linear combination of said landmarks [25]. Additionally, multiple MDS variants have been adapted to utilize Graphics Processing Units (GPUs) in order to accelerate the iterative optimization process [100], e.g., a fast sampling-based variant, which first calculates classical MDS for small subsets of the dataset and then aligns each map by applying classical MDS again on a sample of each subset [162, 101].

In Section 9.2, classic MDS in combination with Procrustes analysis is proven to be an effective projection method with advantageous properties for exploring a growing music collection. In later Sections 10.2, 10.4 and following, classic MDS is used to generated a map-based projection of a subset of objects in order to provide an informative visualization to users in the scenario of exploration and exploratory search.

4.2.3 *Stochastic Neighbour Embedding*

Another non-linear projection method is called Stochastic Neighbour Embedding (SNE) [56]. In contrast to multidimensional scaling, SNE is based on neighbourhood relations and finds a low-dimensional embedding that preserves local neighbourhoods.

Since neighbourhood relations are discrete, i.e., a point is either amongst the k-nearest neighbours of another point or it is not, they can not easily be used as part of a continuous optimization problem. Therefore, SNE models neighbourhood relations as conditional probabilities. For each pair of points x_i and x_j, a conditional probability $p_{j|i}$ is estimated from the dissimilarity of both points. It represents the chance that point x_i picks point x_j as neighbour. Following [150], the conditional probability $p_{j|i}$ is estimated in accordance to a Gaussian distribution centred at x_i with variance of σ_i^2 and calculated as:

$$p_{j|i} = \frac{\exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||x_i - x_k||^2/2\sigma_i^2)}$$

Points close to x_i are assigned a high probability, and points far away are assigned a low probability. The variance σ_i^2 influences the distribution of probabilities $p_{j|i}$ between near and far neighbours. For high values σ_i^2, far neighbours are assigned a higher probability and, thus, will have a higher influence on the embedding. Since the notion of near and far is data dependent, a parameter k called perplexity is introduced, which allows to normalize σ_i^2 for each data point through a binary search. The normalization ensures that the probability mass is distributed independent of data density and covers approximately k neighbours. Higher values of perplexity will lead to larger neighbourhoods to be considered, which preserves larger sub-structures in the embedding.

The optimization process than tries to find embedded points y_i that exhibit the same neighbourhood probabilities $q_{j|i}$ in low-dimensional space. This is achieved via a gradient descend optimization on the Kullback-Leibler divergence between both high-dimensional ($p_{j|i}$) and low-dimensional neighbourhood probabilities ($q_{j|i}$). The cost function C is defined as:

$$C_{\text{SNE}} = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$$

Starting with random coordinates, the embedding is improved incrementally via gradient descent.

Comparison to MDS

In contrast to MDS, when using reasonably small perplexity values, only local relations are preserved. In some scenarios, this can lead to confusing embeddings. For example, when projecting differently sized high-dimensional Gaussian-distributed clusters, the embedding illustrates clusters as similarly sized, since data density is normalized based on the perplexity parameter. Similarly, distances between cluster can be misrepresented. In comparison to classical MDS, the approach requires hyper-parameter optimization of both perplexity and learning rate, which can have a significant impact on the embedding result. Also, due to the incremental optimization, a stop criteria needs to be specified. In addition, due to the random initialization of coordinates, the approach is non-deterministic and will yield different embeddings for every trial.

In Section 9.2, SNE is compared to other dimensionality reduction methods in terms of its projection quality and projection stability when dealing with growing music collections.

t-SNE variation

In [150], a now popular variant of SNE was developed, which models low-dimensional neighbourhood probabilities via student-t instead of Gaussian distributions, and thus, is called t-SNE. It benefits from a symmetric cost function, which can be optimized easier, and avoids the so called crowding problem by using the student-t distribution, which results in improved embeddings [150]. In recent years, Van Der Maaten [149] further accelerated the optimization process, such that large embeddings of many million items can be calculated in reasonable time on modern hardware. Current exploration systems, e.g., [118, 126], utilize this advancement by generating large scale maps for music and document exploration. In this thesis, t-SNE is not considered, since at the time of developing the map-based exploration approach discussed in Chapter 10, the method suffered from high computational costs and required additional hyper-parameter optimization. A detailed discussion is presented in Section 9.1 and 9.2.

4.2.4 *Neighbour Retrieval Visualizer*

The Neighbour Retrieval Visualizer (NeRV) [152] extends the idea of stochastic neighbour embedding (SNE) from an information retrieval perspective. In NeRV, the process of dimensionality reduction is viewed as a retrieval task of relevant neighbours from the high-dimensional space [152]. Accordingly, both precision and recall should be considered. Precision describes the fraction of high-dimensional neighbours also being neighbours to the same point in a low-dimensional embedding. If a high-dimensional neighbour is not visualized as a neighbour in the embedding, precision is reduced. Recall describes the fraction of neighbours in the embedding also being neighbours to the same point in high-dimensional space. If a point is visualized as neighbour, even though it is actually not a neighbour in the original high-dimensional space, recall is reduced. As described in [152], due to the one-sided cost function of SNE via the Kullback-Leibler divergence, SNE can be interpreted as optimizing for a smoothed version of recall only. NeRV extends the cost function of SNE to also optimize for precision. A factor λ is introduced that balances both recall and precision. Following [152], the cost function of NeRV is defined as:

$$C_{\mathrm{NeRV}} = \lambda \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} + (1-\lambda) \sum_i \sum_j q_{j|i} \log \frac{q_{j|i}}{p_{j|i}}$$

When setting $\lambda = 1$, the cost function C_{SNE} remains. The optimization of C_{NeRV} is achieved through a conjugate gradient algorithm. Additionally, the authors suggest to start the optimization process with a large σ_i^2 and iteratively decrease it in order to avoid local optima. Similar to t-SNE, a variant of NeRV called t-NeRV was deduced in [153]. In Section 9.2, NeRV and SNE are compared in the context of visualizing a growing music collection.

4.2.5 *Autoencoder*

The concept of Autoencoder [57] refers to a specific network design architecture of artificial neural networks. A comprehensive introduction to the fundamentals of artificial neural networks can be found in [book:3]. An Autoencoder network implements dimensionality reduction by artificially forcing information through a bottleneck. For this purpose, it uses a reconstruction hypothesis, i.e., information in form of an input vector is first compressed to the dimensionality

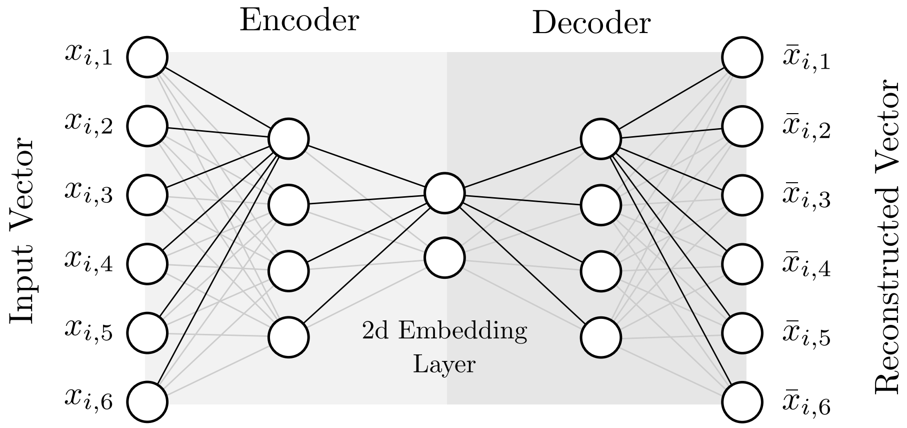


Figure 7: Fully-connected autoencoder network architecture for dimensionality reduction of six-dimensional data to a two-dimensional embedding using encoder and decoder sub-networks.

of the bottleneck, and then reconstructed from the low-dimensional embedding. In terms of an artificial neural network, this information bottleneck is achieved through a network architecture consisting of an encoder and decoder sub-network with gradually decreasing and increasing numbers of neurons per layer, see Figure 7. Optimization of the network weights is done by minimizing a reconstruction error using gradient descend. A common error function is the mean squared error between the original input vector x_i and the reconstructed input vector $\bar{x}_i$:.

$$C_{ae} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x}_i)^2$$

Autoencoders are applied in many research areas. They are used as a general dimensionality reduction method, e.g., as a preprocessing step for classification or clustering. Autoencoders have been applied to generate a semantic hash encoding [125], which allows to efficiently retrieve, e.g., similar images [74]. And autoencoders are used for data visualization [18].

Autoencoder Variations

In recent years, many variations of the general autoencoder network design have been developed. Unmodified autoencoder networks suffer from the problem that there is no regularization regarding the embedding. In the worst case, given sufficient layers and neurons, input vectors could be compressed and reconstructed in a way that the embedding does not represent any semantic meaning. Different approaches for regularization exist, e.g., to favour sparse encodings, see [113]. Another approach is called Variational Autoencoders [68], which introduces a stochastic sampling process into the encoding layer in order to avoid overspecialisation of the embedding. Input vectors are encoded to the parameters of a probability distribution, usually a Gaussian. Then, a sample is drawn and used to reconstruct the input through the decoder network. An in-depth discussion of autoencoder networks and its variations can be found in Chapter 14 of [book:8].

In Chapter 12, a variational autoencoder is used for generating similarity-based maps for information exploration. Additionally, the training process is modified to support integrating user feedback for personalization.

4.2.6 *Growing Self-Organizing Maps*

Self-Organizing Maps (SOMs) [71] are a different type of neural network inspired by how sensory information is processed in the human brain. In contrast to classic multi-layer networks like autoencoders, there are no layers nor activation functions. Neurons are arranged in a topology, usually a 2d grid, which represents a global map. Each neuron corresponds to a sub-area of the global map and is often visualized as a hexagon. Input vectors are embedded by assigning them to one of the neurons. Therefore, input vectors are not embedded as precise continuous coordinates. The assignment is designed as a winner takes all competition between neurons. The neuron whose weight vector is most similar to the input vector wins. Both the winner neuron and neighbouring neurons are adapted via gradient descent to better match the given input. The adaptation of neighbouring neurons is dampened by distance to the winner neuron. Over time, both the learning rate and neighbourhood radius is reduced. As a result, neurons will slowly adapt to fully cover the high-dimensional data.

Growing Maps

Self-organizing maps have been used in a number of visualization tools, e.g., for music exploration in [156, 70, 137]. Unfortunately, often it is unclear how many neurons are required to represent a data set in sufficient detail. Therefore, several growing strategies were developed. All follow a common idea. The learning process starts with a small map of few neurons. A subset of neurons, e.g., boundary neurons, are evaluated for their total error. The total error describes the sum of difference between the neuron's weight vector and input vectors that are assigned to that neuron. If the total error of a boundary neuron exceeds a threshold, one or multiple new neurons are added.

In [34], a grid of neurons is extended by adding full rows or columns of neurons, such that a rectangular shape is preserved, which can be beneficial for visualization. In [137], single neurons are added to the boundary, which can lead do a non-rectangular structure of neurons. In Section 9.2, the same growing strategy is applied. The resulting map is compared with embeddings generated by multidimensional scaling, stochastic neighbour embedding and the neighbour retrieval visualizer.

4.3 PROCRUSTES ANALYSIS

The concept of Procrustes analysis references a mathematical problem described by Gower [43]. The goal is to transform a set of points by translation, rotation, reflection and scaling in order to minimize the squared differences to another configuration of these points. Let us assume as set of n entities and two sets of points $X, L \subseteq \mathbb{R}^{n\times d}$, which are different configurations of the same entities, i.e., their points may have different coordinates, but both x_i and l_i reference the same i-th entity. Then, Procrustes Analysis finds a transformation of points X to $\bar{X}$ by translation, rotation, reflection and scaling, which minimizes the squared differences between all pairs of points of $\bar{X}$ and landmarks L.

Following [121], the transformation can be calculated by this procedure:

1. Translate both sets such that their mean is at the origin
$$L_o = OL\,, \quad X_o = OX \quad \text{for} \quad O = \mathbb{I} - \mathbb{1}/n$$
with $\mathbb{I}$ being the identity matrix and $\mathbb{1}$ being a matrix of all one.
2. Remember both translation vectors:
$$l_t = l_i^o - l_i\,, \quad x_t = x_i^o - x_i \quad \text{for} \quad l_i^o \in L_o\,, \quad x_i^o \in X_o$$
3. Determine the scaling factor for both L_o and X_o such that the average squared norm of each point equals n
$$l_s = \sqrt{\text{trace}(L_o L^T O)}\,, \quad x_s = \sqrt{\text{trace}(X_o X^T O)}$$
and apply the scaling to L_o and X_o:
$$L_{os} = L_o/l_s\,, \quad X_{os} = X_o/xs$$
4. Calculate the rotation and reflection matrix R by singular value decomposition of $L_{os}^T X_{os}$ such that points X_{os} best fit landmarks L_{os}
$$R = VSU^T \quad \text{for} \quad L_{os}^T X_{os} = U\Sigma V^T$$
where S is the diagonal matrix of the signs of Σ, i.e., $s_{i,i} = \pm 1$.
5. Finally, apply the rotation matrix R and reverse both scaling and translation:
$$\bar{X} = X_{os} R \cdot l_s - l_t$$

Procrustes analysis is applied, amongst others, in the area of geometric morphological analysis in order to describe the differences between various shapes found in, e.g., medical images [11]. It is also used as part of evaluation metrics in order to measure the difference between shapes after alignment, e.g., when extracting human shape and pose from images [66]. In Section 9.2 and 10.2 Procrustes analysis is suggested as a method to align consecutive local k-nearest neighbour maps as an interactive approach for exploration.

4.4 NEAREST NEIGHBOUR SEARCH

Searching for similar items or objects is a fundamental strategy applied in many scenarios. In order to design efficient algorithms for similarity search, the notion of similarity needs to be modelled. In mathematics, the concept of similarity and distance can be expressed by a function $d : X \times X \to \mathbb{R}$ between two objects $x_i, x_j \in X$, which assigns each pair of objects $(x_i, x_j) \in X \times X$ a value that represents the distance or similarity between x_i and x_j. Since distance and similarity are inverse concepts, there usually exists a transformation between the two.

Metric Properties

A distance function that satisfies the following three properties is called a metric.

1. If x_i and x_j are the same object, i.e., $x_i = x_j$, than the distance between them has to be zero, and vice versa:

$$d(x_i, x_j) = 0 \Leftrightarrow x_i = x_j$$

2. The distance between x_i and x_j is symmetric:

$$d(x_i, x_j) = d(x_j, x_i)$$

3. For any three objects $x_i, x_j, x_k \in X$, the sum of the distances between two pairs of the three objects must be greater or equal to the distance between the objects of the remaining pair, otherwise known as triangle inequality:

$$d(x_i, x_k) \leqslant d(x_i, x_j) + d(x_j, x_k)$$

However, there also non-metric distance functions, which, e.g., do not satisfy the triangle inequality. Unfortunately, there is no consensus on this terminology in the literature. In some literature, the term distance implies that all metric properties hold, while dissimilarity does not. In this thesis, both distance and dissimilarity are treated as synonyms, and do not necessarily imply metric properties except where explicitly mentioned.

Minkowski Distance

There are a number of metric and non-metric distance functions. Common metrics are the Euclidean distance ($p = 2$), Manhatten distance ($p = 1$) Maximum distance ($p = \infty$), or their generalization, the Minkowski distance. Given two vectors $\vec{x}, \vec{y} \in \mathbb{R}^n$, it is defined as:

$$d_{\text{minkowski}}(\vec{x}, \vec{y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Cosine Similarity

An example for a non-metric measurement of distance is the cosine similarity. It describes the cosine of the angle between to vectors and can be calculated as the inner product of two vectors normalized to unit-length.

$$s_{\text{cosine}}(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{||\vec{x}|| \; ||\vec{y}||}$$

In Information Retrieval, the cosine similarity is often used to measure the similarity of documents. When documents are modelled as term vectors, the cosine similarity allows to fairly compare documents of different size, since the length of the vectors is not considered for the calculation of the similarity. In Section 11.2 and 11.3, the cosine similarity is used as one facet to measure the similarity between complex objects containing textual descriptions, e.g., scientific papers and movies, respectively.

Jaccard Similarity

There are not only distance measures comparing vectors. The Jaccard similarity describes the similarity of two sets A and B. It is defined as the fraction of items common to both sets and the total number of unique items in both sets.

$$s_{\text{jaccard}}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

In Section 11.3, the Jaccard similarity is used as one facet to calculate the similarity of movies. Each movie consists of a set of genres, actors and directors. The Jaccard similarity measures the overlap of genres, actors and directors between two movies.

Similarity Search

Given a query object, the notion of distance or similarity induces a rank or ordering on other objects. Similarity search describes the task to find the most similar or a subset of similar objects from a database given a query object. In the most trivial case, a linear search can be performed. The term neighbour or nearest neighbour is used to reference a similar or the most similar object, respectively. Nearest neighbour search is used in various scenarios. In content-based recommendation, users are presented items, which are similar to items they have already viewed or bought, in the hope of them being relevant as well. Similarity search is often used to transfer knowledge from a set of known items to new items. In classification, a common approach is the k-nearest neighbour classifier, which assigns the majority label of k-most similar items to a not yet classified item. In Section 10.2, nearest neighbour search is used to find a set of similar items and generate a local similarity-based map.

Fast Similarity Search by Indexing

In case of very large data sets, a simple linear search may exceed the available time or resources to find similar objects. Therefore, strategies to accelerate similarity search have been developed. One idea is to build an index that accelerates similarity search. A basic indexing method is known as the k-d tree [7], which partitions the database of objects into a binary tree, such that a nearest neighbour search can be performed in logarithmic time with respect to the total number of objects. Each branch of the tree represents an axis-parallel hyperplane that divides the data in equally sized subsets. However, there are other indexing strategies, e.g., ball trees [81] or R-trees [48]. A downside of indexing methods is that the indexing data structure needs to be constructed in advance and updated over time in case new objects need to be considered, or objects are removed. The time to build an index usually exceeds the computational complexity of a linear search, which means they are only useful in case many individual similarity queries are performed on the same data set, and the data set does not change significantly over time. An advantage of kd-trees and R-trees is that they do not require a precise definition of the distance function at construction time, such that, e.g., weights of a weighted Euclidean distance can be chosen at query time. In Section 10.2, weights of a weighted distance function are suggested to be adaptable by users as a means of personalization. In this context, the nearest neighbour search could still be accelerated using kd-trees or R-trees.

Approximate Similarity Search

Another approach for fast similarity search is based on the idea of approximate search. In approximate similarity search it is not guaranteed that the true most similar objects are retrieved. Instead, retrieval performance is prioritized over accuracy. Current methods include locality sensitive hashing [60] and semantic hashing [125]. Hashing methods follow the idea of finding a hash function that assigns similar objects to the same bucket. A query can be performed by simply applying the hash function to an object. Similar objects will be located in the same hash bucket. However, more similar objects could have been assigned to a different bucket. In semantic hashing, the buckets are represented by binary codes such that similar codes are generated for similar objects. Therefore, the similarity search can be extended to nearby buckets in case no or insufficient similar items have been found.

In Sections 10.4, 11.1, 11.2 and 11.3 nearest neighbour search is not accelerated using indexing data structures or approximate similarity search even though it would be possible. In all experiments, either the data set was sufficiently small, or nearest neighbour search was not the limiting time factor.

4.5 METRIC LEARNING

Distance-based tasks like k-nearest neighbour search strongly depend on a suitable definition of a distance function given a specific usage scenario. However, in some cases, deriving a suitable distance function is difficult. Some feature or feature combination of a high-dimensional information space might not be relevant to a given task, and therefore, should not contribute to a distance calculation. Manual adaptations of distance functions, e.g., by feature weighting, can be challenging if features do not have a clear semantic, or in case there are many features such that a lot of effort is required. Often, it is not clear which feature or feature combination is or is not relevant to a given task.

Metric learning allows to automatically adapt the notion of distance based on simple training examples, e.g., pairs of objects that are considered to be very similar, pairs of objects that are considered to be dissimilar, or relative constraints, such that one object is considered to be more similar to a second object in comparison to a third. Early approaches include Xing et al. [161], who propose to improve clustering by metric learning based on pairwise similarity constraints. They suggest to learn a linear transformation of the feature space, similar to the idea of the Mahalanobis distance. Instead of using covariances in order to normalize distances, the transformation is learned via convex optimization of an error function that models the satisfaction of similarity constraints. Following [161], the distance d between two p-dimensional points $\vec{x_i}, \vec{x_j} \in X$ of a dataset $X \subset \mathbb{R}^{p \times n}$ of size n is defined as:

$$||x_i - x_j||_A = d_A(\vec{x_i}, \vec{x_j}) = \sqrt{(\vec{x_i} - \vec{x_j})^T A (\vec{x_i} - \vec{x_j})}$$

The matrix $A \in \mathbb{R}^{p \times p}$ describes a linear transformation and is required to be positive semi-definite. Again, in case of the Mahalanobis distance, A would correspond to the inverse covariance matrix of the data. The Euclidean distance is obtained by assuming A to be the identity matrix. Assuming pairs of examples $(x_i, x_j) \in S \subset X \times X$ that are known to be similar, as well as pairs of examples $(x_i, x_j) \in D \subset X \times X$ that are known to be dissimilar, with $S \cap D = \emptyset$, a convex optimization problem can be specified:

$$\begin{aligned} \min_A \quad & \sum_{(x_i, x_j) \in S} ||x_i - x_j||_A^2 \\ \text{subject to} \quad & \sum_{(x_i, x_j) \in D} ||x_i - x_j||_A \geqslant 1 \\ \text{and} \quad & A \text{ is positive semi-definite} \end{aligned}$$

Matrix A is optimized such that the distance between similar examples is minimal while the distance between dissimilar examples is kept larger than a constant and A is bound to be positive semi-definite, i.e., $z^T A z \geqslant 0$ for all $z \in \mathbb{R}^p$. The authors of [161] propose a gradient descent algorithm to find an optimal matrix A given training examples S and D.

Other metric learning methods are [22], which follows an information theoretic perspective, or [109], who specifically deal with the problem of few training examples in high-dimensional spaces. A survey of metric learning methods can be found in [76]. Applications include music recommendation [88], music organization [137], or information retrieval [89]. In this thesis, metric learning is suggested as a future extension to personalize the map-based exploration approach presented in Chapter 10 and briefly discussed in Section 10.2.

Part II

ENRICHING KEYWORD-BASED EXPLORATORY SEARCH

5 LOG ANALYSIS OF TRADITIONAL SEARCH BEHAVIOUR

Before tackling any open research problems related to designing effective exploratory search systems, it is considered vital to get a better understanding of how current search and exploration systems are used by end-users. Unfortunately, behavioural patterns of users performing exploratory search tasks are not yet well understood. This can mostly be attributed to the openness of the exploratory task itself, which is difficult to investigate and model. In addition, due to the wide variety of exploration approaches, results may not be transferable between system designs.

Aim

The aim of this work is to study keyword-based search behaviour via log file analysis of state-of-the-art search engines in order to identify potential ways to improve support for exploratory search tasks. Due to a cooperation with the German Youth Institute (DJI), comprehensive log data of three German search engines primarily used by children was available. Analysing search behaviour of children might provide exceptional insights into exploratory search patterns, since children are assumed to be inexperienced with using search engines, and the process of learning to interact with search engines might facilitate exploratory behaviour. In contrast, adults are expected to exhibit much more goal-oriented search behaviour, e.g., when performing navigational queries. Unfortunately, conclusions drawn from search behaviour analysis of children might not be transferable to adults, especially considering exploratory search behaviour, which is expected to be considerably refined with further education. Nonetheless, any insights into exploratory behaviour might generate new ideas to improve support in state-of-the-art search engines. A first analysis of the log data was published in [pub:1].

Related Work

Prior to that only Torres et al. [29] studied search behaviour of children by analysing query logs. However, they analysed log data from *AOL*, which was a common search engine frequently used by users of all ages. Queries were simply assumed to have been issued by children if a user clicked on a result, whose domain was listed in the *DMOZ kids & teens* directory. Unfortunately, this assumption can not be considered sufficient to exclude all search activities of adults, and might introduce additional bias towards certain information needs or search scenarios. In comparison, the following analysis is based on log data gathered from search engines specifically designed for young users. While most search activity should be attributable to children, even in this scenario a small fraction of queries may have been issued by adults, e.g., in case parents check whether the provided information is appropriate. Later, Torres et al. [30, 31] published detailed results for log data that can be linked to user profiles and includes the year of birth. However, even then users might have specified wrong information. In contrast to lab studies, some degree of uncertainty can not be avoided when analysing query logs. Besides research focusing on interaction behaviour of children, there is also ongoing research targeting exploratory search behaviour of adults. Recently, Schwerdt et al. [129] investigate search behaviour of adults using interaction logs and apply Hidden Markov Models to distinguish fact-finding and exploratory search tasks with high accuracy. Exploratory behaviour is characterised by user spending more time studying individual web pages, and less time scanning search result pages. Unfortunately, the following log data did not allow to draw conclusions about a user's dwell time, since it was not collected in a controlled lab environment.

Log Data

The studied log data was collected from three German search engines: *Blinde-Kuh.de*[1], *FragFinn.de*[2], *Helles-Koepfchen.de*[3]. They target a user group from age 6 to 13 and focus on providing child-appropriate content and search, i.e., each search index is curated such that no adult websites can be found and child-friendly websites containing simple explanations are prioritized. Additionally, all search engines visualize content in a way that balances ease of reading, understanding, and user engagement appropriate for children, e.g., by using a larger font size and including many playful and colourful images.

Combining all log files of all three search engines into one single data set yields a total of 2.5 million page requests gathered over one week in January 2011. Page requests can be grouped into roughly 600.000 browsing or search sessions. Each session is defined as a list of consecutive log entries from the same browser without a gap of more than 15 minutes. Page requests contain sufficient information to differentiate between various search interactions, e.g., queries, result page navigations, as well as click through data, i.e., clicks on results that will cause a user's browser to navigate to the actual result website. However, no log data beyond interactions with the aforementioned search engines could collected, e.g., further navigations on a result website, dwell times on particular pages, parallel browser windows, tabs or other interactions outside of the browser window. In total, 725.846 queries were recorded.

1 Blinde-Kuh.de as of 23. February 2011:
`https://web.archive.org/web/20110223005219/http://blinde-kuh.de/`

2 FragFinn.de as of 07. January 2011:
`https://web.archive.org/web/20110107123531/http://www.fragfinn.de/`

3 Helles-Koepfchen.de as of 03. February 2011:
`https://web.archive.org/web/20110203061701/http://www.helles-koepfchen.de`

5.1 SEARCH BEHAVIOUR ANALYSIS

Tables 2 and 3 compare the most frequent search queries between Google[4] and the combined log data of all three child-appropriate search engines. Popular queries for Google mostly reference certain websites, which indicates that users most likely intended to navigate to that particular website. Popular queries from the combined log data reference various topics or objects, which indicates that users most likely intended to inform themselves about these topics. This difference supports the hypothesis that users of child-appropriate search engines, i.e., children, are more likely to follow exploratory search patterns in order to satisfy their broader information need. Unfortunately, popular queries of the combined log data may not accurately reflect the true query distribution, since *Blinde-Kuh.de* provides a static set of categories, e.g. *games* or *animals in winter*, which are illustrated by a set of images and will lead to a search result page when clicking them. In this case, users do not enter search queries themselves. Instead, they are automatically forwarded to a search result page as if they would have entered the name of the category as search query. Unfortunately, there was no way to differentiate these interactions from regular search queries based on the provided log data.

Figure 8 illustrates the distribution of relative frequencies for various search related parameters. Based on the log data, queries consist of on average 1.8 terms, or 2.4 terms when considering only unique queries, see Figure 8 (a). In comparison, queries issued by adults vary on average between 2.4 and 2.7 terms, see [134]. This contradicts previous findings published in [9], which state that children would often use longer queries resembling natural language. In addition, later findings confirm this trend of shorter queries, see Torres et al. [31]. However, as discussed before, the difference in query length may also be partially explained by additional artificial queries generated by *Blinde-Kuh.de*, which consist mostly of single word queries reflecting category names.

All three child-appropriate search engines provide 10 search result per page, however only the first two to four results can be seen at the same time on common wide-screen monitors. The log data reveals that children most often visited only the very first result page and clicked on one of the first three results, see Figure 8 (b) and (c). In comparison, adults also most often only view the first result page. As reported by [134], in 70% of queries adults only view the first top ten results. This circumstance is considered a major obstacle for effective exploratory search, since users will not be able to get a sufficient overview over the result set from inspecting only a few search results.

Based on the log data, the average session length consists of 1.8 queries when ignoring duplicates, see Figure 8 (d). In comparison,

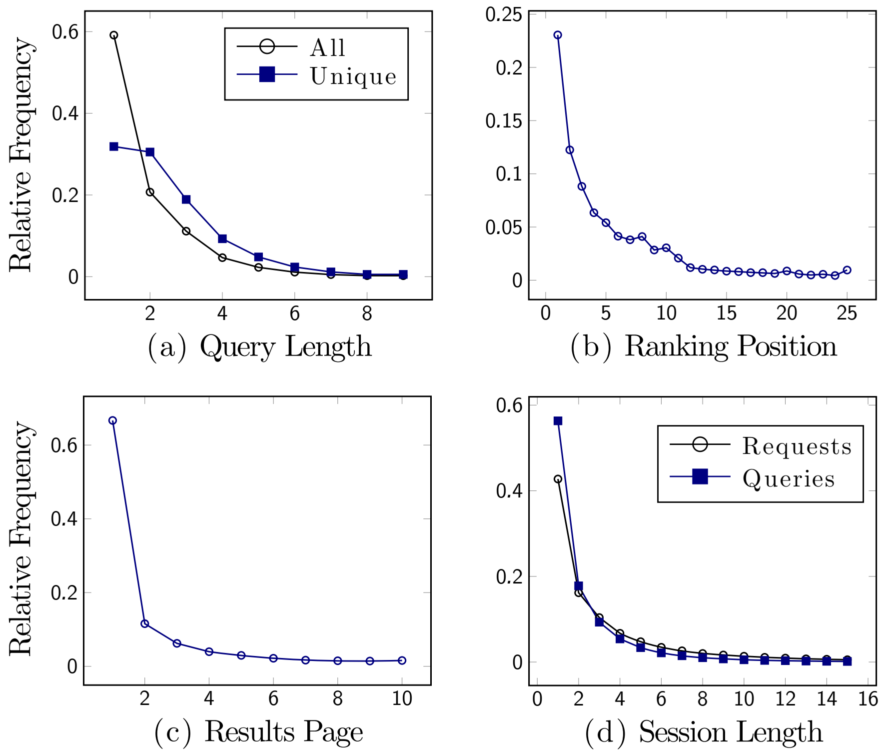
4 As reported by the *Google Insights for Search* service currently known as *Google Trends*: `https://trends.google.com/`

1. games	2. sex	3. electricity
4. animals in winter	5. squirrel	6. whale
7. dogs	8. animals	9. egypt

Table 2: Most frequent search queries (translated from German to English) derived from the combined log data of *Blinde-Kuh.de*, *FragFinn.de* and *Helles-Koepfchen.de*.

1. facebook	2. youtube	3. google
4. ebay	5. you	6. weather
7. amazon	8. web.de	9. gmx

Table 3: Most frequent search queries (translated from German to English) as provided by Google Insights during the same time period the log data was collected.



(a) Query Length (b) Ranking Position (c) Results Page (d) Session Length

Figure 8: Relative frequencies of search parameters over the whole data set: (a) the query length in number of words, (b) the ranking position of clicked results, (c) the number of times a certain result page was viewed, and (d) the session length in page requests including queries and queries only.

the average session length of web search engines used by adults is reported to consist of 1.6 queries, see [134]. Literature suggests that children follow a less structured browsing behaviour, occasionally repeating clicks and queries, whereas adults search and browse more systematic, see [9]. Base on the log data, users visited the same URL repeatedly in 16.6% of all sessions and performed the same query multiple times in 20.5% of all sessions. This observation supports the hypothesis that children may often follow an exploratory search approach, even though they might lack some experience to avoid repeating page clicks and queries.

5.2 IMPLICATIONS FOR EXPLORATORY SEARCH

Current web search behaviour of both children and adults indicates that state-of-the-art search engines and user interfaces lack important features supporting exploratory search tasks. User often only inspect very few search results on the first result page, which provides only a very limited and biased view on the overall result set distribution. In other words, users often do not know about potential relevant results providing novel information, which are hidden in the undiscovered space of search results appearing on second or later result pages. For exploratory search tasks, users would greatly benefit from additional information about the global and local structure of the search or information space. Several new methods supporting exploratory search tasks are presented in the following sections. However, further research is required to fully understand exploratory user behaviour. Unfortunately, due to the openness of exploratory tasks, user behaviour is very difficult to record, model and predict. It can be expected that search behaviour observed from users which know that they are being observed, e.g, as done by [129] in a controlled lab experiment, exhibits a significant behavioural bias, such that results can not be transferred to uncontrolled day-to-day user behaviour. Uncontrolled search behaviour could be considered to be much less focused due to frequent interruptions common in modern life, e.g., email notifications or social network messages. On the other hand, anonymised log studies as suggested above can not effectively capture exploratory search behaviour due to data privacy concerns, which prevents recording all relevant interactions.

Summary

In summary, query logs of three German search engines designed for children were analysed, essential statistics of the average search behaviour were extracted, and results were compared to previous findings for both adults and children. Most previous findings could be confirmed. For example, based on the top ten query terms, adults favour navigational queries, while children prefer informational ones. However, in contrast to previous findings, children formulate shorter queries than adults on average. In context of this thesis, statistics reveal that both children and adults often only inspect the first few search results, and thus, are likely to be trapped in information bubbles induced by the ranking of search results. Accordingly, further research is needed to develop effective and easy to use methods that support the exploration of search results.

6 ONTOLOGY-SUPPORTED EXPLORATORY SEARCH

The following section describes an experiment supporting exploratory search scenarios by annotating search results with entities from a domain ontology. The initial concept was published in [pub:7].

Aim

The aim of this work is to support users during a traditional keyword-based search in an unfamiliar domain. It is hypothesized that users searching in an unfamiliar domain would benefit from additional information presented by annotating search results with concepts from an ontology describing that particular domain. On the one hand, terms from such an ontology may help users to refine their information need and help to formulate more specific search queries. On the other hand, connecting concepts from such an ontology with search results may support users in quickly judging the relevance of search results without a detailed inspection of each document.

Furthermore, the aim of this work is to investigate whether ontologies can be extended by users while they are familiarizing themselves with a domain. The idea is that users getting familiar with unknown terms may be motivated to contribute their newly gained knowledge by adding axioms to the ontology, which then will be immediately used to further annotate search results with these new concepts. As a side effect, by enabling users to extend the ontology, building and maintaining a domain ontology might be more cost effective than paying domain experts.

Related Work

Originally, ontologies have been discussed as a means to support navigation in hypermedia [21]. Later on, a number of methods have been developed, which focus on supporting users during search based on information from ontologies. In [141] concept hierarchies are generated from ontologies, which are then presented to the user as filters of a faceted search interface. In [92], ontologies are used to enable users in formulating complex search queries via a tree visualization of semantic query parameters. In [148, 132], ranking algorithms have been extended to consider the concepts from an ontology that are linked to a document. In contrast to previous work, the proposed approach provides an on-demand annotation of search results suitable for web-scale search systems and allows users to extend the ontology during a live search session.

Scenario

In order to investigate whether users are able to more efficiently familiarize themselves with an unknown domain by the proposed approach, a suitable search scenario and software prototype was designed[1]. The scenario deals with the process of planning fitness exercises. It was assumed that users only have basic knowledge about physical therapy as well as any medical implications of certain physical exercises, e.g., whether some exercise can be safely performed or might even benefit reducing back pain. Hence, the goal was that users would familiarize themselves with any relevant exercises and their effects given a specific medical condition. In addition to a traditional keyword-based search and search user interface, results are annotated with relevant concepts from an ontology of physical training exercises. These annotations are hypothesized to support users in quickly judging the relevance of search results, as well in refining their information need during this exploratory search scenario. Furthermore, in case users would identify new exercises that are not yet contained in the existing domain ontology, users are enabled to add axioms about this new exercise to the ontology, which would then be used to annotate search results.

Such a search system could be envisioned as an assistance system in fitness centres, where users would be able to customize physical exercise plans by independently researching new exercises as well as their suitability given individual medical conditions, e.g., back pain.

In the following, design consideration for a software prototype are discussed, including the annotation process of search results, the visualization of annotations, and a user interface prototype which guides users to extend the existing ontology.

6.1 ANNOTATING SEARCH RESULTS

In order to visualize relevant concepts from a domain ontology given traditional keyword-based search results, links between the ontology and result documents need to be identified first.

Preprocessing vs. On-demand Annotation

This process of linking ontology concepts with result documents could either be done as a preprocessing step when indexing documents, or as an on-demand annotation process that is performed at query time. Since the ontology is supposed to be extended by users during live search sessions, each newly added concept would require a full revision of all indexed documents in order to immediately find links between the documents and newly added concepts. However, a full revision of all indexed documents is not achievable with acceptable response times in an interactive live search session. Therefore, the proposed approach follows the strategy to determine links on-demand at query time. As a side benefit, an on-demand approach allows to augment existing state-of-the-art search systems, e.g., web-scale search systems that provide search results as a web service. A downside of this approach is that information from the ontology can not be used

1 `http://www.dke.ovgu.de/ISWC2015.html`: Video of software prototype demonstrating the envisioned search system as described below.

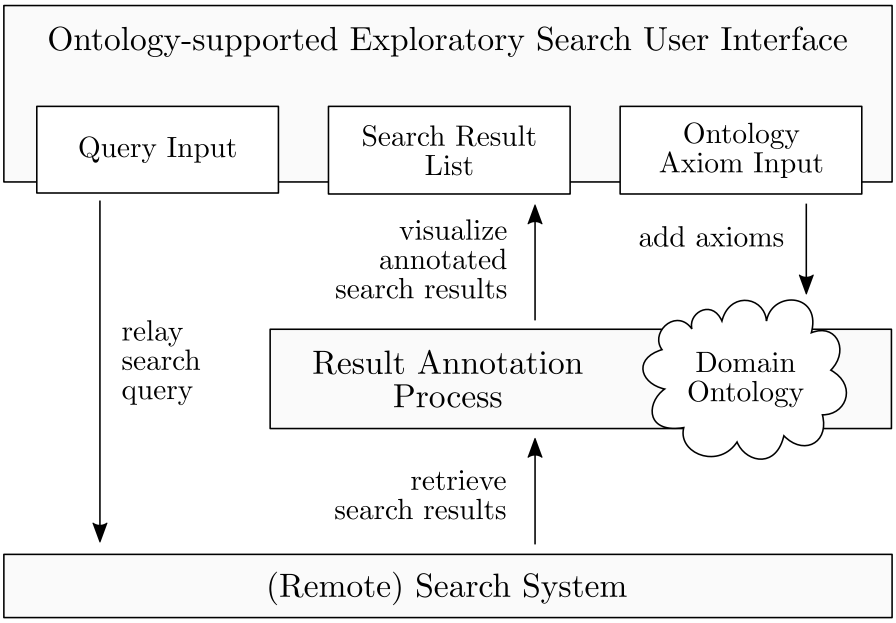


Figure 9: System design of the software prototype: the query input entered by a user is relayed to a (remote) search system; search results are annotated with concepts from a domain ontology before they are presented to the user; users might add new axioms to the ontology during the search session.

to influence the ranking of the search system, which could be highly beneficial when searching documents of a specific domain. In the following, it is assumed that the base search system provides both presentational information about each search result, e.g., the title and snippet, as well as the original document, which than can be analysed for links to the domain ontology. Figure 9 shows the basic system design of the software prototype.

Linking Documents with Ontology Concepts

In order to find links between a document and the domain ontology, documents need to be analysed. There are a number of research approaches to match ontologies and documents. Usually, the idea is to evaluate whether an ontology fully covers a given domain by comparing it against a domain-specific corpus. This approach is considered a data-driven alternative to comparing an ontology with a gold-standard ontology, which might not be available for a domain [110]. In [12] it is suggested to extract terms from both the ontology and corpus and measure the overlap between the two sets. One idea would be to do a simple counting using a vector space representation of the terms. Additionally, more detailed lexical analysis can be performed, e.g. by expanding the corpus term set using synonyms and hypernyms. Furthermore, [12] propose to apply clustering to be able to also capture structural differences that can not represented by considering simple sets of terms.

Unfortunately, most in-depth approaches require a significant amount of processing time to do natural language analysis. Since high system response times would lead to user dissatisfaction, this work proposes a simple but efficient matching method. First, the document text is tagged for part-of-speech. Then, all nouns are extracted and normalized by applying a language-specific lemmatisation method. Finally, the resulting lemma is checked against concepts and synonym sets of the domain ontology. This basic matching can be done within a minimum amount of time, and thus, an on-demand annotation is achievable within an acceptable system response time.

For each search result document, the output of the result annotation process consists of a number of matched concepts from the ontology. For each concept, a short description is provided by the ontology itself. Additionally, relevant text snippets are extracted from the search result document, which describe the context of a match.

6.2 VISUALIZING ANNOTATIONS AND ONTOLOGY INFORMATION

Once relevant ontology concepts are identified, search results and concepts are both visualized to the user. An illustration is given in Figure 10. A screenshot of the final prototype is shown in Figure 11.

This work uses a classic list-based search result visualization. Each search result is described by the document title, the document URL and a short text description consisting of text snippets describing the context of query matches. Ontology concepts that have been matched with a search result document are visualized as a simple horizontal list of terms. By hovering over each term, a short description of the concept is shown in a overlay box together with relevant text snippets describing the context of ontology matches.

For easier interpretation the ontology was divided into three parts: a personal ontology containing only axioms that were added by a user during the live search session, the domain ontology containing prepared axioms about fitness exercises, and a selection of related

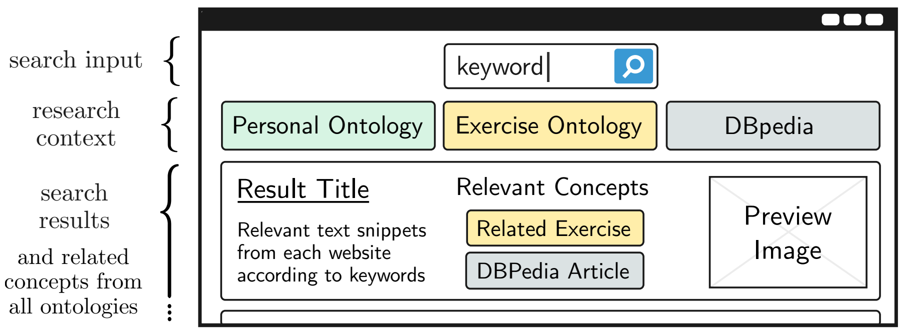


Figure 10: User interface concept enriching search result presentation with additional domain-knowledge, e.g., related exercises from an ontology, see [pub:7].

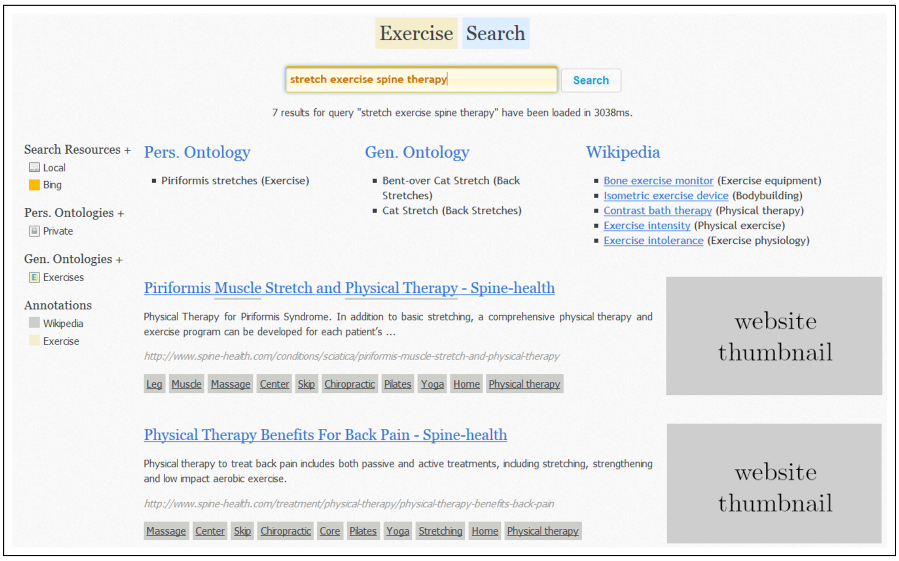


Figure 11: User interface prototype showing related entities from three ontologies both as separate results list and as annotations for each web search result in case a concept was matched inside the corresponding full-text document. Website thumbnails removed due to licensing concerns.

concepts extracted from DBpedia [78]. Annotations are visualized in three different colours depending on the source ontology: green annotations represent concepts matched from the personal ontology, yellow annotations represent concepts matched from the domain ontology, and grey annotations correspond to DBpedia concepts, see Figure 10. Due to the colouring of annotations users can easily identify the source ontology of a concept, which might help them to judge the relevance of concepts. For example, a grey annotation refers to a general concept from DBpedia, which might not specifically correspond to any physical exercise, see grey annotation ″Muscle″ in Figure 11.

DBpedia Ontology

Entities of the DBpedia ontology [78] were extracted by following ″subject″- and ″broader″-predicates of the seed concepts ″physical exercise″ and ″physical therapy″ up to the second degree. The final subset of the DBpedia ontology consists of 4375 entities referencing their respective Wikipedia articles.

Fitness Ontology

The domain ontology was extracted in cooperation with the website ExRx.net[2]. It targets exercise professionals and fitness coaches, and provides a library of articles describing many physical training exercises. The extracted ontology contains 7924 axioms about 367 exercises. For each exercise a number of details were extracted: the name of the exercise, a short description of preparation steps, a short description of execution steps, a general comment, an illustrative animation, muscle

2 http://exrx.net: an online library for information about physical training exercises and data source for the fitness ontology

groups that are engaged by a specific exercise, and more. However, the proposed prototype only utilizes the title, all three description texts and the illustrative animation of an exercise. Further information about each exercise were not shown to the user nor used for matching concepts with search results.

Personal Ontology

The personal ontology was modelled after the domain ontology, but only contains entities that were specifically added by a user during a live search session. Hence, at the beginning of a search session the personal ontology is initially empty. As soon as a user identifies relevant information about an exercise not yet present in either of the other ontologies, the user is enabled to add a new entity to the personal ontology, see Figure 12. This extension process is described in more detail in the following section.

In order to support the user in identifying which exercises and related concepts are already present in which ontology, a result list for each of the three ontologies was added above the actual web search result list visualizing direct matches between the user's search query and entities from all three ontologies. In Figure 11 it can be observed that at the time the user already added one stretch exercise to the personal ontology, which matches the current search query (left column of Figure 11 labelled "Pers. Ontology"), but there are two further stretch exercises present in the prepared fitness ontology (centre column of Figure 11 labelled "Gen. Ontology"). Figure 11 also illustrates that the first two web search results were not annotated with any of these domain specific ontology concepts. However, many generic matches with the DBpedia ontology could be found, which indicates that both search results are at least generally related to the topic of physical therapy, but do not reference any known physical exercises. Hence, the user might be inclined to further inspect these search results in order to find new relevant information about stretch exercises.

6.3 EXTENDING DOMAIN ONTOLOGIES

During a search session, users might identify new relevant exercises, which are not yet contained in any of the ontologies. In order to add new exercises to the search system and benefit from annotations in future search session, users might be motivated to extend the domain ontology by themselves. This work proposes a simple user interface concept, which allows to add a number of predefined axioms by users.

An illustration is given in Figure 12. The user interface is activated as soon as a user clicks on any search result. On the right side of the screen the original result document in presented. On the left side a form allows to enter detail information about new exercises, in particular the title, preparation and execution steps, as well as a comment. Based on the information entered into the form, a fuzzy

Figure 12: User interface prototype illustrating how users are enabled to extend the personal ontology by entering axioms and verifying that no duplicates are added. Webpage blurred due to licensing concerns.

search is performed in order to identify possible duplicate entries that are already present in one of the ontologies. In case the user doesn't assess the new exercise to be a duplicate of already existing exercises, the user can confirm the action of adding the exercise to the personal ontology.

Once new axioms have been added, they are immediately considered during matching tasks of any subsequent search results. Due to the immediate adaptation of the search system, it is assumed that users perceive the creation of a personal ontology as a new tool to customize and improve their search experience.

Outlook

Besides some technical improvements to the prototype, e.g., to enable users to edit or remove previously added axioms, there are a number of possible enhancements to the system design. Instead of requiring users to enter details about new exercises into a form, there could be an automated extraction process that generates candidate axioms that only need to be confirmed by users. In [95] the authors propose such a semi-automatic interactive ontology revision system, where users can approve candidate axioms extracted from documents which are ranked by a measure of their level of validity with respect to already confirmed axioms of the ontology. Furthermore, personal axioms added by users could be promoted to the domain specific ontology visible to all users if a certain level of consensus can be detected, e.g., when multiple users add the same or very similar axioms.

Summary

The proposed search prototype demonstrates how traditional search user interfaces can be extended to support users during exploratory search by providing additional information about each search result in terms of annotated ontology concepts. It is hypothesized that such annotations will improve the time users require to judge the relevancy of search results given an exploratory search scenario similar to that presented in this Section, i.e., researching physical training exercises suitable for certain medical conditions. It is also hypothesized that users will be able to more quickly discover relevant new information in such a scenario, since search result annotations allow to grasp the content of each result given the current context of the search scenario provided by a domain ontology. In contrast, the traditional approach of showing text snippets of query matches in order to convey contextual information about each search result is only based on query terms, but does not benefit from related concepts of a domain ontology. Furthermore, it is hypothesized that users will be most motivated to contribute their research effort during a live search session at the time new information is being discovered. Thus, a concept that allows users to extend the domain ontology was developed.

Unfortunately, initial interviews with experts in technology scouting have not indicated that users would significantly benefit from this approach during exploratory search tasks. This is believed to be connected to two problems. First, the overall responsiveness of the prototype suffered from long waiting times, since the retrieval process of web search results did entail substantial network latency and the annotation process did entail substantial processing time. Users were reluctant to navigate to subsequent search result pages, which would mean even more waiting time and instead focused on refining the search query. Secondly, the additional visualization of related entities did significantly increase the space required to visualize each result. As a consequence, less results could be shown without scrolling, which actually lead to less results being investigated by users.

However, a detailed comparative user study is required in order to be able to confirm or disprove these initial insights. Additionally, both problems can be mitigated in the future. The result visualization could be optimized to save screen space by balancing which information is shown to users depending on several factors. For example, if concepts or synonyms of concepts are present inside the excerpt of a search result, that concept could be visualized by highlighting the respective terms inside the excerpt instead of visualizing it as a separate box. In addition, the responsiveness of the prototype could be improved by using a fixed data set instead of live web search results. Then, both the annotation and search process could be implemented without suffering from extensive network delays, which are the consequence of retrieving web search results and webpage contents live during a search session.

7 A FRAMEWORK FOR FCA-BASED EXPLORATORY WEB SEARCH

The following section describes an approach to support users during exploratory search based on a hierarchical visualization of search results derived by Formal Concept Analysis (FCA). Initial thoughts and an early prototype were published in [pub:6] and [pub:11].

Aim

The aim of this work is to support users during exploratory search scenarios by providing a structured view on search results. In contrast to a traditional list-based representation, this structured view is supposed to convey information about various domains embedded in the set of search results. As a means of structuring, this work investigates the potential of a method known as Formal Concept Analysis (FCA) [book:7], which allows to generate a hierarchical cluster structure representing topics or domains at different levels of granularity: from most general to most specific. It is hypothesized that users will benefit from such a presentation in an exploratory search scenario. For example, users might discover previously unknown domains, which would not be apparent in a list-based representation, but are easily recognized in the proposed hierarchical cluster visualization. In addition, FCA also provides information about shared features for each cluster, e.g., common terms used in documents of a certain domain. Therefore, users might also benefit by being able to more easily extend the original search query using these common terms provided by FCA as a form of domain specific query term recommendation.

Related Work

The method of Formal Concept Analysis has been previously applied to the area of information retrieval, but did not yet gain high traction in the field. In [108] FCA is proposed as a method to support users in formulating queries. Each query condition is represented by a concept lattice, which is supposed to improve the user's understanding of query terms and their relations by visualizing them graphically instead of using a boolean query language. In [17] concept lattices are not directly visualized, but used to provide a query-specific concept hierarchy, which can be used to interactively filter the search result list. In [24, 3], concept lattices are constructed from web resources by extracting keywords and applying FCA on a document to keyword context. The resulting concept hierarchy is used for browsing and filtering of documents and search results. In this work, a similar keyword-based approach is chosen. However, the concept lattice is directly visualized as a graph of search results. Recent work [62] discusses how FCA can be used to extract ontologies from unstructured documents. Assuming that higher-level concepts and relationships can be effectively extracted from search results, improved exploration methods are conceivable.

7.1 EXTRACTING CONCEPT LATTICES FROM SEARCH RESULTS

This work proposes to use a method referred to as Generalization of a One-Sided Concept Lattice (GOSCL) as presented in [14]. The key benefit of GOSCL is that attributes can be independently modelled as discrete values, continuous values or domain specific sub-lattice structures depending on their type, instead of relying only on simple binary values. For example, terms of a document can be modelled not only by their presence or absence, but by the number of their occurrences. The publication date of a document could be represented as a continuous value. A domain specific categorization of documents could be modelled as a set of discrete values representing each class.

Unfortunately, a more detailed modelling of documents also leads to much more complex concept lattices, which are difficult to visualize to a user. In the following, we propose to use an existing graph exploration toolkit that allows for an interactive exploration of small concept lattices. For demonstration purposes, the size of the lattice was artificially reduced by only considering a small number of documents. In the future, it is imperative to do further research regarding an effective exploration of large concept lattices. Alternatively, concept lattices could be simplified before visualization, e.g., by removing concepts that provide only minimal gain of information, meaning concepts that are very similar to its super concept and only differ in a small detail. Thanks to the explicit modelling of attributes by the GOSCL approach, sophisticated scores could be designed that measure the information gain when discerning sub concepts from their super concept similar to the measures used in decision tree learning.

7.2 PROTOTYPE FOR GRAPH-BASED EXPLORATORY SEARCH

In order to demonstrate the proposed method a prototype was developed. The prototype builds on a state-of-the-art web-scale search system, which provides web search results as a web service. When a user confirms a search query, the query is relayed to the web service. A soon as the list of search results is retrieved, results are processed with the aforementioned FCA approach, and the generated concept lattice is visualized to the user, see Figure 13.

Unfortunately, retrieving web results from the chosen web service took a considerable amount of time, often up to a full second for a batch of 10 results. Since high system response times would lead to

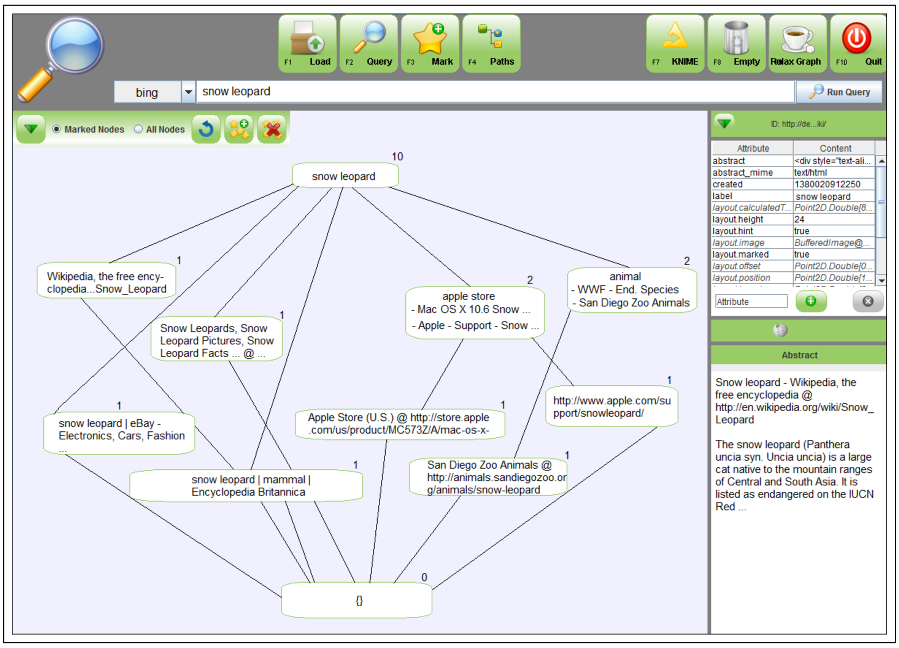


Figure 13: Illustration of user interface prototype showing a concept lattice of search results for the query "snow leopard". At the top, the search query can be specified. In the centre, the lattice graph is visualized from top (most general) to bottom (most specific). On the right side, details of the currently selected node are presented.

user dissatisfaction, the prototype currently only analyses the first 10 search results. At the time, the only alternative was to use a local search system based on a considerably smaller corpus. Since the prototype was supposed to be evaluated in an exploratory search scenario, it was judged that the advantage of a web scale search system outweighs the downside of being limited in the number of results that can be processed. In order to evaluate an exploratory search scenario, it is important that users are able to freely formulate search queries and discover previously unknown documents, which is difficult to achieve with a small corpus.

Also, processing more than 10 documents yields concept lattices with much higher complexity, which can not yet be effectively visualized with the proposed prototype. Unfortunately, this constitutes a major drawback of the prototype. It can be expected that a concept lattice extracted from only 10 search results will most likely not cover all possible domains related to a query. Furthermore, concepts will only contain a very limited number of documents, which will most likely impair the quality of common features extracted from a document cluster.

The lattice is visualized by an existing graph visualization framework called Creative Exploration Toolkit (CET), see [54]. The framework supports automatic layouting of arbitrary graphs, and allows to visualize labels for both nodes and edges. An illustration of the prototype is provided in Figure 13. Unfortunately, the CET tool does not directly support to visualize lattices. Therefore, the user needs to identify the root node, which will not always be positioned as the top node of the screen. In the future, the prototype should can be improved by adapting a graph layouting algorithm to consider properties of concept lattices.

The intended work flow follows the pattern of a classic berry picking model. First, the user specifies an initial, potentially ambiguous search query, e.g., "snow leopard", see Figure 13. The resulting concept lattice will visualize various search results with respect to their common terms, which should yield clusters that represent different domains. The user can inspect both the structured visualization as well as each result individually. Based on the information gained during the exploration phase, the user then will modify the search query to focus on a specific domain related to the user's information need. This process repeats until the information need is satisfied.

Currently, the prototype only considers the presence and absence of a term from a short description of a web result provided by the web service. In the future, the prototype is supposed to be extended to integrate additional meta data in order to utilize the benefits of the proposed GOSCL approach, e.g., the relevancy score or term frequencies of a document.

Summary

In this work an approach for exploring web search results is proposed as well as demonstrated by an early software prototype. The main purpose is to support users by providing an overview over the various domains included in web search results, especially when ambiguous queries are used, which is often the case in exploratory search scenarios. The method is based on recent advances in the field of Formal Concept Analysis, which allows to structure documents in an hierarchical way as concept lattices. The developed prototype demonstrates how concept lattices can be visualized for a limited amount search result. Unfortunately, the prototype still lacks a number of key features that would be required for performing a suitable evaluative user study. Most importantly, due to technical constraints, the prototype is only able to process 10 search result, which significantly limits its usability. In the future, it is imperative to further investigate methods that allow to explore large concept lattices or effectively simplify complex lattice structures.

Part III

MAP-BASED EXPLORATORY SEARCH

8 CHALLENGES AND BENEFITS OF MAP-BASED EXPLORATION

In this section map-based visualizations are motivated and introduced as a method for exploratory search and browsing. Their advantages and disadvantages are discussed, especially in comparison to traditional list-based approaches. In the following, browsing is not considered separately, but viewed as a form of search with no queries, i.e., random browsing, or a selection of predetermined queries, i.e., browsing of items of predetermined categories or orderings.

Issues with Traditional List-Based Search

Traditionally, most search and exploratory search user interfaces are based on a linear result presentation. Search results are ordered by a relevancy score and represented as a list of results with decreasing relevancy. Each result is described both by its query independent key properties, e.g., the title of a document, as well as by query dependent properties, which convey how a search result relates to the query, e.g., the snippet highlighting query terms as they appear in a matching document.

There are a number of benefits of such a list-based representation. A list of results is intuitive and unambiguous in comparison to, e.g., a grid visualization, where results could be ordered horizontally or vertically. Results can be represented in great detail such that users can quickly judge whether they are relevant. Text, such as document snippets, can be fluently embedded. Snippets allow to describe the context of query matches, which can help users to develop a sense of how documents are scored, or provide inspiration on how to modify queries in order to better capture the user's information need. Most importantly, users are familiar with list-based search result visualizations, and thus, are highly efficient in using them. Studies have found no significant improvement in search efficiency for other basic result representations, e.g., a table of results [117]. This is the reason why a list-based representation is and has been the de facto standard in state-of-the-art search systems for many years.

However, there are also a number of significant disadvantages of list-based representations. Since lists are one-dimensional constructs they can not convey multidimensional relationships between search results. They are discrete representations of chains of elements, which means they can not convey relative differences between elements, or model density-based data properties like clusters or outliers. In addition, due to the high detail of how results are often presented to the user, only a few results are presented at the same time. As a consequence, users often inspect only very few search results given a search query, which in many cases will not accurately represent the data distribution at all. Users potentially will miss out on many results of different clusters, which can not all be part of the first few elements of the result list. This issue is sometimes described as users being trapped in so called information bubbles, which are restricted sub spaces caused by this limited exposure of users to a very small minority of all matching objects.

Mitigation through Exploratory Search Components

Some of these disadvantages of list-based representations have been already studied in detail. Many solutions coincide with the goals of researchers that develop new approaches aimed at improving the overall search experience. For example, query suggestion methods aim to provide alternative but relevant keywords, which will lead the user to related result lists improving the variety of search results. Similarly, recent search systems are often designed to immediately display query expansion suggestions and their corresponding search results even while users are still typing keywords. Due to this almost instantaneous response of the search system, users are enabled to quickly and easily switch between queries, which is a key element consistent with the traditional model of berry picking in exploratory search, see Section 2.2. Still, users can only inspect one result list at a time.

Alternatively, faceted search approaches provide additional structural information by allowing users to filter search results based on existing categories. Each category is often annotated by the number of results that match this particular category, which provides minimal quantitative information about the result set distribution with respect to the set of categories. Unfortunately, categories are often static and query independent, which limits their use for discovering query related clusters or outliers.

In this thesis, it is argued that a traditional list-based representation of search results is a vital means of an exploratory search system. However, further methods should be researched to support users in grasping both the full scope and variety of search results as well as their structure in terms of clusters and outliers. Therefore, methods need to be investigated that visualize a representative sample of all matching search results and convey density information about the result distribution. For that reason, map-based approaches are discussed as one kind of such result visualizations.

Benefits of Map-Based Search and Exploration

A map-based visualization is considered a 2-dimensional arrangement of a set of objects. Each object is displayed as some form of icon referencing or summarizing an object. Due to the spatial visualization of objects, density information can be represented, meaning objects can be placed close to each other to form clusters, or be positioned far away from other objects to illustrate them as outliers. A detailed formalization is given in the next section.

Number of results that can be visualized

When visualizing search results as a map, there are both advantages and disadvantages in comparison to list-based representations. First of all, more results can be visualized since lists are traditionally arranged either vertically or horizontally without line breaks. Thus, list-based representations often do not fully utilize all available screen space, especially in case of opposite aspect ratios, i.e., vertical lists on landscape format screens, or horizontal lists on portrait screens. In contrast, map visualizations are not restricted in that way and can utilize all available screen space, which often allows to display many more search results at a time.

Panning and Zooming

Map-based visualizations benefit from the spatial concept of panning and zoom, which are important tools to allow users to both get an overview over the whole collection, but also allow to zoom in to sub-areas in order to inspect a subset of objects with more detail. Getting both an overview and detailed information when needed, is an important pattern that allows to convey structural information. In the extreme cases of exploratory search, where users may not even be able to think of an initial search query, objects can still all be discovered by the means of panning and zooming.

Cluster Selection

In contrast to lists, which are often only presented partially to the user, map visualizations enable users to provide feedback on a more general level. For example, since maps provide a full representation of all objects, users could select a cluster of search results by drawing a border around them. Based on this subset selection, users may than provide feedback to the search system, e.g., that these search results are not relevant to the current query.

Cluster Visualization

Most importantly, map visualizations allow to convey density information about result sets including clusters and outliers, which is simply not possible for list-based representations. Clusters enable users to quickly generalize knowledge from inspecting a few representative neighbours of a cluster. Based on this generalized knowledge, conclusions about the meaning and overall relevancy of clusters can be drawn. For example, users can either zoom into a cluster, in case it seems relevant, or move on to another cluster, effectively filtering many objects without inspecting each single one.

This is the main motivation for applying map visualizations to the scenario of exploratory search and exploration.

8.1 USER INTERACTION CHALLENGES

Unfortunately, map visualizations also need to deal with many challenges not applicable to list-based representations. In the following, a number of key challenges are described, which need to be considered when designing map-based information systems that are supposed to support users during exploration and exploratory search tasks.

Challenge 1 (conveying structural information)**:** In general, the idea is to convey information about the structure of a set of objects. This structural information needs to be sufficiently comprehensible by users. That is, the arrangement of objects needs to be generated in a way that is meaningful to users. While there are many existing projection methods that can be applied, some might be more suitable for generating comprehensible arrangements than others.

Additionally, suitable means of visualizing the arrangement have to be found. For example, while a simple scatter plot, where each object is represented by a plain dot, will convey density information and illustrate clusters, it does not explain the nature of relations between objects of a cluster. Thus, map visualizations need to present additional information about the meaning of clusters or the arrangement of objects in general. For example, clusters could be annotated with descriptive textual labels.

This challenge is further discussed in Section 9.1 in terms of which projection method is best suited for map visualizations, and in Section 9.2 in terms of how well projection methods are able to adapt maps when the underlying data collection is extended with new items. However, generating textual labels describing clusters has not been studied in this thesis.

Challenge 2 (find representative items)**:** Alternatively, a cluster could be portrayed by a representative sample of objects, where each object itself is visualized by a descriptive icon. Then, users would be able to draw conclusions about the meaning of a cluster by comparing multiple representative samples of one cluster with other representative samples of other clusters. This immediately raises the question, which objects of a cluster are "good" representatives of that cluster, what does "good" mean, and which methods can be used to extract good representative samples from clusters.

In the area of image retrieval, finding diverse representatives is an open research question, see [27].

In Chapter 10, this problem is avoided by visualizing only small maps of limited subsets of objects at a time, such that no sampling is necessary.

Challenge 3 (summarizing objects)**:** Another open research question is how to generate suitable icons that summarize objects. This includes not only the question of how to summarize a specific object, but also at which level of detail an object should be summarized given additional constraints or context, e.g., screen space limitations, or the user's current information need. In case of images, thumbnails of various size can be generated. For other media, there is no natural icon representation. Documents could be visualized by a thumbnail of its first page, which is most likely not readable, and thus, does not convey a lot of information. Documents could be visualized as keyword histograms, see [26]. Or they could be summarized into a few sentences, see [142]. Other media or information objects entail similar challenges. However, they are not further investigated in this thesis. In Section 11.2 and 11.3, prototypes use naive icons, e.g., movies are represented by their title and cover image, or scientific papers are represented by their title and authors.

Challenge 4 (overlapping icons)**:** Furthermore, in case objects are represented by some form of icons, they can easily visually overlap in case objects are supposed to be positioned very close to each other. As a consequence, any user interactions would be very difficult, since parts of the object icon would be concealed by other icons.

In Section 9.3, a method for grid regularization is proposed, such that object icons will not overlap and at the same time density information is reasonably preserved.

Challenge 5 (visualize query relevance)**:** Another problem is that objects are no longer necessarily be ordered by relevancy score. That is, since results are positioned to form semantically meaningful clusters, their position will often not reflect the relevance of a search result to the query. For example, the very first search result may be located anywhere on the map or screen and, thus, needs to be highlighted in some other way to the user if desired. As a consequence, map visualizations might not be beneficial for non-exploratory types of search patterns, e.g., when users perform fact-finding tasks often only the first few search result are of interest.

In this thesis, alternative means of visualizing the rank or relevancy score of search results are not investigated. It is assumed that the omission of the relevancy score does not have any significant negative impact on the user's effectiveness in exploratory search scenarios.

Challenge 6 (semantic axis intuition)**:** Furthermore, based on user studies presented in later sections, user often struggle with the notion of a 2-dimensional map representation that does not follow a traditional axis-based diagram visualization. In other words, users often expect a direct mapping between the x- and y-axis of a map visualization and two specific features, e.g., the relevancy score or any object feature. Although it would be possible to only consider map-based visualizations with specific x- and y-axis semantics, this limitation would considerably restrict the ability to represent complex high-dimensional relations between objects visualized on a map. In this thesis, map visualizations based on specific x- and y-axis semantics are considered a special case of the more general problem of arranging high-dimensional objects. If for some reason objects can be reduced to a 2-dimensional feature space, e.g., in case users manually select two features, a traditional axis aligned visualization of these two features should be a natural outcome of the more general mapping approach suggested in this thesis. This fundamental problem of users not being familiar with the general idea of map-based visualizations of objects has been proven to be a significant initial hurdle for many users.

In Sections 11.2 and 11.3, prototypes and user studies will address this issue by providing introductory instructions to the user.

8.2 ALGORITHMIC CHALLENGES

Besides challenges from the area of user interaction design, there are also a number of algorithmic challenges to consider. In the following, relevant challenges are introduced in an informal way. A mathematical formalization of each challenge is presented in later sections.

Challenge 7 (projection error)**:** When generating 2-dimensional arrangements of non-trivial objects, their complexity often makes it necessary to omit or suppress parts of their properties, ideally, those that are currently not relevant to the user. This loss of information is expressed as a so called projection error, which is the error that occurs when trying to reconstruct the original objects from only their 2-dimensional position. For example, an image captured by a camera corresponds to a 2-dimensional projection of a 3-dimensional environment stripped by depth information. Similarly, high-dimensional objects are stripped by certain information that is not preserved by their 2-dimensional projection.

In addition to the conceptual projection error that is the result of the fact that a 2-dimensional space is less expressive than higher-dimensional spaces, many projection methods are based on iterative algorithms and only determine suboptimal solutions, which involve additional errors.

Section 4.2 provides a detailed formalization of dimensionality reduction methods and projection errors. Later on, in Section 9.2, multiple methods are analysed for their projection errors when projecting a growing collection of music albums.

Challenge 8 (user adaptability)**:** Although some information may get lost when mapping objects, users may not even require all information all the time. Depending on the search task, the user's interests and preferences, some information about objects can be safely omitted without disrupting the search or exploration process. Accordingly, projection methods need to be able to integrate prior knowledge that somehow describes which information is supposed to be preserved, and which is not.

Throughout this thesis multiple methods for personalization are discussed. In Section 10.4, user preferences can be specified as feature weights, which will then be considered when projecting objects. In Chapter 12, a method is proposed that allows to personalize maps through direct manipulation, i.e., moving of example objects. Finally in Section 11.4, first results of a user study are presented, which aims at personalizing the process of search and exploration based on gaze and pupil tracking.

Challenge 9 (projection consistency)**:** Furthermore, many projection methods are not computationally stable, in the sense that different projections are generated for similar sets of objects. Some methods even rely on random initialization, which will result in a new projection for each run. However, users would strongly benefit from map visualizations that are stable even when the underlying data changes over time. For example, clusters that have been identified by users in previous search or exploration sessions should be positioned at the same location in the following session, such that user can transfer the knowledge they have gained.

On the other hand, projection methods need to be able to integrate new clusters that are later added to the object collection, such that the overall map structure and map semantic is not compromised. This balance between stability and flexibility is considered to result in a consistent map, which is both reasonably stable and also able to integrate new clusters.

In Section 9.2 projection methods are evaluated for their consistency when projecting a growing music album collection.

Challenge 10 (computational complexity)**:** Last but not least, many projection methods suffer from high computational complexity, meaning the costs for calculating a projection often grows quadratic with respect to the number of objects that are being projected. Due to this high computational cost, often either only small maps can be effectively generated, or algorithms rely on simple approximation strategies, which involve high projection errors.

In Chapter 10, computational costs are considered by projecting only a small number of objects at a time. However, multiple local maps are aligned in order to create the impression of panning a large map.

In summary, there are a number of algorithmic and user interaction challenges, which are important factors when designing information systems that use map-based visualizations for supporting users during exploration and exploratory search tasks. In the remainder of this thesis, all of these challenges are adressed again in varying degrees of detail, potential approaches are suggested and evaluated.

9 PROJECTION METHODS FOR MAP VISUALIZATION

As discussed in the previous section, there are a number of both user interaction and algorithmic challenges that need to be tackled in order to develop an effective map-based exploration approach. Finding a suitable projection method is considered a central problem, since it has a high influence on many of these challenges, e.g., in terms of how well structure information is conveyed to users, whether user-specific preferences can be integrated, and the overall computational costs of the projection. A short general introduction to various dimensionality reduction methods can be found in Section 4.2. In the following, state-of-the-art projection algorithms are compared with respect to their suitability to be used as part of an exploration system. Section 9.1 discusses various dimensionality reduction algorithms and highlights advantages and disadvantages. In Section 9.2, a selection of projection methods is further studied by both measuring the projection error and gathering user feedback on a real world growing music collection. Finally, Section 9.3 discusses the problem of overlapping thumbnails and proposes a simple greedy grid regularization suitable for web-based applications.

9.1 COMPARING PROJECTION ALGORITHMS

There are a great variety of projection algorithms in the literature. In this section, a selection of relevant approaches are discussed in detail. This includes principal component analysis (PCA), two variants of multidimensional scaling (MDS), three variants of stochastic neighbourhood embedding (SNE), as well as growing self-organizing maps (GSOM) and autoencoders. In order to qualitatively compare these projection methods, various criteria are considered.

Projection Value

The *projection value* describes the usefulness of information preserved by a projection method for the purpose of exploration. For example, a linear projection may be considered of low projection value, since it is not able to capture high-dimensional non-linear relations, which often occur in real-world information spaces like document collections. Of course, the value of a projection is highly user-specific and task dependent. Even linear projections might be suitable in certain scenarios. Therefore, general conclusions can not be drawn. However, projection algorithms can be compared based on their fundamental approach, e.g., whether they preserve local structure, global structure, clusters or patterns.

Sensitivity to Changes

Another criteria is an algorithm's *sensitivity to changes* in contrast to *projection consistency*. A projection method that is sensitive to changes suffers from the problem that calculated maps may significantly differ even for small changes in input parameters, e.g., changes in the data or parameters of the algorithm itself. Highly sensitive projection methods are considered unfavourable for the purpose of map-based exploration unless only a static map projection is shown to the user. However, in many scenarios data changes over time and algorithm parameters are dynamically adapted based on user preferences. In these scenarios, users are likely to get confused by being presented structurally different maps given similar input parameters. In contrast, projection methods that are less sensitive to changes will generate structurally similar, and thus, more consistent maps such that users benefit from being able to transfer knowledge from one map to the next. In the worst case, an algorithm may be non-deterministic, i.e., it involves random processes. In this case, projections may differ even for the exact same input parameters. A detailed investigation evaluating the stability of projection methods is presented in Section 9.2.

Adaptability

Since exploration is a highly dynamic process which is influenced by many factors, e.g., the prior knowledge or preferences of a user, it benefits from *adaptable* projection approaches. An adaptable projection method allows to integrate additional constraints or preferences such that the resulting map is tailored to the current exploration scenario and user. However, adaptability can not be viewed as a binary feature. Instead, various projection approaches provide different possibilities to integrate preferences or prior knowledge to a different extend, which in turn involves various advantages and disadvantages. Therefore, the adaptability of projection methods needs to be evaluated for each method individually.

Projection Model

Another facet of projection algorithms concerns the way a projection is calculated. Some methods build a projection model that describes the projection function. Generating a separate projection model has the advantage that new data can be easily projected or used to incrementally refine the model, which can be implemented as a streaming approach. Stream-based model learning allows to process very large data sets in comparison to other approaches that require that all data is held in memory during the calculation of a projection. Other methods do not build a model, and instead only calculate a single projection of the given data points. In this case, data changes may require a full recalculation of the map projection.

Iterative Optimization

Some projection methods are based on a closed form expressions, while others are based on *iterative optimization* algorithms. Closed form approaches have the advantage of finding a global optimum of the error function, which has the potential benefit of generating both accurate and stable projections. In contrast, iterative methods subsequently refine solutions, but have a change of getting stuck in local optima. Furthermore, they require a stop criterion, which is usually defined as a threshold on the amount of improvement over a pre-defined number of iterations. Then, there is no limit on the amount of iterations required to find a solution. This inconsistency can be disadvantageous when applied to interactive information systems, since users might not be willing to wait for a long time until the final projection is available.

Incremental Improvement

Some projection methods allow to integrate new data to *incrementally improve* previous projections. Incremental improvements have the potential to reduce computational costs, since the optimization process starts with a solution that has already been optimized for similar data. Additionally, incremental map projections could be visualized as an animation to users, which could lead to less confusion if the structure of a map changes over time. A detailed discussion how projections methods can be adapted to support incremental improvements in the context of a growing music collection can be found in Section 9.2.

Computational Costs

Finally, an important factor are the computational costs of a projection method. Higher costs require either longer waiting times or limit the maximum number of objects that can be projected in a given time frame. Unfortunately, most projection algorithms are computational expensive, and therefore, large scale real world data sets, which often consist of many million or more information objects, can not be projected in reasonable time on modern hardware. However, there are still significant differences in computation costs between various projection methods.

Principal Component Analysis

In the following, a number of projection methods are discussed in regard to the aforementioned comparison criteria. A classic projection method is principal component analysis (PCA). Due to its linear embedding, PCA is considered to have low projection value, since it can not capture high-dimensional non-linear manifolds. It is partially stable, i.e., it is less sensitive to data changes than other approaches, but may produce a structurally different map if the order of principal components changes. An advantage of PCA is that it creates a model that can be applied to new data. For two-dimensional projections, its computation complexity increases quadratically with the number of items being projected. Unfortunately, in its classic form, the projection of PCA can not be incrementally refined and only be adapted through simple feature weights. However, since PCA has been studied for a long time, there exist many variants that try to tackle these problems. There exist incremental algorithms and kernel variants, which allow to also capture non-linear relations [47]. Furthermore, by choosing appropriate kernels or kernel parameters, projections can be adapted to exploration scenarios or user preferences. Unfortunately, finding suitable kernels and parameters is an open research problem and requires additional design effort for each data set or exploration scenario. This is the main reason why PCA is not further investigated throughout this thesis. However, other researchers suggest Kernel-PCA to explore, e.g., image collections [15].

Classical MDS

In comparison, classical Multidimensional Scaling (MDS) allows to find non-linear embeddings, and thus, its projection value can be considered higher than classic PCA. An advantage of classical MDS is that there is a closed form non-iterative solution, which has the benefit of a global optimization. As a consequence, the time required to calculate a map projection is stable with respect to the input parameters and allows to implement fluid user interactions. In contrast, incremental algorithms rely on a dynamic stop criterion. Of course, stop criteria can also consider the maximum available time, but have to sacrifice projection accuracy if no optima is found within a given time frame. Another advantage of MDS is that it only requires pairwise distances between high-dimensional objects as input. For many high-dimensional information spaces, e.g., document or image collections, there already exist established definitions of distance and similarity. In comparison to PCA, this limits the required design effort to adjust an exploration approach to new information spaces or exploration scenarios. Furthermore, distance functions can be easily adapted to user preferences, e.g., by feature weighting or metric learning, see Section 4.5. Similar to PCA, the computational costs of MDS increase with the quadratic number of objects to be projected. A disadvantage of classical MDS is that all pairwise distances have to be held in memory, which drastically limits the maximum number of objects that can be projected. Another disadvantage is that classical MDS is very sensitive to data changes. Even minimal changes can result in different maps. However, as discussed later, maps mainly vary by rotation and reflection, but share the same global structure. Additionally, rotational and reflective invariance can be resolved through Procrustes analysis as suggested in Section 9.2. Similar to PCA, there exist multiple variants of MDS, e.g., metric MDS [book:4], which provides an incremental algorithm.

Landmark MDS

Another variant is Landmark Multidimensional Scaling (Landmark MDS) [25]. Landmark MDS applies classical MDS to a subset of objects called landmarks, and projects the remaining objects by a linear embedding based on these landmarks. This reduces computational costs at the expense of projection accuracy.

Stochastic Neighbourhood Embedding

Besides PCA and MDS, there is a third projection approach called Stochastic Neighbour Embedding (SNE), see Section 4.2. SNE provides a non-linear embedding, which again can be rated to have high projection value. In comparison to MDS, which focuses on preserving similarities between both neighbouring and distant objects, SNE aims at preserving local neighbourhood relations only, i.e., local structure. As a consequence, MDS and SNE generate differently structured map projections. Unfortunately, it is not clear which of these two approaches is more suitable for what data or exploration scenarios. In the following Section 9.2, both methods are compared based on objective quality measures and user feedback in the context of exploring growing music collections. Similar to MDS, SNE relies on a definition of a distance function, which can be adapted by feature weighting or metric learning. SNE uses an iterative optimization algorithm and can be initialized either randomly or using previous projections. SNE is highly sensitive to data changes. Due to the incremental optimization process, even small changes can lead to differently structured maps. Therefore, in comparison to PCA and MDS, SNE can be considered the least consistent projection method. Besides, SNE requires to specify an additional hyper-parameter, which roughly describes how many neighbours are considered for the projection. In practice, multiple runs with different hyper-parameter settings are necessary in order to generate optimal map projections. This is, in addition to the discussion provided in Section 9.2, the main reason why SNE and its variants were not considered for the proposed exploration approach suggest in this thesis. Again, many variations of SNE have been proposed. The Neighbourhood Retrieval Visualizer (NeRV) provides an extension to SNE, which weighs both precision and recall of preserved neighbourhood relations in order to generate more balanced embeddings. Similarity, t-SNE provides improvement embeddings by modelling neighbourhood relations via student-t distributions. However, the main disadvantages of SNE apply to both NeRV and t-SNE, too. The computational costs of SNE, NeRV and t-SNE increase quadratically with the number of objects being projected. However recently, Maaten [149] proposes an improved optimization algorithm that reduces the complexity to $\mathcal{O}(n \log n)$, with n being the number of objects. As a consequence, t-SNE has gained popularity and is used by some exploration systems, e.g., [118, 126].

Neighbourhood Retrieval Visualizer

t-SNE

Growing Self-Organizing Maps

Another projection method that generates non-linear embeddings of high projection value is Growing Self-Organizing Maps (GSOMs). GSOM are a type of artificial neural networks. An introduction can be found in Section 4.2. GSOMs generate projections guided by data density and its two-dimensional grid structure of nodes. Unfortunately, the projection value of GSOMs can not be easily compared to MDS and SNE approaches. It is not clear, which approach yields better results in what exploration scenarios. However, Section 9.2 compares GSOMs, MDS and SNE using objective quality measures and user feedback. In contrast to previous approaches, GSOMs may assign several objects to the exact same two-dimensional coordinates, which would cause overlappings if multiple objects of the same node are visualized at the same time. As a consequence, an additional strategy that resolves these overlappings is needed. GSOMs are based on an iterative algorithm that is initialized randomly. Thus, GSOMs can get stuck in local optima and may require multiple runs. In addition, there are several hyper-parameters: the initial size of the map, the learning rate, growth strategy and growth threshold. All of these factors result in a high sensitivity to changes. An advantage of GSOMs is that a separate projection model is learned, which supports incremental improvement. New data can be added, removed or changed and an existing map can be updated without starting from scratch. The map generation process can be adapted to user preferences through feature weighting, but also by interactively assigning objects to specific nodes as demonstrated in [137]. Computational costs can be summarized as high and depend on multiple factors: the number of iterations, the learning rate, the stop criterion, the size of the map and growth parameters. In practice, several trials with different hyper-parameters and many optimization iterations are required, which limits the applicability of GSOMs to very large data sets. GSOMs have been previously used for exploration of, e.g., music collections [137].

Autoencoder

Lastly, autoencoders are another type of artificial neural networks that can be used to generate non-linear embeddings of high projection value. Since GSOMs and autoencoders are based on the foundations, they share many attributes. However, autoencoders achieve a dimensionality reduction through information compression and reconstruction. Still, further research is needed to compare the projection quality of various approaches in context of exploration and exploratory search. Autoencoders use an iterative algorithm based on gradient descend, which is initialized randomly. As a consequence, the optimization can get stuck in local optima. Therefore, it is highly sensitive to changes. In addition, there are many hyper-parameters, e.g., the network architecture, learning rate and stop criterion. Similar to GSOMs, a projection model is constructed, which can be incrementally refined and used to embed new data points. The learning process can be adapted to user preferences by feature weighting and additional custom cost functions, e.g., deduced from direct map manipulations as suggested in Chapter 12. Autoencoders suffer from high computational costs, which depend on multiple factors, e.g., the size of the network, learning rate and stop criterion. However recently, gpu-accelerated software libraries have been developed that significantly speed up the optimization process. At the time of evaluating various projection methods in context of exploring growing music collections as discussed in Section 9.2, these advances were not yet available, which is the reason why autoencoders were not included in the evaluation. Still, Chapter 12 discusses autoencoders as a method to generate personalized maps.

Conclusions

In summary, all approaches suffer from high computational complexity, which limits the number of objects that can be projected in a reasonable amount of time. As a consequence, generating global maps of large information spaces that are adaptable in an interactive exploration scenario is considered computational infeasible. GSOMs and Autoencoders allow to learn a separate projection model, which can be incrementally refined and adapted to user preferences in a interactive exploration scenario, see [137] and Chapter 12, respectively. Other approaches are adaptable by classic feature weighting or metric learning, but require a full recalculation of the projection. PCA is excluded from further investigations due to its low projection value or high design effort required to adjust projection parameters to different exploration scenarios and information spaces. MDS shines with its closed form expression yielding globally optimized solutions. However, it needs to be extended in order to generate consistent maps when data changes, see Section 9.2. SNE, NeRV, t-SNE shine through their high projection value, but may lead to user dissatisfaction due to their dynamic stop criteria and inconsistent incremental optimization often requiring multiple trails to generate optimal projections. In the following, projection methods are further compared based on objective quality measures and user feedback in the context of growing music collections.

9.2 PROJECTION STABILITY FOR GROWING COLLECTIONS

As discussed before, most projection algorithm are not specifically designed to generate consistent projections when the underlying data collecting changes. Some are based on incremental learning approaches, which can be resumed when training data has changed. Still, there is the problem of balancing stability, i.e., that items are projected at the same position as before, and flexibility, i.e., that the overall map structure is updated when new clusters are being added. The following work will address this problem, see Challenge 1 (conveying structural information). It was published in [pub:4].

Aim

The aim of this work is to evaluate several map projection algorithms with respect to their suitability for generating comprehensible map updates when new items are being added to the data collection. Ideally, the map structure is supposed to be preserved as much as possible. Any necessary structural changes, which could be the result of new clusters being added, should be implemented with as little as possible changes to the arrangement of objects. Furthermore, a user study evaluates whether users are able to track incremental map updates such that they still perceive the updated map as familiar and are able to transfer knowledge from the previous arrangement of objects.

Related Work

At the time of publication, in 2013, related work focused on the improvement of dimensionality reduction methods in terms of projection quality. Kanth et al. studied the effect of degrading projection accuracy on similarity search in dynamic databases [115]. Other work focused on improving the usefulness of map projections. For example, in [137, 65] user feedback is integrated in the map generation process. Knees et al. [70] proposes to enrich maps with a landscape analogy of islands. And Stober et al. [139] allows to find high-dimensional neighbours in two-dimensional maps through a multi-focus zoom lenses. However, recently the evaluation of dimensionality reduction methods for dynamic data has gained more traction. Rauber et al. [114] proposes an adaptation of the stochastic neighbour embedding (SNE) algorithm discussed below, which integrates an additional cost parameter in order to balance projection consistency and accuracy. Instead, the proposed approach discussed below simply applies the projection algorithm using previous coordinates as initialization for generating a new map, which corresponds to strategy 3 in the evaluation of [114]. In both [119] and [154] a similar objective evaluation of various dimensionality reduction methods on different data sets are presented. Unfortunately, both studies focuses on more recent variants of SNE algorithms, such that direct comparisons are not possible. Besides, other data sets are used. Although in [154] the volatility of map changes is visualized through point trails, the authors do not involve users in order to better understand whether they are able to still follow those changes and are able to transfer knowledge from one map to the next.

Quality Measures

In order to evaluate algorithms on an objective level, both the overall quality of a projection as well as the relative change between projections needs to be measured. The overall projection quality is rated by how well neighbourhood relations are preserved. For this purpose, the measure of trustworthiness and continuity are used as described in [67]. An introduction can be found in Section 4.2.1.

In order to evaluate the quality of relative change between projections, the mean positional change is measured. It is defined as the average of the Euclidean distances between previous and updated item coordinates in the 2-dimensional projection. For normalization, the projection coordinates were adjusted to fit inside a unit square. A high mean positional change is assumed to cause users to get confused about how the map changes when adding new items. A low mean positional change corresponds to a relatively stable map and is more likely to allow users to transfer knowledge from previous item arrangements.

Ideally, projection algorithms should achieve high trustworthiness even when adding many new items, but require only few small positional changes.

Projection Algorithms

In the following, five different projection algorithms are compared: multidimensional scaling (MDS) [75], landmark multidimensional scaling (Landmark MDS) [25], growing self-organizing maps (GSOM), stochastic neighbour embedding (SNE) [56] and neighbour retrieval visualizer (NeRV) [153]. An introduction to each algorithm is given in Section 4.2. Since most algorithms are not designed to support incremental updates of projections, the following modifications are implemented.

In case of MDS, projections are generated from a pairwise distance matrix, which does not allow to include prior knowledge about previous projections. Any changes to the input matrix might result in a completely different projection. In practice, similar input matrices will often result in projections of similar structure. However, MDS does not normalize a projection in terms of its orientation. Therefore, Procrustes analysis [43] is used to align subsequent maps by translation, rotation, reflection and scaling without changing relative distances between objects.

In case of Landmark MDS, objects of a previous projection are all used as landmarks of subsequent projections. Therefore, objects will stay at the same position once added to the map. In the following experiments, the mean position change of Landmark MDS is expected to always be zero. Of course, with increasing size of the collection, the overall map structure will suffer significantly, since new clusters can not be well represented. Thus, it is expected that the measures of trustworthiness and continuity will decrease when adding new items to the collection. In comparison to MDS, which always generates a new projection from scratch, and thus, prioritizes map trustworthiness, Landmark MDS can be viewed as the counter approach, which prioritizes map stability.

GSOM directly supports growing data collections by adding new cells to its map. For initialization, a hexagonal grid of 2×2 cells was used. In order to avoid that many objects are added to the same cell, which would lead to overlapping items in the final visualization, a low threshold for growing new cells was chosen. Additionally, the final coordinates of objects were calculated as a distance-weighted mean of the cell's coordinates and its direct neighbour cells, which further reduces the problem of overlappings. In contrast to MDS, GSOM requires objects to be represented as vectors. As recommended in [138], MDS can be used for vectorization while preserving neighbourhood relations. Unfortunately, since not all objects are available at the start, an incremental vectorization approach has to be used. In the following, objects are vectorized using Landmark MDS.

SNE and NeRV do not specifically support growing data collections. However, they both can be initialized with coordinates from previous projections. Assuming that only relatively few new item are being added, it is likely that the projection will not change very much and stays structurally similar to the previous one. However, there is no guarantee since the optimization process will not directly consider previous coordinates.

Objective Evaluation on Growing Album Collections

For evaluation, a collection of 180 songs from 12 albums of the band *The Beatles* was chosen. The albums were added in order of their release date one album at a time. Pairwise distances between songs

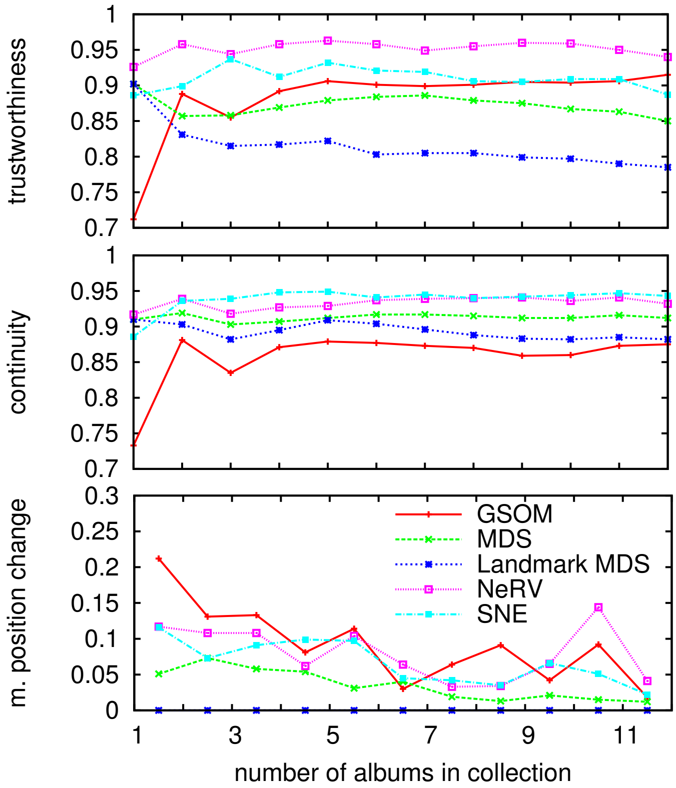


Figure 14: Progression of projection quality measures for each algorithm when adding new albums to the collection.

were computed as the weighted sum over three state-of-the-art audio content features: Mel Frequency Cepstral Coefficients (MFCCs) [86], fluctuation patterns as described in [103], and chord frequency distributions extracted from ground truth annotations, see [51].

The neighbourhood parameter was chosen to $k = 5$ for both the quality measures trustworthiness and continuity, as well as NeRV and SNE. In case of GSOM, the vectorization dimensionality using Landmark MDS was experimentally determined to $d = 13$ dimensions.

Figure 14 illustrates the progression of all quality measures when adding new albums to the collection one at a time. Based on trustworthiness alone, NeRV clearly achieved the best results. Although SNE is specifically designed to optimize continuity, it could not significantly outperform NeRV in terms of continuity. As expected, projections generated by Landmark MDS suffer from decreasing trustworthiness due to its focus on map stability. MDS achieves average results for both trustworthiness and continuity.

When taking the position change into consideration, MDS achieved a very good results considering that it does not use any information about previous projections except for alignment using Procrustes analysis. Both NeRV, SNE and GSOM struggle with inconsistent mean position change ratings, suggesting that projections are restructured to a greater amount from time to time.

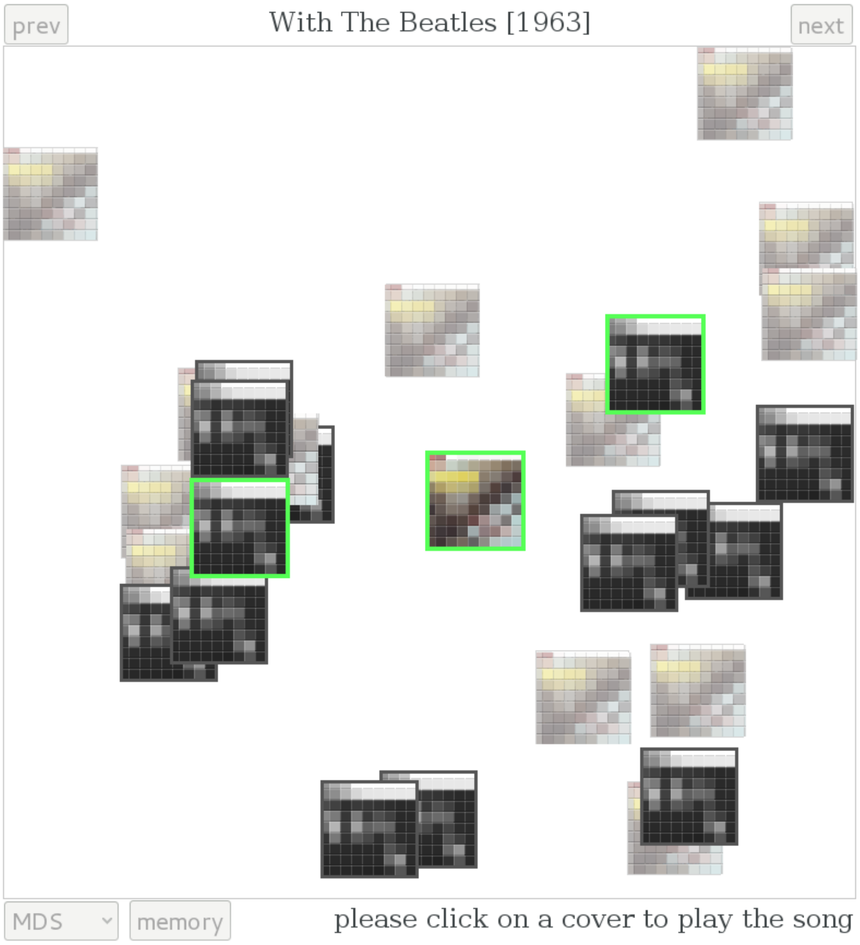


Figure 15: Screenshot of prototype for evaluating projection algorithms for a growing album collection. Each cover image represents a song of that album. Songs that are supposed to be remembered are highlighted with a green border. Cover images blurred due to licensing concerns.

Overall, NeRV is believed to produced the best projections. However, it remains to be seen whether the amount of change between subsequent projections is still acceptable for the purpose of map visualizations, and whether users are able to transfer knowledge from one map to the next. For that reason, the following user study was conducted.

User Study on Growing Album Collections

The aim of the following comparative user study is to investigate how well users are able to follow position changes of individual items, such that they can still transfer knowledge about the structure of the previous map to the next. In order to measure the participants ability to track map changes, a gamified approach was implemented in form of a simple memory game.

Figure 15 shows a screenshot of the prototype. It allows to visualize songs of one or multiple albums in a map visualization. Coordinates for each song were generated using the algorithms described before.

With each new album, more and more songs were added to the visualization and coordinates were updated accordingly. Songs are represented by the cover image of the respective album. That means, songs of the same album could only be visually differentiated based on their position on the map. Selecting a song shows a label of the song name and starts playing the song.

The study was designed as follows. First, participants were interviewed for demographic information, their prior knowledge of general computer skills and map-based visualizations, as well as their knowledge about *The Beatles*. Then, a lab experiment was conducted, in which each participant was asked to perform the same task for each projection algorithm. In order to reduce workload, only MDS, GSOM and NeRV were compared. A Latin square design ensured that there would be minimal bias with regards to the order in which participants were solving the tasks. Before the actual memorization task was started, participants could briefly interact with the system in order to get familiar with the map-based visualization design.

The memorization task consists of two steps. First, three random songs were highlighted for five seconds as illustrated in Figure 15. Then, an animated transition between the current map and the next map was shown. Afterwards, participants were asked to select the same three songs previously highlighted. For this step, no time constraint was given. In case participants selected a song not previously highlighted, the selection was counted as an error.

It is hypothesized that the error count of such a memorization game reflects how well participants are able to follow map changes. Since songs of the same album are represented by the same cover image, participants were required to visually track each of the three songs while they were moving on the screen. Thus, high error counts should signal that participants struggle following map changes, and thus, can not identify the same songs previously highlighted. If each song would have been represented by a unique cover image, participants could have instead memorized each cover image, and would not have needed to track their changes in position. At the end, participants were asks about their personal opinion regarding the overall projection quality of each algorithm.

Study Results

In total, 19 participants between the age of 23 and 38 years (28 on average), 8 women and 11 men, have completed the user study. Most participants (68%) were students or doctoral students in computer science. Some (21%) were well acquainted with the music of The Beatles. About half (47%) were familiar with map visualizations.

Figure 16 shows the progression of selection errors for each projection algorithm averaged over all study participants. Based on the overall trend illustrated Figure 16, the more songs are added to the collection, the more participants made selection errors. This can be attributed to the fact that all songs are visualized in the same screen

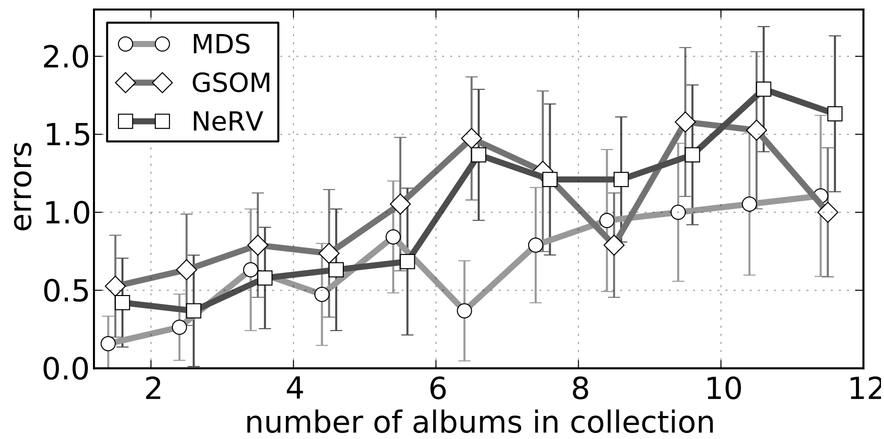


Figure 16: Progression of average selection errors for each projection method. With increasing size of the collection, more and more selection errors are made by study participants.

space, and overlappings are more likely when many songs are shown at once. Although a high error variance can be observed, MDS has clearly resulted in the least amount of selection errors. On average over all interactions, participants made 0.69 errors ($\sigma = 0.85$) for MDS, 1.03 errors ($\sigma = 0.93$) for GSOM and 1.02 errors ($\sigma = 0.98$) for NeRV. These results coincide with the preferred choice of participants. Out of 19 participants, 12 chose MDS, 5 chose NeRV, one picked GSOM, and one picked both MDS and NeRV as their favourite projection algorithm. Furthermore, participants often described MDS to result in less positional changes, NeRV to better preserve cluster structures and GSOM to have less overlappings, which conforms to the results concluded from the objective evaluation discussed before.

Summary

In summary, based on both the objective measures and user study, MDS in combination with Procrustes analysis showed the most promising compromise between projection quality and projection stability. Although NeRV generated more accurate projections, participants favoured MDS due to its higher stability. Of course, since both evaluations were performed for only a single dataset, results can not be easily generalized to other datasets. In the future, additional measure could be tested. For example, the mean position change does not consider the uniformity of item movements. It can be hypothesized that users would perceive certain uniform movements less distracting than seemingly random movements, e.g., movements that resembles a perspective zoom of the map. Accordingly, a measure could be designed which is invariant to certain uniform movements, or focuses on relative position changes amongst items. Furthermore, algorithms could be adapted to specifically balance projection quality and position changes, e.g., by introducing additional components to their cost functions.

9.3 GRID REGULARIZATION OF MAP-BASED PROJECTIONS

Apart from an accurate and stable projection, there are many more challenges related to a map-based projection. For example, a map is not useful if users can not identify individual objects in order to find patterns or clusters. When visualizing a map as a simple scatter plot, where each item is represented as a dot, density information is illustrated and users will be able to discover clusters and outliers. However, users can not form concepts about the meaning of a cluster, pattern or outlier. Also, users can not easily interact with dots of a scatter plot, e.g., in order to select an individual item. Therefore, it is vital that a subset of items is described in more detail and with sufficient spatial dimension, e.g., by a label or thumbnail image. Unfortunately, not all challenges can be covered by this thesis. In the following, neither the problem of finding a good selection of representative items, nor the challenge of finding a good visualization for these items is discussed.

Aim

Instead, the goal of this work is to provide a method that prevents overlappings in map-based visualizations that illustrate each object by a rectangular icon. Most projection methods do not consider objects to have any spatial dimension, shape or size. Therefore, overlappings of icons is a common problem when displaying objects that are projected too close to each other. Users benefit from a map visualization without overlappings, since more information about individual objects can be shown in a single view. On the other hand, manipulating map projections for the benefit of a more readable visualization will introduce additional projection errors. Therefore, a method needs to be found that minimizes additional projection errors and prevent overlappings between object icons.

Related Work

The problem of overlappings in map-based visualizations has been investigated before. A common strategy is to assign items to a fixed grid of cells, where each cell is large enough to fully enclose each icon. Naturally, no overlappings remain. However, a sufficient number of grid cells needs to be available. Then, an algorithm has to be found, which assigns items to grid cells such that the projection error remains minimal. Several of such approaches are discussed in chapter 5 of [4], including a naive greedy assignment, as well as a genetic algorithm. Liu et al. [80] propose an sub-optimal assignment algorithm based on sorting index sets. However, non-grid-based approaches exist as well. Reinert et al. [116] implement a GPU accelerated iterative algorithm that distributes items of different size and shape with the goal to achieve even distances between the boundaries of all items.

Setting

For the envisioned prototypes, which are described in detail in future Sections 10.4, 11.1, and 11.3, multiple surrounding conditions had to be met. These conditions are the result of an incremental evolution of prototype implementations, and thus, reflect various design decisions that had been made in order to maximize the overall usability of the prototype given limited time and resources. For example, for ease of implementation, all object icons are visualized with the exact same shape and size, showing information at the same level of detail. However, in other scenarios, it could be more reasonable to visualize some objects in greater detail than others, e.g., if there is prior information about the user's information need. Also, since the prototype is aimed at non-scientific exploration scenarios, e.g., the exploration of media collections, preserving exact pairwise distances when visualizing objects on a similarity map is not considered essential, as long as the local and global cluster structure remains intact. In addition, the prototype is supposed to be experienced by a wide variety of users on common hardware, i.e., personal computers, tablets or smartphones. Therefore, the prototype is implemented as a web application, which requires that the map visualization is adapted dynamically inside the user's browser, since relevant factors depend on the current browser configuration, e.g., the available screen size and font size.

Overall, a regularization method needs to be found, which prioritizes the following criteria in descending order: most importantly, it prevents overlappings between object icons; secondly, it can be applied live inside a browser window within an acceptable runtime; also, it preserves local and global cluster structure of the map projection; and lastly, it minimizes additional projection errors. Unfortunately, the requirement for an highly efficient algorithm suitable for browsers rules out all of the previously discussed related algorithms except for simple naive ones. Therefore, a naive greedy assignment algorithm was implemented, which follows a similar procedure as *Greedy2* in [4].

Greedy Grid Regularization

Given a fixed grid of cells such that each cell can fully enclose a given rectangular shape of fixed size, each object is subsequently assigned to its closest grid cell that is not yet occupied by another object.

Due to the incremental assignment of objects, some objects may not be placed at their respective closest cell, in case it is already occupied. Therefore, the algorithm will not find an optimal assignment of objects that minimizes projection errors. This is especially problematic if the number of grid cells is equal or only slightly larger than the number of objects. Objects that are assigned last will potentially have to be positioned far away from their true projected position. Thus, the order in which items are assigned can have significant influence on the overall result. However, experiments reveal that for sufficiently large grids, i.e., grids which have significantly more grid cells than the number of objects that are supposed to be visualized, object positions as well as local and global clusters are preserved reasonably well for the purpose of map-based exploration of media collections.

Most importantly, this simple procedure can be easily implemented in a browser and allows to reposition objects live during an exploration session, even if the current browser size or font size is dynamically changed by a user.

Summary

In summary, a naive grid regularization method has been identified from a number of related approaches, which satisfies the requirements of a web-based exploration prototype. Based on user feedback gathered from several user studies described in future Sections 11.1, 11.2 and 11.3, the proposed grid regularization algorithm effectively preserves object positions and cluster structures for sufficiently large grids. Therefore, no further investigation was necessary. In the future, detailed comparisons between different approaches can help to identify algorithms that provide more optimal solutions, introduce less projection errors, and at the same time, are suitable for live layouting in modern browsers.

10 VISUAL BERRYPICKING BY ALIGNING CONSECUTIVE MAPS

As discussed in Section 9.1, projections methods that are suitable for map-based search and exploration suffer from a high computational complexity, and as a consequence, can not project large object collections in acceptable time or with reasonable quality. With increasing number of objects projections become inaccurate, i.e., dissimilar objects are projected close to each other (degrading *trustworthiness*), and similar objects are projected far apart from each other (degrading *continuity*). Furthermore, many projection methods require a full recalculation when user preferences change, which is often the case during a search and exploration process. Therefore, in this section, the influence of computational complexity and increasing projection error is limited by only considering a subset of the full object collection at any given time. Given a reasonably small subset, many suggested projection methods can be applied in an interactive setting, i.e., projections are calculated live during search sessions, which allows to integrate user feedback. On the other hand, by projecting only a subset of the data, users will not be able to observe all objects at the same time, and thus, will not be able to draw conclusions from the global structure of the data.

Aim

The aim of this work is to develop a method for effective map-based search and exploration given the limitation of only being able to project small subsets of the whole data collection to a local map. It is hypothesized that map-based visualizations of small subsets of search results are still sufficient to convey structural information in context of the current search query. Users may discover local clusters and, thus, may still be able to extract knowledge about the local structure of the collection, or be able to quickly filter irrelevant objects without scanning each single one.

Unfortunately, due to the limitation in size, a local map can not convey global structure, since it does not give an overview over the whole collection. However, it is hypothesized that users are able to draw partial conclusions about the global structure of the collection from a number of local maps. Ideally, when each local map corresponds to a zoomed-in area of a global map, their combination would yield the global map again. Hence, users might be able to extract knowledge about the global structure from a number of local maps. In the following, an interaction approach is suggested that targets this idea. It was published in [pub:5].

Related Work

As discussed in Section 3.2, there are a number of map-based exploration systems that allow users to interact with 2-dimensional projections of high-dimensional spaces. For example, in [139] users can explore a music collection by navigating in a global similarity-based map. However, projection accuracy is sacrificed by generating maps using a variant of MDS that approximates positions for all songs but a random selection of samples. Recent advances in projection methods, e.g., t-distributed stochastic neighbour embedding [149], can be used to generate large scale maps of millions of documents as described in [118]. However, almost all methods do not combine both search and map-based exploration such that user can benefit from their synergistic effect. Only a few approaches investigate the possibility to visualize search results as maps. Rubner et al. [123] proposes an iterative search in large image collections through consecutive query-by-example searches. Each result set is visualized by a similarity-based map in order to support users in identifying clusters and outlier. Rubner et al. also recognize the exploratory nature of such an iterative map-based query-by-example search, but do not further investigate. In the following sections, a similar exploration approach is proposed, discussed and evaluated in detail.

10.1 FROM TRADITIONAL TO VISUAL BERRYPICKING

As introduced in Section 2.2, in an exploratory search scenario the user's information need is usually not well defined and evolves during the search process. In case of keyword-based search system, users will perform a series of modified queries, subsequently refining their information need by extracting new information from currently relevant search results. This search pattern continues until the user's information need is fulfilled. The process is referred to as *berrypicking* [6].

The pattern of berrypicking can be directly applied to map visualizations, e.g., by visualizing a subset of keyword-based search results as a map using some similarity measure. Then, users can inspect each search result and modify the keyword-based query. However, subsequent maps of various keyword queries are unlikely to be correlated with respect to the similarity measure, since search results are sorted by their relevancy to query terms and not with respect to the similarity measure. As a result, users will struggle to connect multiple subsequent maps and may not be able to extract knowledge about the global structure of the collection.

Hence, this work suggests to adopt the pattern of berrypicking to the scenario of map-based search and exploration. Instead of query terms, the current search query is modelled by an example or *seed* object. Search results are generated based on a query-by-example

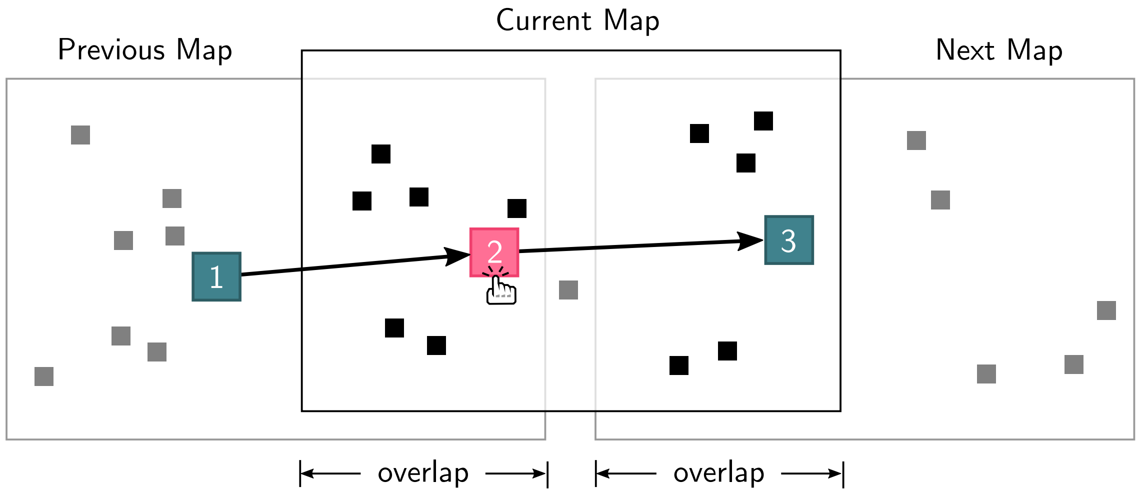


Figure 17: Illustration of map-based navigation by jumping from one local k-nearest neighbour map to the next. Maps are aligned based on common objects present in both subsequent maps, which ideally creates the impression of stable navigation directions.

search, which uses the same similarity measure that is applied when projecting objects. It is hypothesized that search results of a query-by-example search will represent a local neighbourhood of the collection, and thus, a map visualization of these search result will correspond to a local area of a global map.

Furthermore, users are enabled to modify the search query by selecting a new example or seed object from the current visualization of search results. Selecting a new seed object allows users to navigate the collection by quickly jumping from one local map to the next. Since subsequent queries are based on the same similarity measure, they are also likely to share search results. In contrast to [123], local maps are proposed to be aligned based on shared search results, such that their orientation matches previous maps. It is hypothesized that users are able to extract knowledge about the global structure of the collection when jumping between overlapping and aligned local maps, which might be perceived as panning a structurally stable global map. Figure 17 illustrates the concept of navigating local maps by selecting new seed objects. In reference to the traditional definition of berrypicking, this map-based navigational pattern is called *visual berrypicking*.

10.2 VISUAL BERRYPICKING USING K-NN AND MDS

As discussed before, the process of visual berrypicking consists of two steps: a query-by-example search and a map-based visualization of search results.

Query by Example

Given a similarity measure, a query-by-example search can be implemented by a k-nearest neighbour search. There are both exact and approximate methods that allow for an efficient nearest neighbour search in large collections, e.g, kd-trees [7] or Locality Sensitive Hashing [60]. An introduction can be found in Section 4.4. A downside of using k-nearest neighbour search is the emergence of hubs and orphans, i.e., objects that appear very frequently or not at all amongst the k-nearest neighbours of all other objects. The hubness phenomenon introduced in Section 2.4.2 and further discussed in detail in Section 10.3. Hubs can be viewed as objects of the collection, which will be present in many local maps. Therefore, hubs might help users in establishing a sense of orientation by acting as landmarks. Orphans, on the other hand, will correspond to objects that will not appear in any of the local maps generated from other seed objects. Hence, orphans can not be found by the means of jumping from one map to the next. Luckily, as identified in Section 11.3, in practice, only very few items are orphans of a movie collection.

Map Visualization

Visualizing maps can be achieved by a projection algorithm for dimensionality reduction. Based on the evaluation in Section 9.2, multidimensional scaling [75] has been shown to be a suitable mapping algorithm balancing both projection error and flexibility when adding new objects to the collection. Unfortunately, MDS is invariant with respect to rotation and reflection. As a consequence, consecutive maps might appear as a completely different maps to the user due to their random orientation, even when maps share many common objects. In order to avoid this problem, Procrustes analysis [43] is applied in order to align consecutive maps based on their common objects present in both maps. Procrustes analysis reduces the sum of the squared differences between pairs of objects positions by translation, scaling, rotation and reflection. Aligning consecutive maps in such a way allows to animate transitions between maps with minimal movement of common objects. Ideally, this transition is then perceived as panning a structurally stable map. Section 11.1 discusses results of a first user study evaluating whether users are able to effectively navigate a collection of artificial objects using k-NN search, MDS and Procrustes analysis combined in an initial exploratory search user interface prototype.

Reduce Alignment Error

In order to align consecutive maps, both of them have to share common objects. For small maps or small numbers of k-nearest neighbours, only very few objects may be present in both maps, limiting the ability to effectively align maps via Procrustes analysis. In order to make transitions between consecutive maps as consistent as possible, it is suggested to maximize the overlap between two consecutive maps by using higher values of $k_p > k$ nearest neighbours, even if only k objects are supposed to be visualized to the user. These

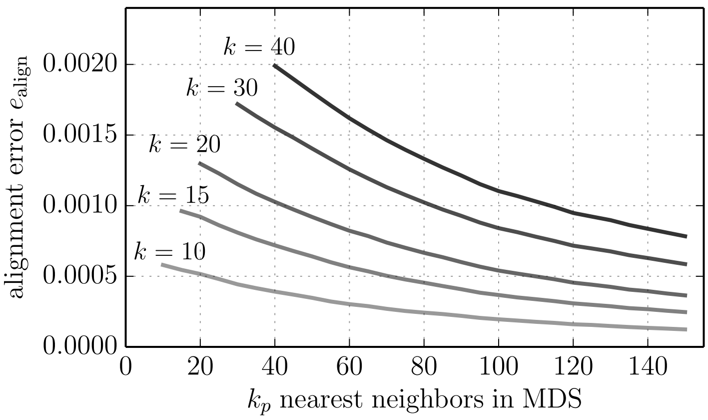


Figure 18: Mean squared alignment error between common neighbours and their respective positions in consecutive maps when using $k_p \geqslant k$ nearest neighbours for generating each map.

larger virtual maps of size k_p are expected to contain more common objects, and thus, are expected to be aligned more accurately.

In order to measure the quality of alignment, an alignment error is introduced. The alignment error e_{align} corresponds to the difference between the two-dimensional positions $m_i(j)$ of common nearest neighbours $j \in knn(i_{\text{prev}}) \cap knn(i_{\text{next}})$ in consecutive maps for seed objects $i_{\text{prev}}, i_{\text{next}}$ after alignment by Procrustes analysis.

$$e_{\text{align}} = \frac{1}{n} \sum_{j} (m_{i_{\text{prev}}}(j) - m_{i_{\text{next}}}(j))^2 \qquad (1)$$

A small alignment error means that the relative positions of common objects do not change much when transitioning from one map to another. However, due to the visualization of maps on a display of fixed size, common nearest neighbours have to be moved even when perfectly aligned (i.e. alignment error of 0). However, if both maps are aligned, this movement should correspond to a consistent movement of all common objects and thus, ideally, is perceived as panning a large structurally stable map. Furthermore, this benefit of continuity helps to transfer knowledge from one map to the next. For example, users might be able to recognize stable navigation directions throughout multiple interactions, and thus, may be able to infer information about the global structural of the data set as discussed in a later Section 11.1.

Figure 18 illustrates the mean alignment error when switching from one map to the next given the polygon image dataset presented in Section 11.1. For higher values of $k_p > k$ the alignment error consistently decreases. Thus, it is suggested to use higher values of $k_p > k$ for map generation in order to improve the alignment of consecutive maps via Procrustes analysis. Of course, as discussed before, both the computational costs involved in generating maps and additional projection errors caused by projecting a large number of objects to 2-dimensional space limit the maximum size of k_p.

Zoom

In contrast to global maps, local maps can not easily be zoomed out to generate a larger overview over bigger parts of the data set, which might allow users to get a better understanding of the global structure of the data set. However, some form of zooming out can be achieved by using a larger $k_z > k$ when searching for k_z-nearest neighbours, and reducing the resulting set of neighbours to k representative objects that are being visualized to the user. Choosing k representative samples should provide a good overview of the larger subset of k_z objects. This can be achieved by various methods, e.g., centroid-based clustering algorithms in case objects can be described via a vector representation, or by a clustering variant of the reservoir sampling method that requires only pairwise distances between objects as described in [35]. However, introducing non-deterministic random samples might confuse users since maps will not longer be consistent when navigating back and forth using the same seed objects, if possible at all. In the following Section 11.3, a prototype is presented that implements such zooming by using a deterministic strategy of greedy min-distance sampling, see [91]. Each of the $k_z > k$ objects is added greedily to a reservoir of k representatives one by one. Once the reservoir is fully populated, each new object will replace an object of the reservoir only if replacing it will reduce the minimum distance between all objects of the reservoir. This simple strategy will generate a set of k representative items. In contrast to clustering, which often requires algorithms of higher computational complexity, representative objects are not chosen based on population density, and outliers are favoured due to their high distance to other objects. However, a good balance between performance, stability and overview can be achieved.

User Feedback

User feedback can be integrated by adapting the distance metric used for the k-nearest neighbour search. In the simplest case, features of a Euclidean distance function can be weighted according to user preferences determined from direct input by the user. Since kd-trees do not require the similarity metric to be known at tree construction time, weights of a similarity metric can be adapted without additional computational costs such that k-nearest neighbours can be efficiently determined even when weights are changed dynamically by a user during a live search session. Alternatively to direct input from a user, feature weights could be estimated by metric learning. However, this requires additional user feedback in form of labeling, ranking or positioning objects in some way. Section 4.5 gives an introduction to metric learning. Once a new set of weights has been determined, a new projection can be calculated and immediately visualized to the user. Animations of object movements caused by weight changes may allow users to get a better understanding of how certain features are embedded in the map visualization. Again, maps generated based on different weightings are aligned via Procrustes analysis in order to achieve consistent transitions with minimal movements.

10.3 LIMITATIONS OF K-NN NAVIGATION

In the following, limitations of a k-nearest neighbour-based navigation in high dimensional information spaces are discussed. As introduced in Section 2.4.2, the hubness phenomenon describes the observation that in high dimensional data some objects appear much more often amongst the k-nearest neighbours than others. When using a k-nearest neighbour search for map-based exploration, the hubness phenomenon may have significant effects on user navigation patterns. For example, users will observe hubs in many more local maps than objects that appear less often in nearest neighbour lists of selected seed objects. In the worst case, when an object does not appear in any nearest neighbour list, it can never be found by means of jumping from one map to the next. This poses a significant limitation of the proposed exploration method of visual berrypicking and k-nearest neighbour search in general.

In the following, properties of the hubness phenomenon are studied using various artificial data sets in order to get a better understanding and potentially derive methods to avoid or mitigate said limitations. Initial results were published in [pub:3].

Hubness Measures

In order to study the hubness phenomenon in detail, a number of terms and measures need to be defined. Given d-dimensional data points $\vec{x}_i \in X \subseteq \mathbb{R}^d$ with $\vec{x}_i = (x_{i,1}, \ldots, x_{i,d})$ and $i = 1, \ldots, n$ for $n = |X|$, the k-nearest neighbours of a point $\vec{x}$ are defined as $N_k(\vec{x}) \subseteq X \setminus \{\vec{x}\}$ such that $\forall \vec{y} \in N_k(\vec{x}) : \forall \vec{z} \in X \setminus N_k(\vec{x}) \setminus \{\vec{x}\} : \delta(\vec{x}, \vec{y}) \leqslant \delta(\vec{x}, \vec{z})$, and $|N_k(\vec{x})| = k$. In case of distance equality, ties are broken arbitrarily.

The frequency of a point $\vec{x}$ appearing in k-nearest neighbour lists of all other points is called the *hubness* of this point, the *size of the hub* or in some literature k-*occurrence*. Formally, it is defined as $h_k(\vec{x})$:

$$h_k(\vec{x}) = \sum_{\vec{y} \in X \setminus \{\vec{x}\}} \mathbb{1}_{N_k(\vec{y})}(\vec{x}).$$

$\mathbb{1}_{N_k(\vec{y})}(\vec{x})$ represents the indicator function, which equals one in case $\vec{x} \in N_k(\vec{y})$, else zero. The Hubness Phenomenon can now be described as a significant skew of the distribution of the size of hubs $h_k(\vec{x})$ for all points $\vec{x} \in X$.

The *skewness* γ of the hubness distribution can be described as the third moment about the mean:

$$\gamma_X(k) = \frac{\frac{1}{n}\sum_{i=1}^{n}(h_k(\vec{x}_i) - \overline{h_k})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(h_k(\vec{x}_i) - \overline{h_k})^2\right)^{3/2}}.$$

Another measure can be defined as the size of the largest hub. However, since there are usually just a few very large hubs, taking the maximum would make this measure very sensitive to the data. Instead, in the following the average size of the largest q percent of all hubs is considered, hereby denoted as $M_X(q) \subseteq X$:

$$m_X(k, q) = \frac{1}{|M_X(q)|} \sum_{\vec{x} \in M_X(q)} h_k(\vec{x}).$$

Instead of the average, it would also be possible to use the more robust $(1-q)$-quantile directly. However, because of the integer nature of the hubness, which limits the number of possible values especially for low k, the average of the largest q percent is preferred. In addition, the $(1-q)$-quantile does not capture the distribution of values beyond it.

The following two measures are based on the graph that is induced by the k-nearest neighbour relations. The measure of $\text{coverage}_X(k, q)$ is defined as the fraction of points that contain at least one of the fraction q of data points with the largest hubness in their k-nearest neighbour lists:

$$\text{coverage}_X(k, q) = \frac{1}{n} \sum_{\vec{x} \in X} \left(\max_{\vec{y} \in N_k(\vec{x})} \mathbb{1}_{M_X(q)}(\vec{y}) \right)$$

This measure indicates the power of influence of the largest hubs. It is closely related to $m_X(k, q)$ as defined above. The sum in $coverage_X(k, q)$ effectively counts the size of all largest hubs in $M_X(q)$, unless two or more of the largest hubs are contained in a point's k-nearest neighbour list (which is rare if q is small). As is shown later, $coverage_X(k, q)$ behaves very similar to $m_X(k, q)$ for small q and k.

Finally, the fraction of data points that never show up in any other points' k-nearest neighbour lists is called *orphans*. Given $\overline{N}_k(\vec{x}) = \bigcup_{\vec{y} \in X \setminus \{\vec{x}\}} N_k(\vec{y})$ as the union of all k-nearest neighbour lists of points other than $\vec{x}$, the fraction of orphans is defined as:

$$\text{orphans}_X(k) = \frac{1}{n} \sum_{\vec{x} \in X} \left(1 - \mathbb{1}_{\overline{N}_k(\vec{x})}(\vec{x}) \right).$$

Hubness in Artificial Data Sets

Radovanovic et al. [112] show that the hubness phenomenon occurs in high-dimensional data generated uniformly distributed in a (hyper-)cube by observing that the distribution of the size of hubs is skewed to the right, see Figure 19 on the left. In order to demonstrate that such experiments strongly depend on the data distribution characteristics, four types of artificially generated data sets are studied: data of a normal distribution, data uniformly distributed in a (hyper-)cube, on a (hyper-)sphere and in a (hyper-)ball.

Figure 20 shows how each hubness measure performs with increasing dimensionality for each data set. The largest hubs occur in

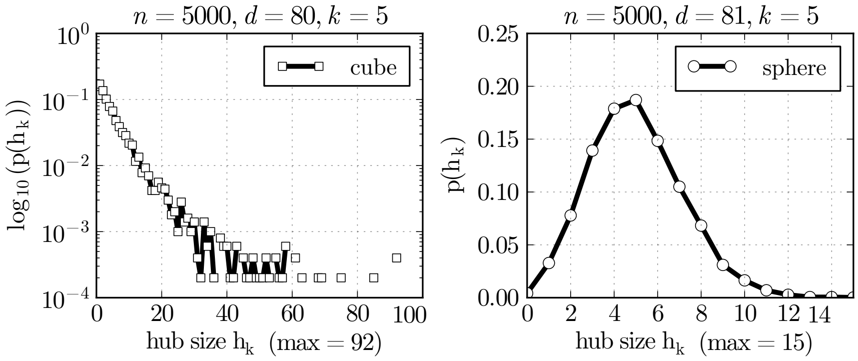


Figure 19: Distribution of the size of hubs for data uniformly distributed in a 80-dimensional (hyper-)cube and 81-dimensional (hyper-)sphere.

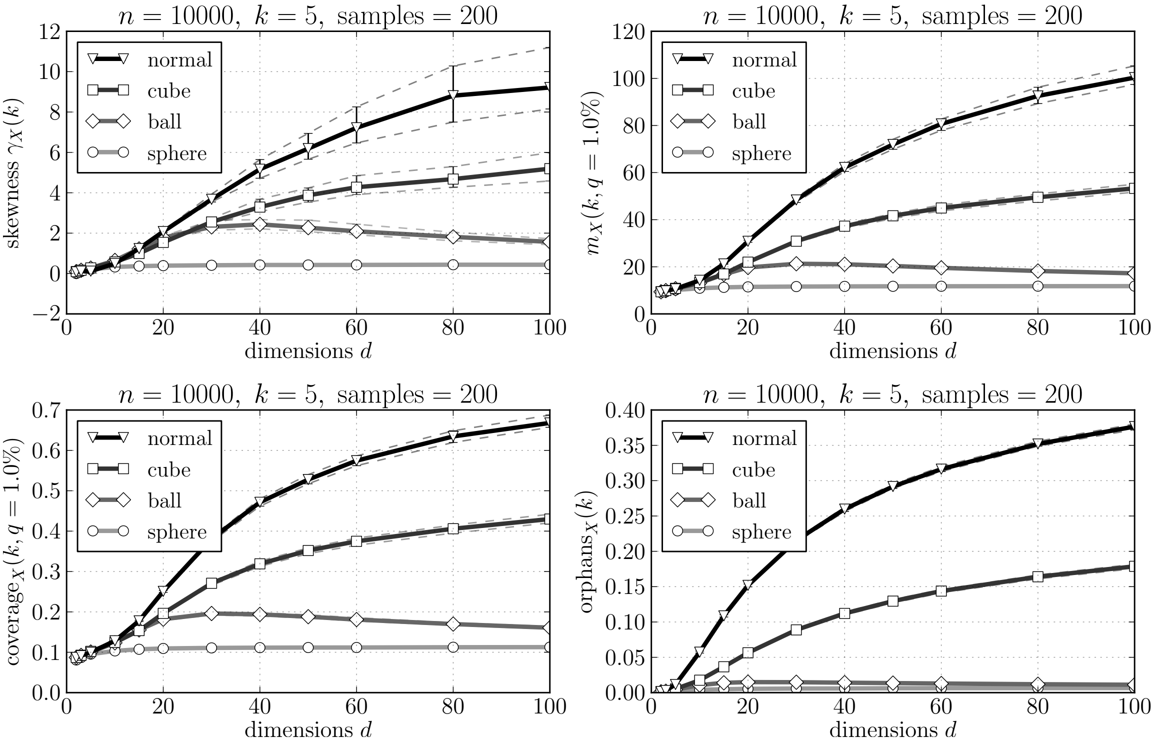


Figure 20: Median of hubness measures γ_X (top left), m_X (top right), coverage$_X$ (bottom left) and orphans$_X$ (bottom right) applied to data distributed uniformly in a (hyper-)cube, a (hyper-)ball and on a (hyper-)sphere, as well as normally distributed data for increasing dimensionality, and their first and third quartiles (dashed lines)

data generated from a normal distribution. Most importantly, hubness is not present in data distributed uniformly on a (hyper-)sphere. This is again confirmed by comparing the the distribution of the size of hubs between a 80-dimensional (hyper-)cube and 81-dimensional (hyper-)sphere as illustrated in Figure 19. Also, hubness occurs at lower dimensions for data distributed in a (hyper-)ball, and decreases at higher dimensions. Based on this observation, the following hypothesis is proposed:

Hubness Hypothesis

The presence of large hubs in high-dimensional data, i.e., the Hubness Phenomenon, is the result of density gradients in the probability density function of the random process that created the data set.

Data that is sampled from a bounded region exhibits density gradients at the boundaries of that sampling region. In case of uniform data in a (hyper-)cube, density gradients are present only at its faces and edges. Normally distributed data possesses density gradients everywhere, which might be the reason why its hubness is pronounced.

Another argument in favour of this hypothesis can be made when considering the k-nearest neighbour relations near strong density gradients. Points that lie in regions of relatively low density, but close to a strong density gradient, do not have as many neighbours to choose from and are thus *forced* to pick points from nearby high density regions. As a result, points in such high density regions are more likely to be chosen as k-nearest neighbours, i.e., they may become hubs. In case of data uniformly distributed in a (hyper-)cube, hubs are therefore expected to lie close to the (hyper-)cube's edges and faces. Similarly, hubs are expected to be occur everywhere in normally distributed data.

Vice versa, points in high density regions will most likely not contain points from nearby low density regions as their k-nearest neighbours. This explains the large amount of orphans for the data sets "normal" and "cube" in the bottom right diagram of Figure 20.

On the other hand, data uniformly distributed on a (hyper-)sphere possesses no density gradients. An n-dimensional (hyper-)sphere can be interpreted as a finite but boundary-less $(d-1)$-dimensional space. Hence, large hubs do not occur. This can be seen, e.g., in the top left diagram of Figure 20. The skewness (see line labelled "sphere") remains low even at higher dimensions. For illustration, the hubness distribution of such data is visualised in the right diagram of Figure 19.

The fact that the hubness of data distributed in a d-dimensional (hyper-)ball decreases at higher dimensions can be traced back to the same property. As the dimensionality increases, the mean of the Euclidean norm of data points distributed in a (hyper-)ball converges to 1, i.e., they get pushed out closer and closer to the surrounding (hyper-)sphere. Thus, in high-dimensional space, data distributed in a

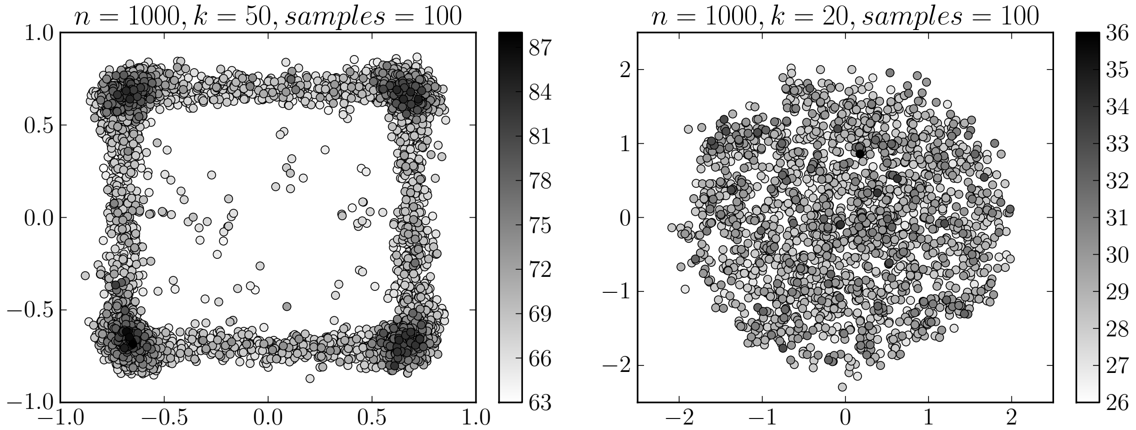


Figure 21: Collection of the 50 largest hubs in data uniformly distributed in a square (left), and the 20 largest hubs in normally distributed data (right) for 100 samples (the size of a hub is encoded as its grey value)

(hyper-)ball exhibit similar hubness in comparison to data generated on a (hyper-)sphere.

As a conclusion, the Hubness Phenomenon can not only be caused by the high dimensionality. Rather, it is a combination of both the dimensionality and density gradients in the data.

Hubness in Low Dimensions

Figure 20 suggests that hubness does not occur in low-dimensional data. However, our hypothesis states that hubs are the result of density gradients. Since density gradients exists in low-dimensional space as well, hubs should be observable to a lower extent in low-dimensional space, too.

In order to demonstrate that hubs occur at density gradients in low dimensional space, we collected the largest hubs over multiple samples of the same data distribution. Results are shown in Figure 21. In the left diagram, the 50 largest hubs are drawn for 100 samples of data uniformly distributed in a 2-dimensional (hyper-)cube, i.e., a square. As hypothesized, hubs are not equally distributed over the whole square. Most hubs appear near the boundary, and almost none close to the centre. Also, the largest hubs are located close to the edges. In contrast, when sampling from a normal distribution and selecting the 20 largest hubs from 100 samples, see Figure 21 on the right, hubs are spread evenly distributed and do not exhibit areas of increased hub size. Again, these observations support the hypothesis that hubs occur at density gradients of the data.

Upper Bound on the Size of Hubs

Considering that hubs appear in low-dimensional space, it still can be observed that they are much smaller. Radovanovic et al. [112] mentioned without a detailed discussion that the maximum size of hubs is the result of geometric constraints and related to the so-called *Kissing Number Problem*. In the following, this link is discussed in more detail.

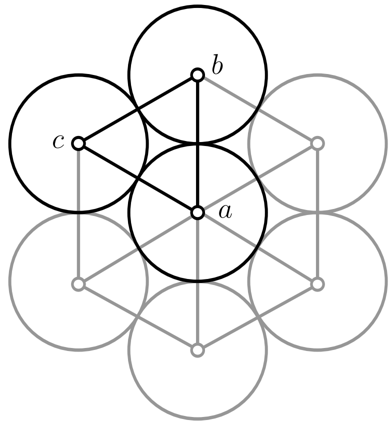


Figure 22: Maximum hubness in 2-dimensional Euclidean space traced back to the Kissing Number Problem

Geometric constrains imply that a single point can not be chosen as the nearest neighbour of arbitrary many other points. Figure 22 illustrates these constraints by geometrically trying to add new points such that a centre point is chosen as their nearest neighbour. Each new point b, c and following need to be positioned such that their respective distance is larger than the distance to a, which is supposed to be a hub of maximum size. This is modelled by surrounding (hyper-)spheres that may not overlap and are packed as dense as possible. As a consequence, each centre of a (hyper-)sphere has the same distance to each other. Also, it assumed that for distance equality point a is chosen as preferred nearest neighbour. Essentially, this problem translates into the equivalent question of how many (hyper-)spheres of the same size can be placed around a centre (hyper-)sphere without overlapping, which is known as the *Kissing Number Problem*.

The Kissing Number Problem has been previously discussed by Newton and Gregory in 1694 and is still subject of ongoing research, since it is not clear how to arrange as many (hyper-)spheres as possible in a high-dimensional space. In recent years, the problem was solved for $d = 4$ by Musin [93]. However, starting at $d \geqslant 5$ mostly upper and lower bounds are known yet. An exception are dimensions 8 and 24, which were proven by Odlyzko and Sloane [98] to have a *Kissing Number* τ_d of 240 and 196.560, respectively.

When using data that is structured according to the solutions found for the Kissing Number Problem, very large hubs can be constructed in high-dimensional space. This underlines the importance of this research, since real-world data may also exhibit complex structures, and thus may also contain much larger hubs than presented in Figure 20.

A lower bound for $k > 1$

For $k > 1$ it is difficult to grasp what kind of structures can be constructed that create hubs of maximal size. We propose without

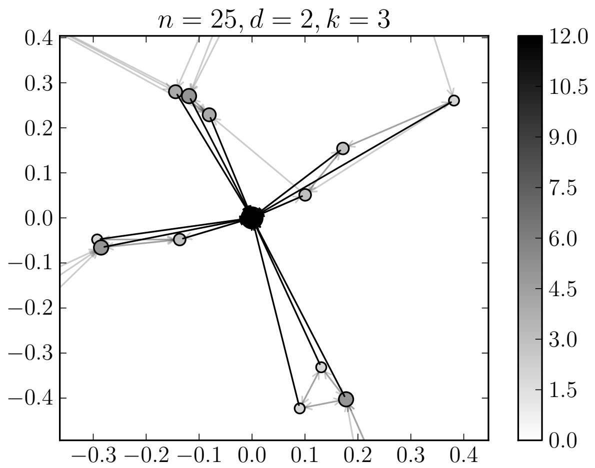


Figure 23: Best arrangement of points for maximal hubness of size 12 at $k = 3$ after 5000 generations. k-nearest neighbour relations are visualized as arrows.

proof that hubs with a size of at least $k \cdot \tau_d$ can be constructed by arranging compact groups of k points in the same way as for $k = 1$.

In order to support this idea, an experiment was designed that uses an evolutionary algorithm in order to generate an arrangement of points in a 2-dimensional Euclidean space such that a fixed centre point becomes the largest possible hub. Figure 23 illustrates a hub of size 12, whereas a maximum hubness of $k \cdot (\tau_2 - 1) = 15$ should be achievable. Based on observations of various partial solutions, points seem to be usually arranged in compact groups of k points that are located in cones around the centre point. These results fundamentally match the expectation as stated above, but do not represent a proof.

Measuring the Goodness of Hubs

Previous approaches mitigate the Hubness Phenomenon by either modifying the distance function or the k-nearest neighbour search itself, see Section 2.4.2. However, modifying the distance function or k-nearest neighbour search severely affects the interpretability of a k-nearest neighbour relationship. Instead, in the following a method is proposed that measures to what extend a hub has been created as a result of a density gradient.

It is assumed that points which lie in regions of relatively low density (e.g., near the faces or edges of the (hyper-)cube as shown in Figure 21), are *forced* to pick points from nearby regions of relatively higher density as their k-nearest neighbours. Thus, a measure that identifies to what extend a large hub has been chosen as k-nearest neighbour by points from low density regions may allow to distinguish *forced* hubs from hubs that are the result of structure in the data.

The main idea is to measure the change in the local density around points that contain a hub as their k-nearest neighbour. A hub is considered "good", i.e., not created as a result of the Hubness Phenomenon, if the regions around all points that contain this hub in their k-nearest neighbour lists exhibits a similar local density. In order to measure differences in local density around a point $\vec{x}$, the norm over the sum of all vectors from $\vec{x}$ to its k-nearest neighbours is used.

$$\Delta_{\text{dens}}(\vec{x}) = \|(\sum_{\vec{y} \in N_k(\vec{x})} \vec{y} - \vec{x})\|$$

It can be interpreted as the norm of the centre of mass of a point's k-nearest neighbours. In case of the presence of a density gradient, it is likely that more than one k-nearest neighbour is located in the nearby region of relatively higher density. Thus, the centre of mass would be pulled into the direction of high density regions, and its norm would become non-zero. Of course, this rough approximation is not suitable for low values of k. However, it has the advantage that only the k-nearest neighbours are needed for its calculation.

The goodness of a hub is then measured as the mean of the change in local density Δ_{dens} of all points that contain this hub in their k-nearest neighbour lists.

$$g(\vec{x}) = \frac{1}{h_k(\vec{x})} \sum_{\vec{y} \in X} \Delta_{\text{dens}}(\vec{y}) \cdot \mathbb{1}_{N_k(\vec{y})}(\vec{x})$$

Experiment with Artificial Hubs

Note that "good" hubs have lower values of $g(\vec{x})$.

In order to test this measure, we introduced artificial hubs in data uniformly distributed in a (hyper-)cube. These hubs were generated by arranging $2 \cdot d$ points around a seed point $\vec{p}$ such that $\vec{p}$ is contained in the k-nearest neighbour lists of all $2 \cdot d$ points. Thus, its hubness $h_k(\vec{p})$ equals $2 \cdot d$, unless some uniformly distributed points interfere with this structure. The surrounding $2 \cdot d$ points were arranged in a naive way similar to the 2-dimensional structure presented for the Kissing Number Problem. All points were grouped into d pairs such that each pair of points differs to $\vec{p}$ only in one dimension by adding and subtracting some small $\epsilon > 0$, respectively. Of course, such an arrangement implies that $\Delta_{\text{dens}}(\vec{p})$ equals zero, since the sum of all vectors from $\vec{p}$ to all $2 \cdot d$ points cancels out (unless some point from the uniformly distributed data interferes).

Figure 24 visualizes $g(\vec{x})$ for points generated uniformly in a (hyper-)cube (black points), 10 artificially introduced hubs (white square) and their supporting $2 \cdot d$ points (grey points). As expected, $g(\vec{x})$ is close to zero for all 10 artificial hubs of size 100. However, it is much larger for hubs introduced by the data that is uniformly distributed.

As a conclusion, this measure allows to distinguish between hubs that are the result of density gradients and potentially interesting hubs. However, more research is required to confirm these results for real world data sets.

Implications for Map-Based Exploration

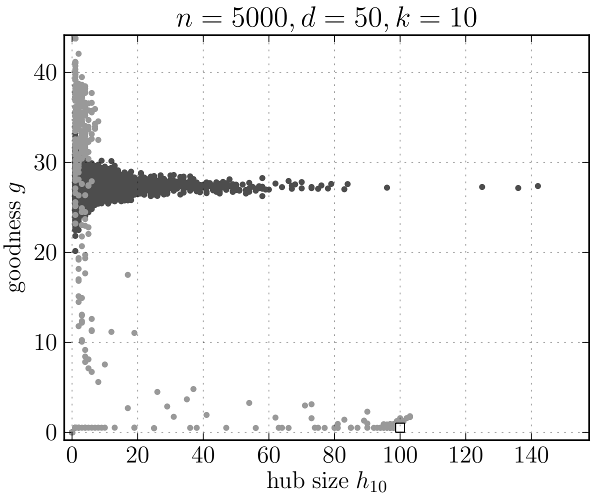


Figure 24: Uniform data distributed in a (hyper-)cube (black) with some artificially introduced hubs (white square) and their supporting points (grey).

In summary, the Hubness Phenomenon provides multiple challenges for k-nearest neighbour navigation in high dimensional information spaces. For one, the data might contain orphans, which can not be found by the means of jumping from one local map to the another. However, in order to avoid orphans they could be artificially introduced in k-nearest neighbour lists of their own nearest neighbours. Hubs on the other hand might even have a positive impact on user navigation patterns, since users might recognize them during navigation, such that they act as a similar point of reference or landmark. Furthermore, hubs could be evaluated via the proposed measure of goodness of a hub in order to artificially hide or demote bad hubs, and highlight or promote good hubs. Unfortunately, no further research considering hubs and orphans has been included in the following user studies and prototypes. Apart from hubness, another limitation of k-nearest neighbour search can be self-similar clusters with a size larger than k. In this case, users would be "trapped" inside a group of objects whose nearest neighbours are objects from that same group. Then, user can not escape by the means of jumping from one local map to another. This problem was identified as part of a user study discussed in Section 11.3.

10.4 A USER INTERFACE PROTOTYPE FOR IMAGE EXPLORATION

Exploration is both a highly dynamic and user-specific process. Users have different preferences, prior knowledge and abilities. Thus, evaluating any novel exploration method is a challenging endeavour. Traditionally, human computer interfaces are developed in a user centred design process, which was also adopted in this thesis. Over the course of multiple iterations an interface prototype was developed, evaluated and improved. Different stages of this prototype will be described in this and following sections. However, very early drafts of the design process, e.g., paper prototypes, are not included in this thesis.

Aim

The purpose of this first prototype is to create a proof-of-concept implementation of the visual berrypicking method as described in the previous Chapter 10 and to to collect initial feedback from first users and fellow researchers. In later Sections 11.2, 11.3 and 11.4, various aspects of this prototype are further improved, discussed, analysed and evaluated. Both the initial concept of the visual berrypicking method and the following prototype were published in [pub:5].

Dataset

The prototype is based on the *Caltech101* image data set [33]. It consists of 9144 images from 101 categories, e.g., images of various kinds of animals and objects. Due to the data set's focus on various object classes, it was assumed that the data contains both local and global structure. For each category, the data set contains a number of very similar images, e.g., the same animal is captured in various poses, which should form clusters that represent local structure with respect to the similarity metric. On the other hand, due to the large number of different categories, the data is also assumed to contain global structure in form of different categories that are related to various degrees. For example, the categories "butterfly" and "dragonfly" are expected to be visually more similar than categories "butterfly" and "airplanes", since images of the former categories share more features. Thus, images from the categories "butterfly" and "dragonfly" may have overlapping k-nearest neighbours, whereas images from the categories "butterfly" and "airplanes" are less likely to share common neighbours. However, the category information itself was not directly used by the prototype as a feature.

Image Similarity

The similarity metric consists of four commonly used similarity functions from the area of content-based image retrieval (CBIR) that are weighted based on user preference. The first three similarity functions correspond to the recommended distance function for each of the following image feature from the MPEG-7 standard: colour layout, scalable colour and edge histogram [book:11]. The colour layout descriptor uses the weighted Euclidean norm of coefficients determined from a discrete cosine transformation of the average pixel colour after partitioning the image in 8x8 blocks. Scalable colour and edge histogram both use the Manhatten norm. Finally, the last feature was selected to be the YCbCr colour histogram, which was again evaluated by the Manhatten norm. In order to improve performance of the prototype, each image feature was extracted in advance and stored in a database.

Map Generation

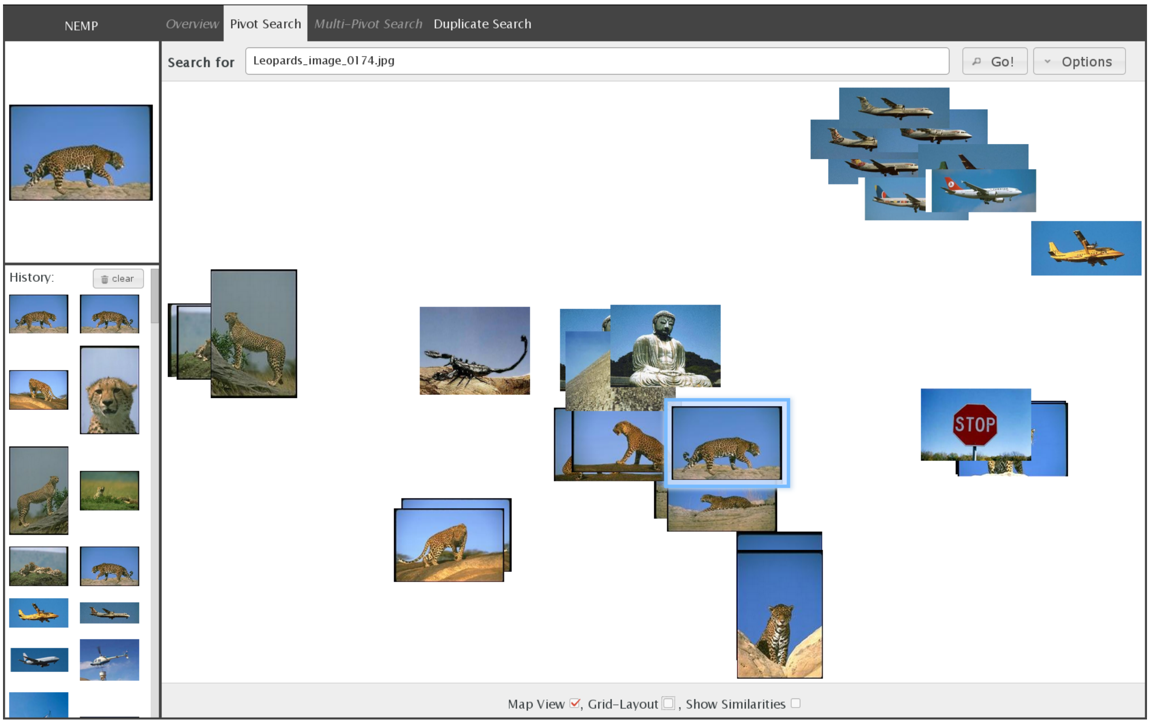


Figure 25: First prototype implementing the visual berrypicking method: thumbnails of images are arranged according to their pairwise similarity (centre); the current seed image is highlighted with a blue border and visualized by a larger thumbnail (top left); a list of past seed images allows to navigate to previous maps (bottom left).

Following the approach presented in Section 10.2, the prototype uses a k-nearest neighbour search given a seed image to determine the images that are presented to the user. Due to the relatively small number of images, the nearest neighbour search was implemented as a simple linear search. Given the list of k-nearest neighbours, a pairwise similarity matrix is calculated. Then, the final map projection is generated by applying both MDS and Procrustes analysis.

User Interface

The resulting map is presented to the user by a web-based user interface. A screenshot of the prototype can be found in Figure 25. Each of the k-nearest neighbour images is rendered using a small thumbnail. The currently selected seed or query image is highlighted by a blue border. When a user clicks on any thumbnail, the corresponding image is selected as a new seed object, and a new set of k-nearest neighbours is retrieved and visualized. By iteratively jumping from one image to the next, the user is enabled to navigate and explore the collection as a whole. Ideally, each transition from one local map to the next is perceived as panning a structurally stable global map.

Figure 26 illustrates how seemingly unrelated categories of images can be explored due to visual similarities between images of different categories. In this example, the user starts with an image showing a sailing boat on the ocean. Since most image features are colour-based, images of airplanes flying in front of a blue sky - matching the

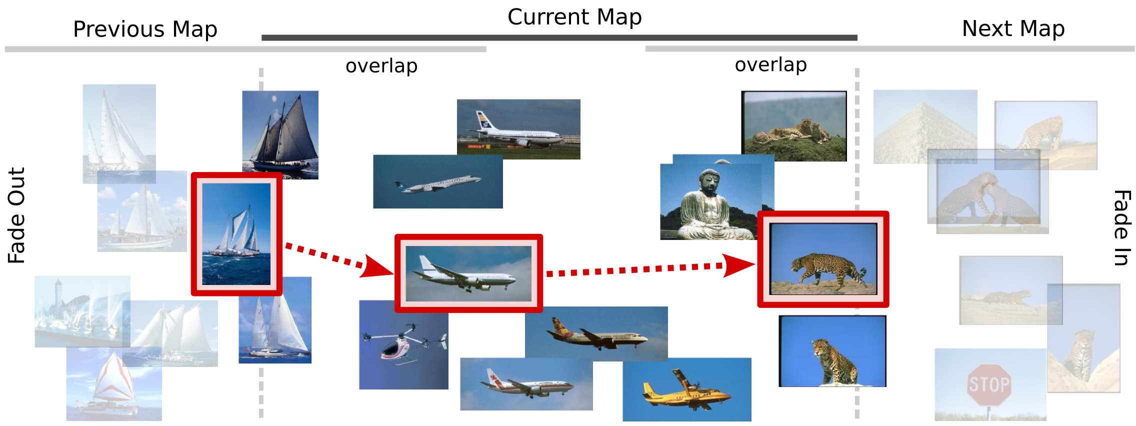


Figure 26: Navigation path when jumping from one seed image to the next: consecutive maps partially overlap, which allows to align local maps and create the impression of panning a larger, structurally stable global map.

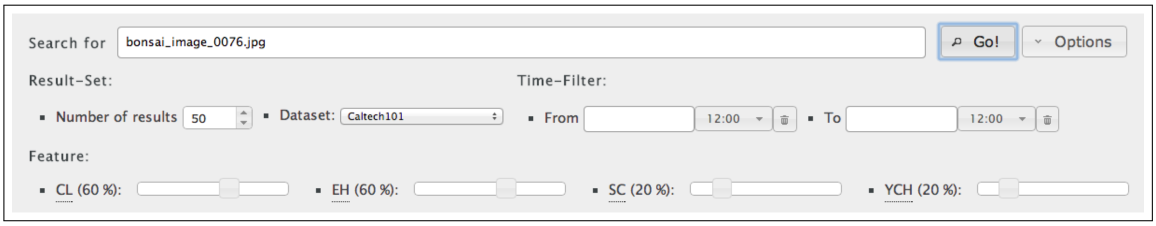


Figure 27: Query parameters can be modified by the means of a simple form: the number of k-nearest neighbours, a time range (not applicable for the Caltech101 data set) and weights for each feature.

colour of the blue sea - are determined to be similar and, thus, appear amongst the k-nearest neighbours. Next, the user selects an image depicting such an airplane. Again, due to the mostly blue background, a few images of leopards are included in the k-nearest neighbours as well, which allows to navigate to an animal category, and so on. This way, the whole collection of images can be explored step by step. Due to the alignment of each map, common neighbours move from one side of the screen to the other, creating the impression of panning a large map. Consistent panning animation are believed to support users in identifying navigation directions and a sense of orientation. In a later Section 11.1 this hypothesis is further investigated.

Feature weights can be adapted live during a search session by the means of sliders, see Figure 27. Each distance function is then scaled by a factor between zero and one. As soon as a user confirms new weight settings, both the k-nearest neighbour search and the map arrangement are updated accordingly.

In order to avoid overlappings between thumbnails of very similar images, a grid regularization method was implemented. Each thumbnail is slightly moved to a nearby free grid cells, see Figure 28. Due to the grid arrangement, overlappings are prevented at the expense

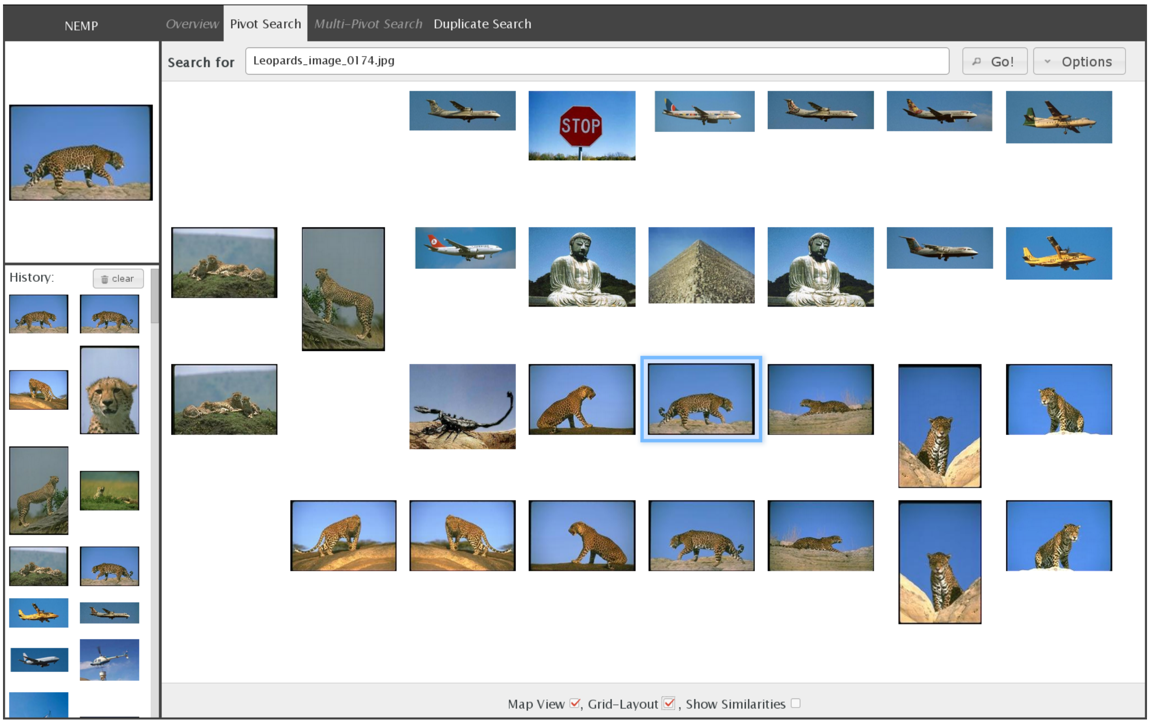


Figure 28: A grid regularization of thumbnails avoids overlappings at the expense of projection accuracy. Clusters are generally preserved, but require a number of empty grid cells to be visualized as separate groups of thumbnails.

of projection accuracy. However, when using grid regularization, the number of k-nearest neighbours being visualized should be chosen to be significantly smaller than the number of available grid cells, such that a good portion of cells will remain empty. Only then clusters are preserved and can be easily identified by users. A detailed discussion of grid regularization can be found in Section 9.3.

For comparison, the prototype also allows to visualize k-nearest neighbours as a traditional ordered list of thumbnails, see Figure 29. The ordering is based on the similarity of an image to the current seed image. Below each thumbnail a normalized similarity score is presented, which allows to gain additional insights in the similarity function.

Furthermore, the user interface also provides a history panel containing an ordered list of recent seed images, see Figure 25, 28 or 29 at the bottom left. This history is assumed to support users in better understanding the navigation path that was used to jump from one map to the next. Ideally, the list allows to observe a gradual transition of image features, e.g., the change in foreground colour when navigating from airplanes to leopards. Additionally, users are enabled to easily revert back to a previous map by clicking on any of the thumbnails inside the history panel.

Summary

The proposed prototype provides a first implementation of the visual berrypicking method, which allows to interactively explore a

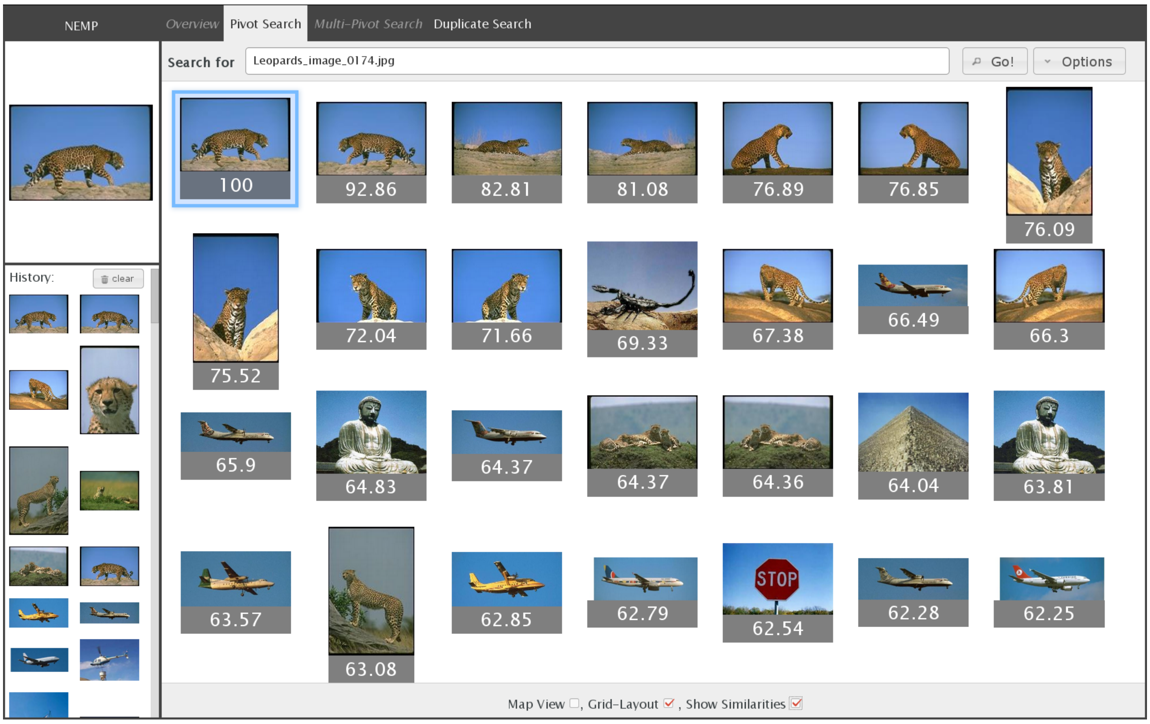


Figure 29: Traditional list-based search result presentation: images are ordered by their similarity to the query image; a normalized similarity score is presented below each thumbnail.

small data set of images. Images can be discovered by jumping from one k-nearest neighbour set to the next, whereas each set is visualized via MDS in order to illustrate pairwise similarities. Initial feedback was gathered from short interviews of fellow researchers at a relevant conference demo session. A few users struggled with the similarity-based arrangement and reported to have expected some kind of fixed x- and y-axis definition. Other users were slightly confused by the k-nearest neighbour selection bias, which favours images of similar colour distribution, and therefore often contains images of multiple categories. Instead, those users initial expectation was that only images of a single category should be shown. However, such a selection bias was introduced intentionally to allow users to jump from one category to another. Further user feedback will be discussed as part of detailed user studies in the following sections. Each study will investigate different aspects of this prototype in different scenarios.

11 POTENTIAL OF VISUAL BERRYPICKING: USER STUDIES

In the following, three consecutive user studies are discussed. Each user study builds upon the experience gained during the former. The first study is conducted as a controlled lab experiment on artificial data with a limited number of study participants. The second study evaluates the method of visual berrypicking on a real world data set using a web-based online prototype. Unfortunately, the study was designed too openly, which significantly restricts any conclusions that can be drawn from results. The third study deals with this problem and provides a guided online user study comparing two user interfaces in context of a specific exploration task.

11.1 PRE-STUDY ON ARTIFICIALLY GENERATED IMAGES

Aim

The aim of this work is to evaluate the proof-of-concept implementation for visual berrypicking described in Chapter 10 on a fundamental level. Instead of using a real world data set, artificial images are generated, which are expected to allow to better track whether users are able to grasp the concept of a similarity-based map visualization, as well as whether users are able to navigate the data set using the proposed interaction pattern. The goal is to objectively measure a difference in retrieval performance when comparing both the proposed map exploration approach with a traditional list-based result presentation. Besides, the user study is designed to gather user feedback regarding the overall usability of the prototype implementation. Unfortunately, the following user study was not successfully published. Reviewers expected additional results using a real world data set, which at the time did not exist yet, but were later published as described in the following two Sections 11.2 and 11.3.

Related Work

Evaluating exploration and exploratory search systems is a challenging task, since exploration success is difficult to measure. It involves users learning about an information space, which is a highly dynamic and user-specific process. White et al. [158] suggest to evaluate exploration systems in a comparative user study that indirectly measures the time to learn by posing a task that benefits from the user's understanding of an information space, assuming that users which have a better understanding will be able to solve a given task more quickly than others. In addition, [158] propose that exploration systems can be evaluated similar to other highly interactive systems by measuring usability and user satisfaction. In the following user study, both strategies are applied. Due to the novel design of the exploration approach, no user studies that are directly comparable could be identified in the literature. However, other exploration systems are evaluated in a similar way. An discussion of evaluation approaches can be found in Section 2.3.

Study Prototype

The study prototype is implemented as a simplified version of the prototype described in Section 10.4. A web-based user interface renders thumbnails of the k-nearest neighbours of a given seed image. Both a list and map visualization are available, see Figures 30 and 31. The map projection is calculated by using MDS and Procrustes analysis. The list-based presentation shows images ranked by their similarity to the seed image in a grid with decreasing similarity from top left to bottom right. Clicking on a thumbnail will select a new seed image and result in an update of the current map or list. Map transitions are animated by moving each image that is contained in both k-nearest neighbours sets from its previous position to the next. Subsequent interactions allow to explore the whole collection of images.

In addition to the list and map visualization, a side panel provides information about the current retrieval task and allows to controls various options of the user study, see the left side of Figure 30 and 31. In contrast to the prototype presented in Section 10.4, feature weights can not be adapted by users, and there is no history panel showing a list of the most recent seed images.

Dataset

The data set consists of a set of $n = 2000$ images of regular polygons of varying size, number of edges, background colour and foreground colour, see Figures 30 and 31. The range of colour tones, polygon size, as well as the number of edges were specifically tuned to generate a set of images that reaches a certain level of complexity with regards to the similarity space. Also, image parameters were limited in such a way that feature variations can be easily distinguished by most users, e.g., the number of edges of polygons was restricted to nine edges, since polygons with higher number of edges would approximate a disc and be very hard to differentiate. The hue of a colour was not limited. Therefore, participants were interviewed for potential colour vision deficiencies.

In comparison to an otherwise highly unstructured real world datasets, using artificial images allows to carefully design the complexity of the similarity space. Each feature of the similarity space is clearly represented by each image, which allows to observe the exact differences when comparing two images. Furthermore, the data set is sufficiently complex, i.e., it consists of more than two features such

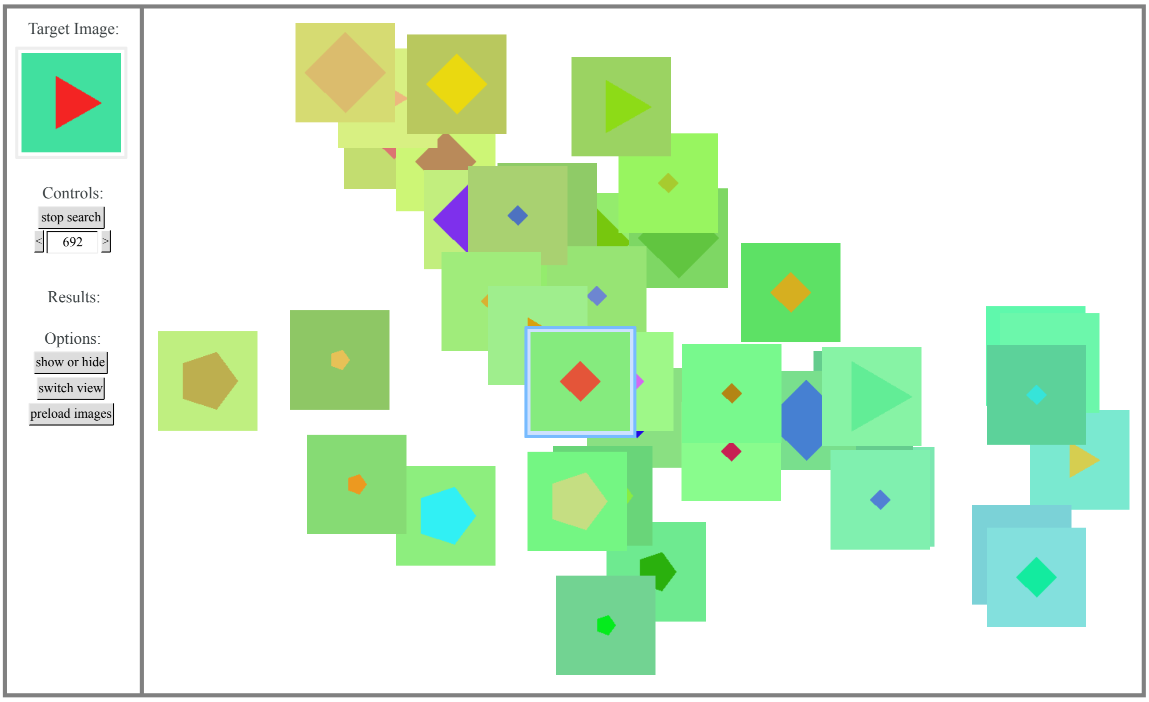


Figure 30: Screenshot of 40 polygon images arranged in the proposed map interface. The highlighted image (blue frame) is the currently selected seed or query object. The left panel visualizes the target image of the current retrieval task as well as additional options that control the user study.

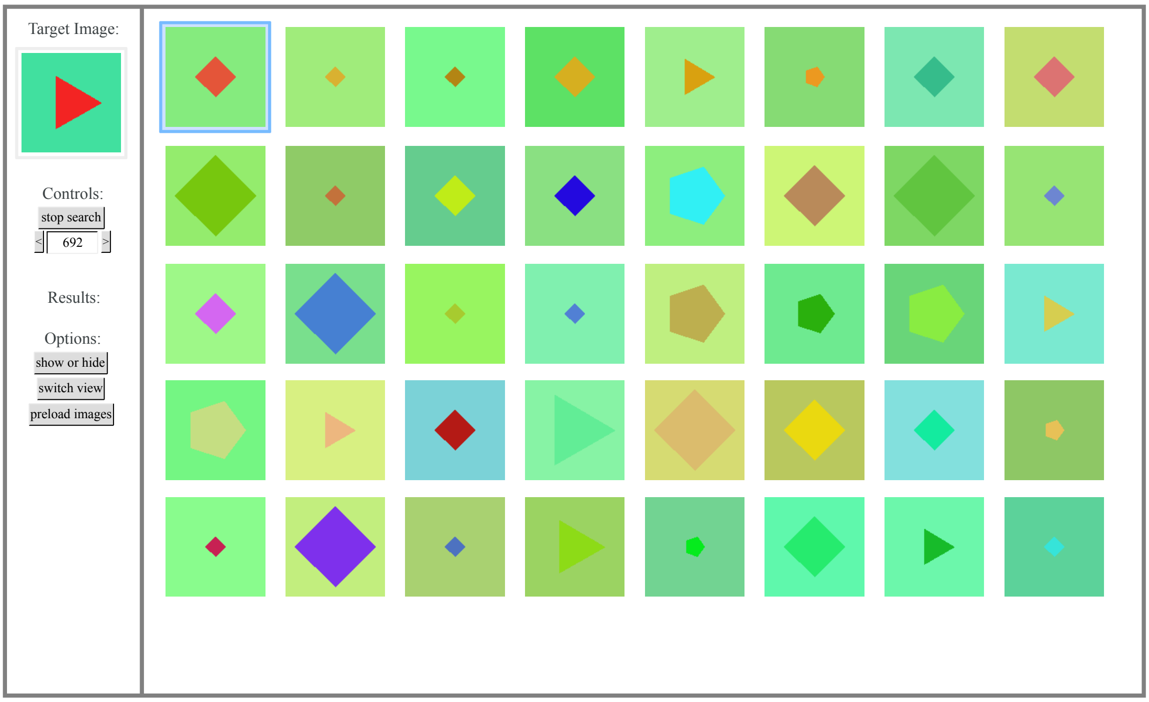


Figure 31: Control interface showing polygons in a traditional list representation from top left to bottom right. The list order is determined by the similarity of a polygon image to the currently selected one (the first in the list).

that a projection to 2-dimensional space will result in projection errors, imperfect clusters and non-obvious navigation patterns.

Similarity Space

Each parameters of the image generation process is directly used as a feature of the similarity space. In total, each image is described by a 8-dimensional vector: the size or radius r of the polygon, the number of edges e, the foreground f and background b colour both in HSV colour space. Then, the similarity space is described by the following distance function.

$$\begin{aligned} d(i, i') =& w_b * \|b_i - b_{i'}\|_2 + w_f * \|f_i - f_{i'}\|_2 + \\ & w_r * \|r_i - r_{i'}\|_1 + w_e * \|e_i - e_{i'}\|_1 \end{aligned} \tag{2}$$

It corresponds to a simple weighted sum of the squared differences of each feature. Weights were determined empirically in pre-tests to $w_b = \frac{5}{9}$, $w_f = \frac{1}{9}$, $w_r = \frac{1}{9}$ and $w_e = \frac{2}{9}$. The hue was modelled in an interval between 0 and 1, such that the colour red would be split to either sides of the interval. Unfortunately, the distance function did not respect that red tones with a hue near 0 and 1 represent similar colours. Since most study participants were students of computer science, this circumstance was quickly observed and understood by the majority of study participants.

Map Generation

Maps are generated in the same way as discussed in previous sections. Given a seed polygon image, the $k_p = 120$ nearest neighbours are calculated based on the distance function described above. Then, a 2-dimensional projection is performed using MDS. Subsequent maps are aligned via Procrustes analysis based on common neighbours. Finally, $k = 40$ nearest neighbours are visualized to the user. Both k and k_p were determined empirically based on the monitor size available during the user study, i.e., $k = 40$ polygon images could be visualized on the screen without significant overlaps between images (no grid regularization was performed), and the number of nearest neighbours $k_p = 3k$ was deduced from the alignment error evaluation described in previous Section 10.2 as illustrated in Figure 18.

Study Design

The study is modelled as a total of four retrieval tasks. Participants are presented a target image, which is then supposed to be retrieved, i.e., found and clicked on by the means of interacting with the user interface. Exploration effectiveness is measured indirectly by comparing the average retrieval performance of participants between the list- and map-based interfaces. Users that are able to better learn from either the list- or map-based visualization of the information space are expected to more efficiently solve subsequent retrieval task. All participants are given the same tasks in order to ensure comparability. Both the target image and the initial seed image were specifically selected in advance, see Table 4. As discussed in Section 2.4.2 and 10.3, due to statistical properties of high dimensional spaces, the number of times an image occurs amongst the nearest neighbours of other images is usually not equally distributed. As a consequence, some target

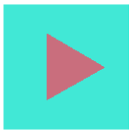
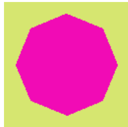
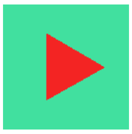
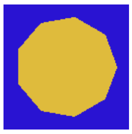
Task	1	2	3	4
Seed Image				
Target Image				
User Interface	A	A	B	B

Table 4: Seed and target images for each of the four retrieval tasks. Initial tasks 1 and 3 are designed to target a triangle, which was generally perceived to be easier due the fewer number of edges that otherwise have to be counted when comparing higher number polygons.

images might occur never (orphans) or only in very few other nearest neighbour sets, which makes it impossible or very hard to navigate to using the proposed exploration method. Therefore, seed images were selected semi-randomly for each target with the restriction that the corresponding target image is always reachable by six navigation interactions, and that all features were sufficiently different between seed and target images. Target images were selected manually, which allowed to influence the difficulty of each retrieval task. Images showing polygons with a small number of edges (triangles, squares) were found to be easier to distinguish by users, and thus, easier to retrieve, since polygons of more than six edges are less familiar to most users and often require to specifically count the number of edges in order to effectively perform a retrieval task.

The first two out of four retrieval tasks were solved by one visualization method, e.g., the list-based visualization. The last two tasks by the other method, e.g., the map-based visualization. In order to avoid response bias due to the order in which interface variants are presented, participant were split into two groups and assigned alternately. Depending on the group, the list or map-based visualization was used first. After completing two retrieval tasks of a certain visualization method, participants were asked a number of questions regarding the comprehensibility of the arrangement of images and the usability of the user interface. Before each variant, participants were given a short amount of time to get accustomed to the respective interfaces. For each retrieval task, the time and number of navigation interactions was recorded.

Study Results

In total, 21 participants, 7 female and 14 male, of age between 23 and 42 with an average of 30 years, completed the user study. Table 5 lists four metrics recorded for each participant and user interface variant. The first two metrics describe the average time in seconds and the

Measured Value	List View	Map View
Time (in secs)	107	73.0
No. of Clicks	10.5	7.7
Comprehensibility ($\in [1, 5]$)	2.75	3.40
Usability Score (SUS $\in [0, 100]$)	66.9	73.8

Table 5: Four evaluation metrics averaged over 21 participants for both the map and list-based user interface.

average number of navigation interactions required to retrieve the target image. When comparing averages for both the first and second retrieval tasks separately, almost all participants were able to improve the retrieval speed, which suggests that most participants required additional time to get accustomed to the task and user interface. As a consequence, the results presented in Table 5 only consider the respective second retrieval tasks, which is assumed to better represent the actual retrieval performance of each participant. Additionally, in order to reduce the influence of outliers, both the smallest and highest value for each metric were omitted. The third and fourth metric of Table 5 describe the average rating of the question "How understandable was the (re-)arrangement of the images during navigation?" on five-point Likert scale, and the average System Usability Score (SUS) calculated for six questions targeting the overall usability of each interface, respectively.

All metrics suggest that the map-based user interface allows for a higher retrieval performance on average, and is easier to comprehend and use given the artificial retrieval task at hand. Unfortunately, all results are also not highly significant and, as consequence, further research is needed.

Additional user feedback partially confirms these results. Many users (8 out of 21) specifically report that they perceive the map-based interface to be visually more intuitive or more usable. In contrast, some users (6 out of 21) report to experience the list-based arrangement as completely random. However, this perception might be the result of the design decision to order images horizontally row by row instead of vertically in a single column, which is more common in search user interface design but would have required scrolling, since not all images could be visualized in one single column. Some participants (6 out of 21) suggested to reduce the overlap of the images in the map view. Unfortunately, at the time of the user study, the map-based prototype did not support grid regularization as discussed earlier in Section 9.3. Further feedback from participants included the idea to add a history of previously selected seed objects and the suggestion to show more images in the list-based visualization since a lot of screen space is not used.

User study supervisors observed that the learning effect between the first and second tasks often was the result of participants discovering navigation directions corresponding to individual features. For example, the projection illustrated in Figure 30 seems to represent the background hue more clearly and consistently than any other feature, which - once discovered by a user - can be exploited to quickly navigate to the desired target hue. As a consequence, explicit or implicit feature weighting might further improve both the usability and retrieval performance if appropriately controlled during a retrieval scenario.

Summary

In summary, this work proposes a user study concept to evaluate the method of visual berrypicking. First results using artificially generated images do not clearly identify the map-based exploration concept as more efficient in comparison to a traditional list-based visualization. However, many users were quickly able to comprehend the map-based arrangement of items and utilize recognized navigation directions to successfully solve several retrieval tasks. Even though no statistically significant results could be obtained, the goal of this user study can still be considered to be achieved. Both retrieval performance results and initial user feedback confirms the potential of visual berrypicking as an effective method for exploratory search. However, a detailed evaluation with more study participants is required. Also, the design gap of using classic retrieval tasks to evaluate exploratory effectiveness needs to be addressed. In the next Section 11.2, first another qualitative user study is performed. Instead of artificial data, a real world document collection is used in order to facilitate true exploratory search scenarios. Also, the prototype is extended with many additional features, e.g., a full-text search, such that users can follow multiple behavioural patterns. Later in Section 11.3, the study design is refined and implemented as a online web study, which allows to evaluate the method of visual berrypicking with many more study participants.

11.2 QUALITATIVE EVALUATION FOR DOCUMENT EXPLORATION

In this section, a second qualitative evaluation of the proposed visual berrypicking method is conducted. Results of an initial user study have suggested that users are able to comprehend the proposed similarity-based map visualization when using simple artificial images. Unfortunately, many real world information spaces are composed of highly complex information objects, e.g., documents that consist of multiple facets like title, abstract, authors, and full text.

Aim

Therefore, the aim of this work is to qualitatively evaluate whether visual berrypicking can be an effective method for search and exploration using a real world data set. In contrast to the initial prototypes presented in Section 10.4 and 11.1, which are based on images that can be quickly grasped and compared by users, the following user study allows to search and explore a document data set, which requires a much more detailed inspection of each object in order to judge its relevancy. As a consequence, any benefits that may be the result of being able to transfer knowledge about the relevancy of documents due to their clustered arrangement in a map-based visualization are expected to be more pronounced and easier to validate in a user study. In addition, using full text documents provides the opportunity to combine both traditional keyword search and map-based visual berrypicking within a single user interface prototype. Such a combination allows to directly investigate whether users actually make use of the available map-based navigation or stick to classic incremental query modification as suggested in traditional keyword-based berrypicking.

Both prototypes were published at the ISMIR conference, once in 2015, see [pub:8], and once in 2019, see [pub:12]. In 2019, the data set was updated and study results were extended. In the following, both study results are discussed in combination since they are based on very similar versions of the prototype as well as highly overlapping datasets.

Related Work

Similar to before, due to the novel design of the exploration approach, no user studies could identified in the literature that can be directly compared to the results below. However, there exist other search and exploration approaches for document collections. For example, the ISMIR Cloud Browser [45] performs topic analysis and enables users to search and filter ISMIR publications based on these topics. In contrast to the propose approach, documents are not embedded in a 2-dimensional map, but visualized as a classic list of search results. Recent work includes [105], who generate a global map of hierarchical clusters of keywords, which than can be explored by zooming into individual clusters until the documents for specific keywords are shown. Repke et al. [118] propose a method that generates a combined map and graph layout of documents by fusing optimization objectives of document similarity and document to entity relations, e.g. authorship and co-authorship. In contrast, the proposed prototype only generates a similarity-based map for a small subspace of k-nearest neighbourdocuments for a given seed document or search query.

Dataset

In order to facilitate an independent exploration and discovery of relevant result objects by study participants, which are mostly comprised of computer science staff and students, a data set of scientific papers published at the ISMIR conference for Music Information Retrieval was compiled. In its first version, it contained 1362 papers released up until 2015. Later in 2019, the data set was updated to a total of 1810 papers. For each paper, its title, the list authors, abstract and full text was extracted. Additionally, a preview image of each paper's first page was generated for illustrative purposes. Finally, each paper's citation count was extracted from Google Scholar[1].

The similarity metric used for generating a two-dimensional projection is calculated as the cosine similarity on tf-idf vectors extracted from the 1000 most frequent terms of each paper after stemming. In contrast to earlier prototypes, additional features were not considered and mixed together, e.g., by including the Jaccard similarity on the set of authors. There are two factors leading to this decision. On the one hand, the aim was to focus on content similarity between papers in order to support users in discovering related papers independent of their citation count or authors. On the other hand, due to the discrete nature of the set of authors, a similarity metric would be highly sparse since it is unlikely that two papers share common authors. Mixing these similarity metrics by linear weighting might result in inconsistent projections, which requires further research. In a later Section 11.3, various features of movies actually are combined in this way, e.g., the Jaccard similarity of movie genres or the list of actors, and the cosine similarity of tf-idf vectors on a movie's description. However, there are only a few movie genres in total, and for each movie the full cast of actors is known, such that the likelihood of common actors and genres is much higher. Additional insights are discussed in Section 11.3.

Interface Prototype

The prototype is based on previous implementations presented in earlier Sections 10.4 and 11.1. An illustration can be found in Figure 32. It uses a k-nearest neighbour search ($k = 30$) using the previously described similarity metric, MDS for projecting the set of nearest neighbours to a 2-dimensional map, and Procrustes analysis for aligning consecutive maps when seed objects change due to user interactions. The first seed object is selected randomly.

In contrast to previous prototypes, a traditional search query input is shown at the top of the user interface. When entering and confirming query terms, a keyword search on the full text of each paper is performed. In order to mimic a classic linear search as much as possible, the top $k = 30$ most relevant results are retrieved and visualized. Alternatively, the very first most relevant search result could be used as a new seed object for a new k-nearest neighbour search. However, there is no guarantee that any nearest neighbour of a search result matches a query term. As a consequence, users might get confused that papers which do not match a query are shown when performing a classic keyword-based search. Instead, the k most relevant results are visualized in the same way as a k-nearest neighbour set, i.e., a pairwise similarity matrix is generated between the k most relevant results and used for projection via MDS.

1 https://scholar.google.com/

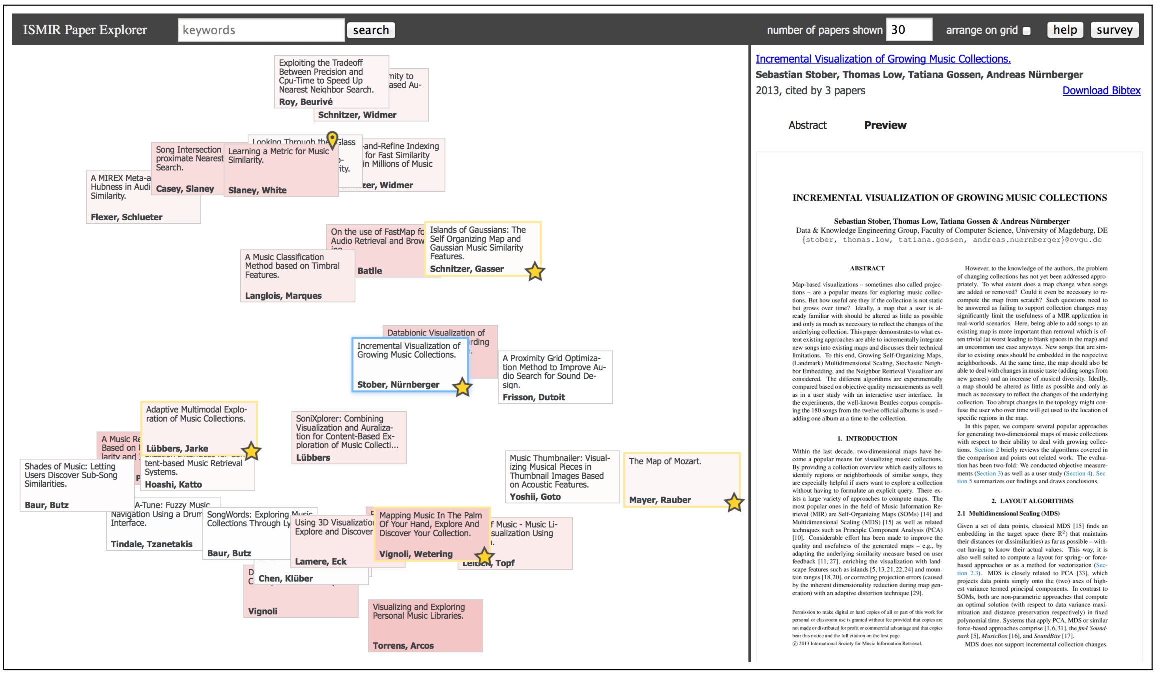


Figure 32: Screenshot of the full user interface: at the top left, search queries can be entered; at the top right, map visualization parameters can be changed; at the right, an information panel presents detailed information about the paper last selected by the user by hovering over it with the mouse; at the left, the map visualization clusters thumbnails representing each paper utilizing their pairwise distances.

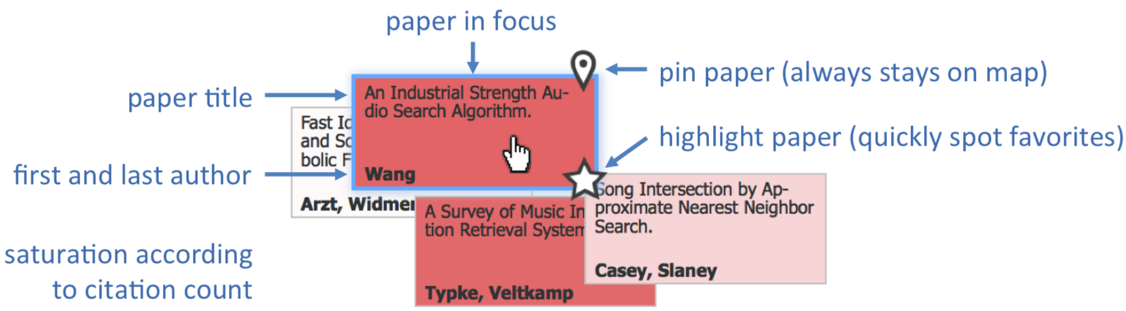


Figure 33: Four box shaped thumbnails representing a paper of the ISMIR conference by its title, the last name of its first and last author, as well as a specifically coloured background depending on the number of citations. The current seed object is highlighted by a blue border.

Another difference to previous prototypes is that scientific papers can not be easily represented in full detail by simple thumbnails. However, designing a suitable thumbnail visualizations for complex objects is another research question on its own, which is not specifically addressed in this thesis, see Challenge 3 (summarizing objects). For this user study, a simple box containing both the title as well the last names of the first and last author is used as a thumbnail for each paper. The background colour of the box correlated to the number of citations of a paper. Papers with a higher citation count are highlighted with a more saturated background. An illustration can be found in Figure 33. In addition, two interactive icons allow to toggle the following two options for each paper individually. A star symbol allows to mark a paper as a bookmark. After clicking on the star, the star icon will remain visible for that particular paper such that it can be found more easily amongst the current set of papers. However, marking a paper as bookmark has no influence on the map projection itself. In contrast, a paper can also be pinned such that it remains visible at all times by clicking on a pin symbol. Pinning a paper will permanently keep a paper visible, even if it does not appear amongst the k-nearest neighbours of the current seed object or match any of the provided query keywords.

In 2019, the prototype was adapted in hopes of better supporting users in distinguishing old and new research papers. In addition to representing the citation count via the saturation level of a box background colour, the hue of the colour was used to reflect the publication year. Unfortunately, as mentioned before, the specific design of a thumbnail and its implications on the effectiveness of search and exploration processes is not subject of this thesis. It is expected that a good thumbnail design is highly user and task specific, e.g., starting with basic formatting parameters as the correct font size, which depends on a user's eye sight capabilities and influences the overall size of the thumbnail, which, again, influences the number of thumbnails that can be shown at a time or the amount of overlap between thumbnails.

Further information about each paper can be found by inspecting an additional information panel, see Figure 32 on the right side. When a user hovers the mouse pointer over a thumbnail box of a paper, all available meta data about that paper is visualized inside of the information panel, including a paper's title, the full list of authors, its abstract and a preview image of the first page.

In addition, the user is given the option to change the number of k-nearest neighbours or search results being presented at a time by a simple input field, and the grid regularization can be enabled by a tick box, see Figure 32 on the top right.

User Study Design

The user study was designed as an open online study specifically addressed to mailing list subscribers of the ISMIR conference. Invitees were asked to try out the prototype however they pleased, and then, submit a questionnaire. However, the prototype was published online without any access control. In theory, any person could have participated in the user study. Since most participants were not expected to be familiar with the user interface design and the concept of map-based exploration, a short introductory page was presented upon first visit of the prototype, explaining the fundamental idea of visual berrypicking as well as user interface elements, similar to the illustration shown in Figure 33. Then, users could freely interact with the prototype. There was no task or further help. At any point in time, users could click on a link to visit an online form allowing to submit a questionnaire. In comparison to the previous user study of Section 11.1, this open task design was chosen in order to facilitate exploratory search behaviour. On the other hand, due to a missing task, there are no performance metrics that could be collected. Instead, the user study solely is based on qualitative user feedback and simple usage logs. In order to neither overwhelm nor scare of potential study participants, the questionnaire was designed highly anonymously and contained only a few statements that could be rated on a 5-point Likert scale. Additionally, a user's search session id was disclosed, allowing participants to remove it in case they do not wish that their search behaviour and questionnaire submission can be linked together. Finally, the questionnaire also included a full text input for potential comments or other feedback.

As mentioned before, by allowing users to follow a traditional keyword-based search pattern via the search query input, the prototype technically allows to analyse whether participants would prefer a traditional query focused search over a map-based search and exploration. For each user session, a log file is generated, which contains many interaction a user is performing, including search queries, clicks on papers causing map navigations, changes to map visualization parameters, and more. Unfortunately, due to time constraints and the underestimated reluctance of invitees, there were not enough study participants to draw statistically significant conclusions. Additionally, since there was no specific task, some participants only spend a very short amount of time interacting with the prototype before completing the questionnaire. As a consequence, the log data did not allow for a robust quantitative analysis of search and usage behaviour. In the next Section 11.3, both the prototype and study design is modified to improve the quality and quantity of log results: on the one hand, a more accessible data set is used, such that a larger user basis can be invited. On the other hand, the next study is designed as a guided tour and builds upon a specific exploration task that will facilitate longer interaction sessions.

User Study Results

In total, only 8 questionnaires were submitted. Half of the participants did not include their search session id, i.e., their usage be-

Statement	++	+	0	-	- -
Compared to these tools, the ISMIR Explorer better helped me to ...					
identify literature related to my topic of interest.	5	3	-	-	-
discover additional related topics.	4	3	1	-	-
efficiently find what I was looking for.	5	2	1	-	-
Overall, ...					
the interface was easy to use.	5	3	-	-	-
the spatial grouping helped me to infer the relevance for unknown papers.	4	2	2	-	-
I was able to effectively change (refine or extend) the topic focus through navigation.	3	3	1	1	-
during navigation, changes in the arrangement of the papers were comprehensible/plausible.	4	1	2	1	-

Table 6: Number of responses for each statement of the questionnaire on a 5-point Likert scale from strongly agree (++) to strongly disagree (- -).

haviour could not be associated with questionnaire results. Seven participants reported to have used other search tools to find relevant papers of the ISMIR conference (3× DBLP[2], 4× ISMIR Paper Search[3], 1× ISMIR Cloud Browser[4]), which supports the hypothesis that most users were in fact subscribers of the ISMIR mailing list. Table 6 provides an overview over the number of responses for every statement included in the questionnaire on a 5-point Likert scale. The first three questions were aimed at a direct comparison of the proposed prototype to any of the existing search tools mentioned above. Most participants confirmed that the proposed prototype is an effective tool to identify related literature and discover related topics. However, one has to keep in mind that some of the compared search tools are designed for a simple lookup search only, specifically DBLP. As a consequence, some users might have been easily satisfied by any search tool that provides even the slightest more options than a simple lookup search. The other four questions were aimed at the overall usability of the prototype, with focus on whether users were able to understand the similarity-based two-dimensional arrangement of papers and whether users were able to navigate the paper collection by the proposed visual berrypicking method. Here, the results do not clearly confirm that all users were able effectively utilize the prototype to their satisfaction. Since the user study was designed without a

2 http://dblp.uni-trier.de/db/conf/ismir/index.html

3 Not available as of 21.01.2022, last version see Internet Archive on 27th March 2019 at https://web.archive.org/web/http://www.ismir.net/proceedings/index.php?function=show_search_form&table_name=papers

4 http://dc.ofai.at/browser

specific task and without a specific comparison baseline, meaningful conclusions can not be drawn. For example, some users might not have fully appreciated or utilized the map-based visualization in case they were already satisfied with an ISMIR specific full text search tool that includes detailed meta data and previews. Due to the simple form-based submission, it also possible that questionnaires with missing search session ids were submitted by participants that may not have used the prototype at all, or one participant might have submitted multiple questionnaires. In summary, the study results confirm that at least some users are able to effectively use the prototype for search and exploration of scientific papers.

In the next Section 11.3, shortcomings of this study design are mitigated by specifically guiding study participants during the user study, by providing every participant with the same exploration task, and by comparing the proposed map-based visualization to an existing search and exploration tool in context of the same exploration task.

Summary

In summary, a qualitative user study design is proposed that aims at identifying whether users are able to use the method of visual berrypicking for effective exploratory search in a real world data set. A prototype was implemented, which demonstrates that a collection of scientific documents can be explored by subsequently navigating aligned similarity-based maps. Unfortunately, due to multiple circumstances - especially the narrow target audience in combination with an open task design - only very few meaningful interaction logs could be recorded. As a consequence, only minimal conclusions could be drawn from these results. Most importantly, study results confirm that related literature can be identified using the proposed prototype, and that the interface was generally easy to use and helpful in judging the relevance of papers. However, a higher number of detailed interaction logs is required in order to draw statistically significant conclusions. The next user study presented in Section 11.3 specifically focuses on the aspect of collecting a large quantity of study results, but also introduces an exploratory search task, such that performance metrics can be directly compared to a state-of-the-art approach.

11.3 QUANTITATIVE EVALUATION FOR MOVIE EXPLORATION

Since exploration involves learning about an information space, which is a highly user-specific process depending on, e.g., prior knowledge, user abilities, current mood, and more, an objective evaluation is difficult. Furthermore, when inviting study participants to a dedicated lab experiment, users might feel pressured to meet perceived performance goals, which may introduce bias. In the following, a user study design is proposed which aims at measuring exploratory search effectiveness in a casual exploration scenario, where users are still guided towards solving a specific exploration task, but do not feel pressured since they can participate online through an unsupervised web interface.

Aim

The aim of this user study is to validate the method of visual berrypicking in direct comparison to an existing traditional search and exploration user interface. Based on the experience gained in previous user studies described in Section 11.1 and 11.2, the goal was to implement an online study, which allows to efficiently collect a large number of results, and at the same time motivate study participants to follow a specific exploration task, which would allow to draw statistically significant conclusions about the effectiveness of the visual berrypicking method in comparison to a baseline user interface. In comparison to the ISMIR study discussed before in Section 11.2, which used scientific papers of a niche research topic as data set, the following study is based on a large collection of movie data from the *The Movie Database*[5] (TMDb). A prototype allowing to explore movies is expected to appeal to a much larger variety of potential study participants. At the same time, movie data contains many features, e.g., the title of a movie, the plot description, the list of actors, a list of movie genres, and so on, which provides a much more complex exploration scenario than previously introduced by the artificial dataset discussed in Section 11.1. In addition, most users will be somewhat familiar with the data, but most likely not know all available movies, which further motivates an exploration process. Finally, TMDb provides a suitable online user interface that both combines search and exploration components via traditional interaction strategies and, thus, can be used as a comparison interface. The results of this work were published in [pub:10].

Related Work

Many media browsing and retrieval systems exist. The conference Multimedia Modelling hosts an annual competition evaluating video retrieval systems in the context of known item search. Even though known item search does not compare to an exploratory search scenario, due to the large scale of the video collection and sometimes ambiguous queries, exploratory components are implemented as part of the video retrieval systems. For example, in [84] the user interface allows to specify a wide variety of query parameters, including keywords, but also colour, face and text sketches, which will filter results based on where a face or text is found inside the image. Rich and intuitive query formulation modalities support information exploration, since users will be able to more efficiently specify their information need even if it is vague. In contrast, in the following user study, the content of movies is not analysed. Instead, the prototype allows to explore a collection of movies based on meta data, e.g., its title, actors and description. Recently, [94] propose to generate a global map of movies and evaluate different arrangement concepts, i.e., a fully random map

5 `https://www.themoviedb.org/`

compared to a clustered map-based on movie genres. They use a similar evaluation strategy based on a watch list of movies, such that results are somewhat comparable to the results discussed below. Still, the following prototype visualizes a small subspace of k-nearest neighbourmovies depending on a current seed movie or search query in a similarity-based map combining multiple features, e.g., genres, actors and plot description.

Dataset

The data set consists of the 10,000 most popular movies from TMDb. The popularity of a movie is provided by TMDb itself, and is based on various usage information of their website, e.g., the number of views for the day for each movie. Even though TMDb provides many information about a lot of movies, e.g., a movie's total production budget, not all information are available for every single movie. After careful consideration, only the following features were retrieved through the open API provided by TMDb: a movie's title, cover image, a list of genres, a list of actors, a list of directors and a full text plot description. In case a popular movie did not include any of the above features, it was omitted, and a less popular movie was added instead. In addition, the IMDB rating was retrieved for each movie by accessing the open API provided by the *Internet Movie Database*[6] (IMDB).

Combined Similarity Metric

The final similarity measure is derived from a weighted linear combination of individual distance measures for each feature: the linear difference in release date in days, the jaccard distance of the set of genres, the jaccard distance of the set of directors, a modified jaccard distance of the list of actors and the tf-idf cosine distance of the plot description after stemming. Since the data contained the full cast of a movie, often including many dozens of actors, the jaccard distance was modified to give a higher weight to matching lead actors over matching supporting actors of two movies.

In order to provide a fast nearest neighbour search and at the same time allow to adapt feature weights during a live search session, pairwise distance matrices were calculated and stored in advance for all features. Of course, such an approach is only feasible for a limited number of movies, since pairwise distance matrices grow quadratically with the number of objects. For larger datasets approximate nearest neighbour methods can be used, e.g., locality-sensitive hashing [38]. Additionally, all distance matrices were normalized to a maximum distance of one, such that weights of a linear combination of individual distances can be specified without having to consider the range of a distance function and feature, e.g., in case of the linear distance of the release date.

Still, each feature exhibits a different distribution in pairwise distances. As a consequence, some features have more influence on the combined distance function even when choosing equal weights between all features. This is especially obvious when comparing fea-

6 `https://www.imdb.com/`

tures consisting of few discrete values, e.g., the movie genre, with more detailed features, e.g., the plot description. If movies of different genres are contained in the set of k-nearest neighbours, the resulting map projection is often primarily clustered by genre, since the Jaccard distance between two movies of different genre is relatively high due to the small total number of genres. Other features provide less *mass* to the combined distance. Unfortunately, user might get confused when observing map visualizations dominated by features which are potentially weighted lower than others. After careful consideration, this effect was mitigated by selecting fixed feature weights based on preliminary test trails, and disabling the option to change feature weights live during a search session. Still, even with fixed weights, a simple linear combination is not sufficient to completely avoid the problem of feature dominance. In the future, other normalization methods could be considered, e.g., quantile normalization. Alternatively, the set of movies that is visualized could be determined by combining the results of multiple k-nearest neighbour searches for each feature individually. On the one hand, this would ensure that each feature contributes the same number of nearest neighbours and, thus, is fairly represented. On the other hand, some features are only relevant in context with other features, e.g., in most search and exploration scenarios the release date alone would not be considered sufficient in order to compare movies.

A downside of fixed feature weights is that k-nearest neighbours might form a cluster of highly similar movies, whose nearest neighbour sets all match the same k movies, e.g., all "James Bond" movies only might have other "James Bond" movies as their nearest neighbours. In this case, there would be no way to escape this navigational dead end, except when allowing to change feature weights dynamically during the exploration session. In practice, this problem could be partially observed for the proposed movie data and similarity metric, but not in its most extreme case of all $k = 30$ nearest neighbour movies having the same movies as nearest neighbours.

User Interface

The user interface is again based on the prototypes discussed before in Section 11.1 and 11.2. An illustration can be found in Figure 34. Movies are arranged in a two-dimensional map using the proposed visual berrypicking method by applying MDS and Procrustes analysis. The number of k-nearest neighbours was fixed to 30. Grid regularization was permanently enabled as described in Section 9.3. The first set of k movies that is presented to each user upon first visit was specifically selected to match the movies shown on the TMDb website at the time of the user study. Thumbnails consist of a simple cover image provided by TMDb, the title of the movie, and a yellow bar representing the IMDB rating of a movie. Similar as before, the user has the ability to do a full text search on the title, directors, actors and plot description of all movies via a separate query input at the

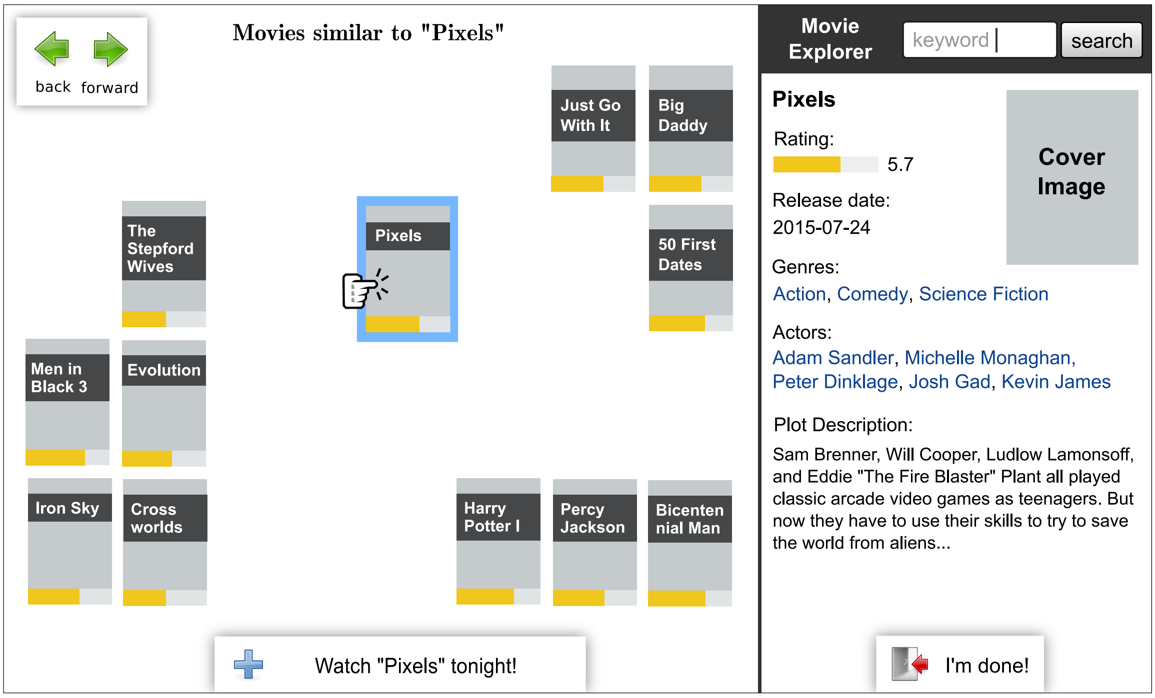


Figure 34: Illustration of the movie explorer user interface including study controls used by study participants: Movies are arranged according to their similarity score. Clicking on a movie (highlighted in blue) will select a new movie at the current seed object and trigger a new nearest neighbour search. Movie cover images were omitted due to licensing concerns.

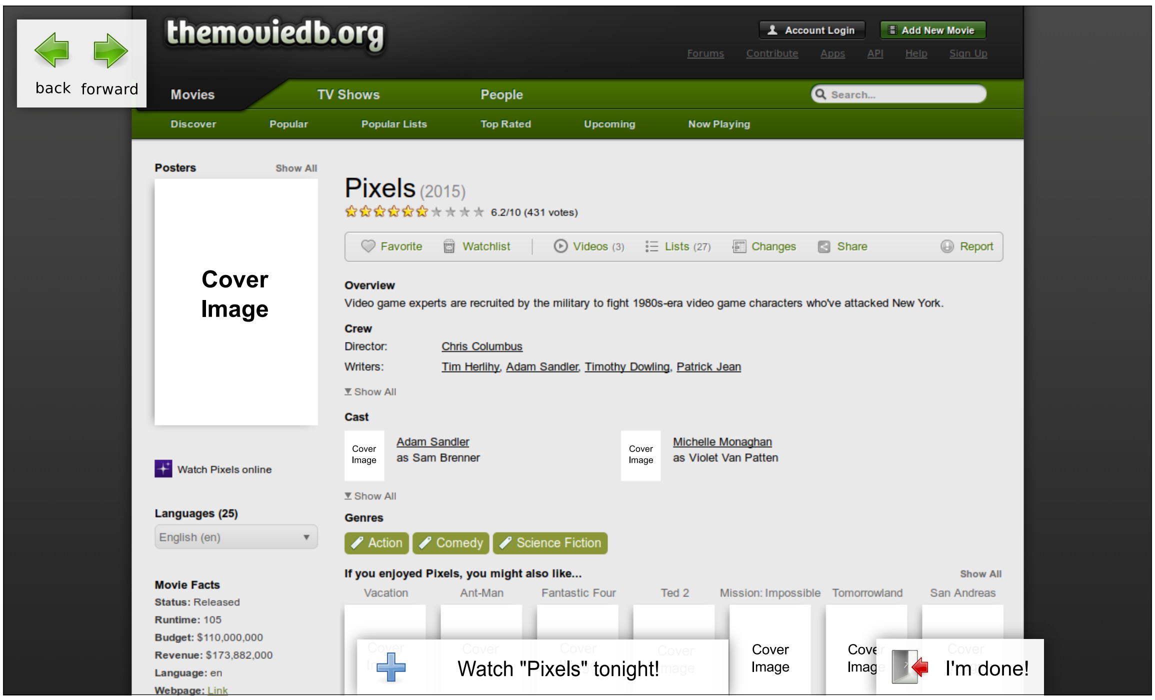


Figure 35: Movie detail page of the comparison user interface TMDb including study controls. Cover images were omitted due to licensing concerns.

top right. When searching, the top k-most relevant search results are shown instead of nearest neighbours.

An information panel on the right provides detailed information about each movie. When hovering the mouse pointer over a thumbnail, the information panel will show all meta data of that particular movie. In addition, genres, directors, actors and words in the plot description, which appear in both the current seed movie and hovered over movie, are highlighted in bold. This allows users to quickly get a better understanding of the meaning of clusters. Users may scan over multiple movies of a cluster, and observe that certain features, e.g., an actor, occur in all of them. Additionally, individual genres, actors and directors are presented as hyperlinks. Clicking on a hyperlink will result in an automated search for movies of that particular genre, actor or director.

In addition, the prototype implements the feature of zooming as described in Chapter 10. Users may change the current zoom level by moving a slider from left to right. At its highest setting, i.e., fully zoomed in, the behaviour of retrieving only k nearest neighbours is preserved. At lower zoom levels, the number of nearest neighbours being retrieved is increased by a factor of $z > 1$, and then, reduced again to k representative movies using a deterministic sampling algorithm, which maximizes the minimum distances between items, see [91]. Unfortunately, even such a simple sampling strategy requires $\mathcal{O}(z \cdot k^2)$ run time, which does result in significantly higher response times for large values of $z \gg 1$ using the proposed study prototype. Due to these high response times, the zooming feature was disabled for the following user study.

TMDb Interface

The comparison interface follows a traditional page-based browsing approach. A modified screenshot can be found in Figure 35. On its start page, it lists the current most popular movies in a vertical arrangement. However, other sorting or filtering criteria can be chosen, e.g., the top rated movies, or movies starring a specific actor. Additionally, a full text search is provided, which lists matching movies in the same kind of vertical arrangement, presumably by sorting them with respect to their relevance to the input query. When clicking on a movie, a separate page is loaded, which presents all available information for that movie similar as the information panel of the proposed prototype. However, many more properties are included if available, e.g., the total production costs of a movie. Also, lead actors are illustrated by a portrait photo. Additionally, at the bottom of each detail page, a list of related movies is presented as recommendations. Unfortunately, it is not known how TMDb does generate these recommendations.

Study Design

The study was designed around a single exploration task, which was later adopted by [94]. Participants were asked to explore the collection of movies in order to find movies they would consider worth watching that day tonight. Each relevant movie that would be

helpful	I have the impression I was able to find interesting movies.
easy	I thought the interface was easy to used.
complex	I found the interface unnecessarily complex.
intuitive	I found navigation using this interface intuitive.
inconsistent	I thought there was too much inconsistency when new movies were shown.
interesting	I was able to find interesting links between different movies.
random	The movies presented seemed random.

Table 7: Evaluation statements that were rated according to a 5-point Likert scale by all participants immediately after using an interface.

identified by using either the proposed prototype or the comparison interface was supposed to be added to a personal watch list. Both interfaces were presented in succession, but in random order for every participant. For each interface, participants were asked to rate a selection of statements on a 5-point Likert scale, and provide additional comments if applicable. At the end, participants were asked to select their final movie favourites from their personal watch lists.

It is hypothesized that participants on average would be better equipped to solve such an exploration task via the proposed map-based user interface, since it directly facilitates exploratory search behaviour, which supports serendipitous discoveries. As a consequence, it is expected that more movies are being added to watch lists while using the proposed prototype, and more participants chose their final favourites from the set of movies that were found while using the proposed map-based prototype. In order to prevent bias, participants were asked to write down and exclude all relevant movies they could easily remember before using any of the interfaces.

This complex study design was enforced by implementing a custom step-by-step online interface. It guides every participants through all steps, beginning with prompting for demographic information, e.g., the age, gender, current profession and a self-assessment of general movie knowledge on a 5-point Likert scale. Every user input was validated live through simple pre-defined rules. In case a user entered invalid information, e.g., a negative age, the study interface would highlight the incorrect values and prompt users to correct their input. Following demographic information, participants were introduced to the exploration task of finding movies they would consider worth watching. Then, a short description of the study controls was presented. It consists of simple overlay buttons for adding or removing a movie from their personal watch lists, and exiting the search session if satisfied. An illustration can be found both in

Figure 34 and 35 at the bottom. At last, users were asked to switch their browser windows to fullscreen mode in order to improve the overall usability of both interfaces and avoid unnecessary distractions through other browser tabs or programs. Finally, users were given 10 minutes of time to use either the proposed prototype or comparison interface. The exact interface was determined randomly. Unfortunately, technical limitations required that each study participants would be given a truly random first interface, which resulted in an imbalance in the total number of participants starting with either of the interfaces. After 10 minutes of time, users were presented a simple hint requesting to continue with the study by clicking on the exit button. However, users were allowed to continue using both tools as long as they pleased. After each interface, participants were asked to rate seven statements targeting the overall usability of each interface on a 5-point Likert scale, see Table 7. After both interfaces were rated, participants were asked to select their final favourites from their personal movie watch list. A complete illustration of every single step is given in Appendix A.2.

Study Results

In total, 110 people completed the online user study from beginning to end. Many more people have partially completed the study. Due to the unsupervised online design results are unsurprisingly highly diverse and had to be filtered rigorously. In the following, only fully completed studies are considered.

On average, participants were 31.8 years old, with 48 being female and 62 being male. Their self-assessed movie knowledge was rated 2.8 on average, with standard deviation of 1.1, where 1 corresponds to the statement of being an occasional movie-goer, and 5 being a movie expert. Most participants are either students, researchers or general staff at universities. However, their background highly varies: history, linguistics, neuroscience, biology, political science, computer science, and others.

In total, 47 participants started with the proposed map-based interface, 63 with TMDb. As mentioned before, due to technical limitations, it was not possible to even out both scenarios, because each study process performed a truly independent random choice. In order to be able to do a fair comparison and draw fair statistical conclusions, results of 16 random participants that started with TMDb were removed from the following analysis.

Figure 36 shows aggregated ratings for all usability statements of Table 7. Most prominently, a strong bias towards the first interface can be identified, which highlights the importance of randomizing the order of interfaces. On average, participants spent 7:09 (min:sec) using the first interface, compared to 4:56 for the second interface. When comparing the proposed map prototype with TMDb, no clear winner can be identified, but some statements show significantly different ratings. Most prominently, the map interface was rated to be better for finding interesting links between movies, which indicates that users

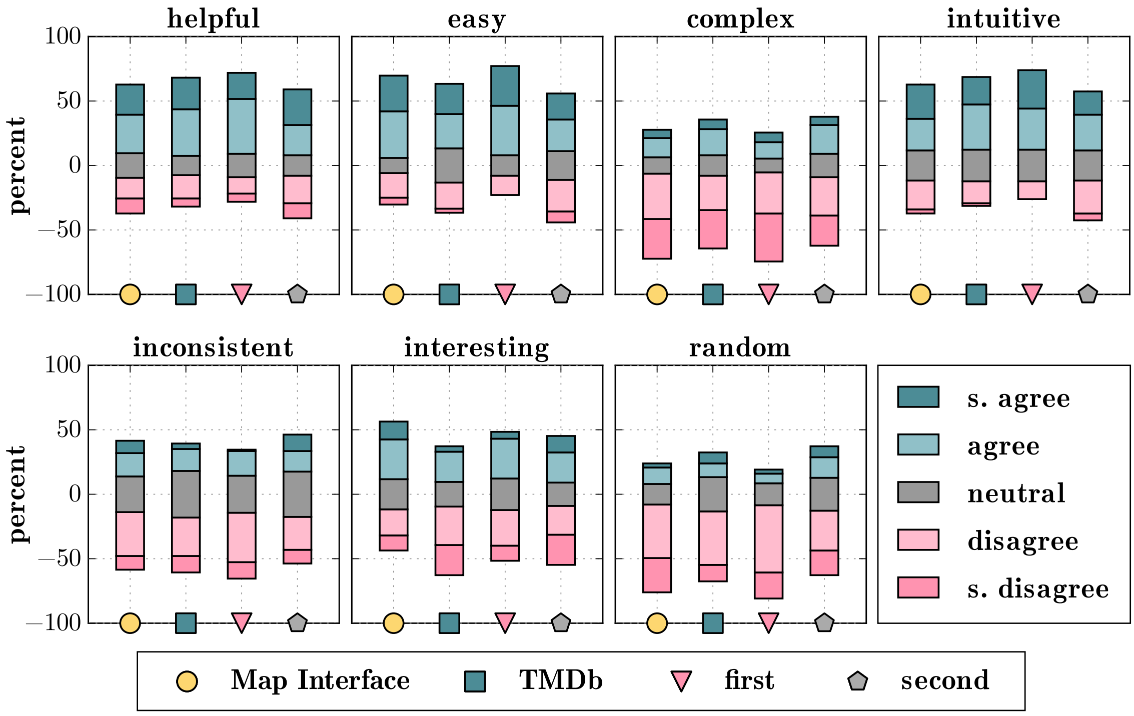


Figure 36: Usability ratings on a 5-point Likert scale for each statement of Table 7 as stacked bar chart both for the proposed map interface and TMDb. Order bias can observed by comparing ratings of the respective first and second study interface.

are able to effectively use the proposed interface for an exploration task. On the other hand, the map interface was rated slightly less helpful in finding interesting movies, which might be explained by the fact that TMDb presents many more information about a movie, and thus, might better appeal to users when deciding whether a movie is relevant or not.

Even though participants were explicitly asked to maximize their browser windows, such that both interfaces can be used to their best potential, a total of 67 participants chose to ignore this request although a red cross was displayed, indicating that their browser size is considered too small. At the same time, when correlating both browser size and usability ratings a strong positive correlation can be identified for the proposed prototype. For example, the map-based interface was rated less complex (Pearson correlation coefficient $\text{pcc} = -0.12$) and more helpful ($\text{pcc} = 0.16$), see Figure 37. On the other hand, TMDb was rated more complex with increasing browser size ($\text{pcc} = 0.25$). These results are consistent with the expectation that users would benefit from large browser sizes when using the map-based prototype, since movie clusters can be identified easier on large screens. However, a large screen should have no significant influence on the usability of the TMDb interface.

There is a strong correlation between a participant's usage time of the map prototype and the perceived randomness of movies shown

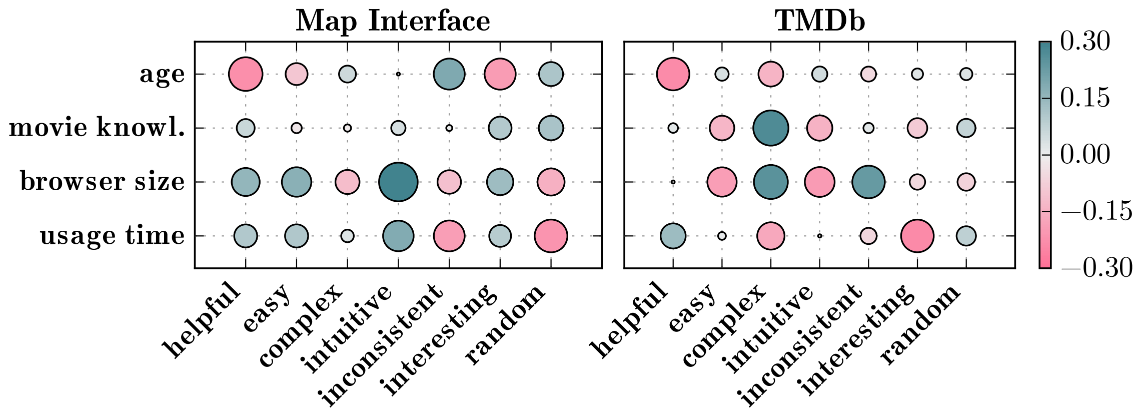


Figure 37: Correlations between study parameters and usability ratings for both the map-based and comparison interface (TMDb). Disc size and colour saturation represent strength of a correlation, hue represents its direction.

by it, suggesting that users needed some time to get familiar with the concept of the interface.

Finally, participants added 5.08 movies (standard deviation $\mathtt{std} = 4.45$) to their personal watch list while using the map-based interface, and 5.77 movies ($\mathtt{std} = 5.17$) using TMDb. Similarly, in participants in total chose 112 movies as their final decision from the map interface, and 173 movies from TMDb. As discussed before, many circumstances may have contributed to this final result, e.g., participants that use small browser windows might have struggled with the map-based interface, participants being more interested in movies discovered through TMDb because more meta data was presented, or because users required more time to get familiar to the proposed interface concept. In summary, these results do not confirm the hypothesis that users would be able to find interesting movies more easily via the proposed map-based interface. However, there are also multiple indicators that suggest that the proposed map-based interface can be effective for movie exploration.

In [94], the authors follow a very similar approach for evaluation, i.e., users were also asked to add movies to their personal watch list. Surprisingly, users added slightly more movies to their watch list if movies were arranged randomly on a global map in comparison to a clustered arrangement based on genres. On the other hand, as [94] suggest, users might find more interesting movies using a random presentation, since the randomness forces users into a less structured exploration, which could increase the likelihood of serendipitous discoveries.

For the proposed movie exploration prototype, user comments included various general usability issues and ideas for improvement, e.g., one participant reported that long movie tiles would cover most of the available space of a thumbnail, and thus, not allow to see the actual cover image of a movie. Another participant suggested to add further categories like "Oscar nominated", which then could be used for filtering. A few participants reported that they were unsatisfied with the increasingly "narrower" movie selection of the map-based interface, because they perceived it to become more and more specific after each navigation interaction. Apparently, they were under the impression that every click would somehow contribute to an underlying recommendation model, which would adapt itself more and more to their interests. Of course, this is not the case when retrieving k-nearest neighbours based on a static similarity score. Many users reported that they wished they could follow multiple interesting movies without navigating back and forth. Unfortunately, movie thumbnails did not include bookmark and pin buttons as introduced in Section 11.2, which could have provided that functionality. Bookmark and pin buttons were excluded to reduce the overall complexity of the interface such that users would not be overwhelmed when first using it.

Summary

In summary, this work presents a comprehensive online user study that investigates whether the method of visual berrypicking allows for an effective exploratory search in a large collection of movies. An advanced user interface prototype was developed, which supports both full text search, simple facet-based filtering, and map-based exploration. Movie exploration has proven to be a popular scenario that attracts many interested study participants, who are willing to invest a significant amount of time and provide detailed user feedback. The study compares the proposed exploration concept with a traditional state-of-the-art movie browsing interface. Participants are asked to solve an exploratory search task, which involves identifying interesting movies users are eager to watch. Study results are mixed. Some indicators favour the proposed map-based interface. For example, user ratings confirm that the map approach can be used to find interesting links between movies, which supports the main hypothesis that movies can be effectively explored using the proposed method of visual berrypicking. Other indicators favour the baseline interface, which may be partially attributed to a more detailed presentation of movie information, including additional meta data, e.g., the total production costs of a movie, and additional illustrations, e.g., actor thumbnails. In total, no clear winner can be declared. However, the vast majority of users were able to successfully use to map interface, discover interesting movies, and add them to their watch lists. Given that most users have most likely never used a similar user interface before, they were quickly able to comprehend the interaction design and tackle a complex exploratory search task. As a consequence, it can be concluded that the proposed concept of visual berrypicking is indeed a valid approach for exploratory search in a large collection of movies. In the future, further user studies might reveal how users interact with the proposed map-based system over a longer period of time, whether they need more time to get fully accustomed to the novel interaction approach, or whether new behavioural patterns emerge with time.

11.4 TOWARDS INTENTION DETECTION USING GAZE TRACKING

When interacting with traditional keyword-based search systems, users are expected to express their own information need in terms of keyword queries. In case of exploratory search, which assumes a vague and evolving information need, this translation process can be challenging. Users might struggle to formulate suitable queries that precisely describe their current information need. This often applies to search scenarios that involve a user's individual taste or preference, e.g., when looking for interesting movies. The exact properties and criteria of interesting movies can often not be precisely specified. As a consequence, interaction methods or feedback mechanisms need to be found that are both easy to use or apply in context of vague information needs, and at the same time expressive enough to eventually be able to clearly separate relevant from non-relevant information.

The movie exploration prototype discussed in Section 11.3 deals with this problem by providing an initial selection of popular movies, and allowing users to navigate the collection of movies by simple browsing interactions, i.e., clicking a movie thumbnail, which is considered to be a very low hurdle to engage users in an exploratory search process. Then, users can subsequently navigate towards sub-areas of movies that are perceived to be more interesting or relevant. Unfortunately, depending on a user's subjective perception of movie similarity, there might exist no sub-area or neighbourhood that actually contains only interesting movies. Then, the underlying similarity function would not conform to user preferences.

Aim

The aim of this work is to support users during exploratory search by adapting search or visualization parameters, e.g., feature weights of a similarity function, based on user interests inferred from the user's own search behaviour. A common strategy is to analyse relevance feedback, e.g., by finding correlations in movie properties of movies that have been recently clicked. However in this study, biometric indicators are investigated as a form of passive relevance feedback. As a first step, a preliminary lab experiment was conducted that evaluates whether eye movements and pupil parameters can be suitable detectors that potentially allow to draw conclusions about the user's interests. The experiment is based on an eye tracker system that allows to record eye movements and pupil dilation while users perform a search task using the map-based prototype presented in Section 11.1. Based on suitable detectors, an assistive search system can be envisioned, which tailors the search process to each user's individual and evolving information need based on additional information gained by tracking the user's eye movements and pupil parameters. For example, in case of the proposed map-based prototype, feature weights of the k-nearest neighbour search could be adapted during the search process based on preferences deduced from correlating varying pupil dilation and object parameters. Preliminary results of the experiment were published in [pub:9].

Related Work

In the past, it has been verified that pupil size not only varies due to a change in brightness of the environment, but also with cognitive and affective load [104]. As as result, recent research investigates the possibility to use gaze parameter as implicit relevance feedback. An overview of the role of eye tracking as a tool for human computer interaction can be found in [85]. Li et al. [79] propose to improve content-based image retrieval systems using real-time eye tracking as implicit feedback to adapt query parameters. Similarly, in [39] eye tracking is used for disambiguation of keyword queries, assuming that gaze and EEG parameter differ between relevant and irrelevant results. In [49], fixation duration and pupil dilation have been identified as promising indicators for relevance feedback. In the following user study, gaze parameters are investigated in the scenario of exploratory search. Recently, Schwerdt et al. [128] investigated the possibility to classify user behaviour into fact finding and exploratory behaviour based on user interface logs and eye tracking parameters.

Study Design

Since at the time of the study, there was no basis for potential correlations between exploratory search behaviour and eye movement or pupil parameters, the study was designed as an exploratory lab experiment, recording all available information provided by the eye tracker system. In order to identify connections between the user's information need and eye tracking data, each user's specific information need has to be known in advance. Therefore, the same search task design as presented in Section 11.1 was used, i.e., users were given a specific target image and asked to search for it in a collection of 2000 artificially generated polygon images. This contradicts the goal of studying exploratory search behaviour, where the user's information need is considered vague and evolving. However, it is hypothesized that correlations observed during a targeted search process may also be observed during an unstructured exploratory search, even if potentially less pronounced.

Figure 38 illustrates the user interface prototype used to perform each search task. It is closely based on the prototype presented in Section 11.1. At the top left of Figure 38, the current target image is presented for each search task. On the right, participants can interact with the prototype by clicking on any of the $k = 30$ images arranged in a similarity-based map. Clicking an image selects a new seed object, which results in an animation that smoothly transitions the current

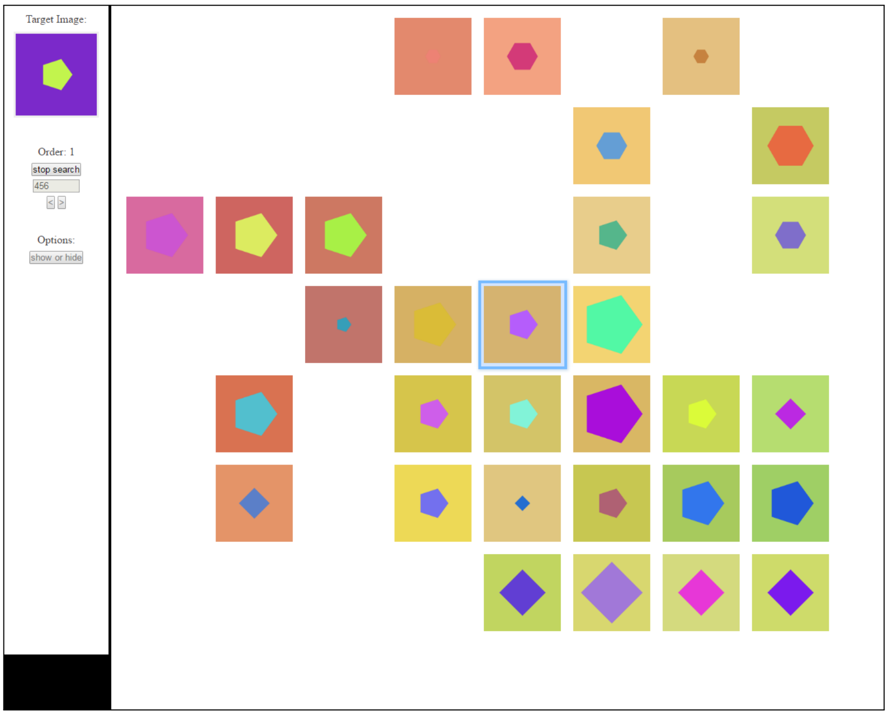


Figure 38: User interface prototype for studying correlations between gaze parameters and search behaviour. On the left, the current target polygon image is presented to the user. On the right, a similarity-based map is shown for $k = 30$ nearest neighbours of the seed image highlighted by a blue border.

and next map visualization. Over the course of multiple interactions, participants are able to select new images which are increasingly more similar to the target image. At some point, the final target image is found, which concludes a search task. For future experiments aimed at combining Electroencephalography (EEG) and eye tracking data, an additional rectangular area was added, see Figure 38 on the bottom left. It switches between a black and white background depending on user interactions, such that the exact monitor response time can be measured via an additional photo diode. However, due to time constraints the experiment was limited to eye tracking only.

As eye tracking device, the *Tobii Pro X2-60* was used. It allows to sample gaze parameters at 60Hz with an accuracy of 0.4° maximum offset to the true gaze position, a precision of 0.3° maximum variance of subsequent measurements to the same gaze position, and a latency of less than 35ms. As monitor, a standard 20″ LCD screen of resolution 1280x1024 pixel was used.

The study was structured as follows. At first, demographic information was collected. Then, the eye tracker was calibrated and participants were introduced to the prototype and search task. Each participant was given the same 6 search tasks. The initial task was not considered and allowed users to get familiar with the prototype.

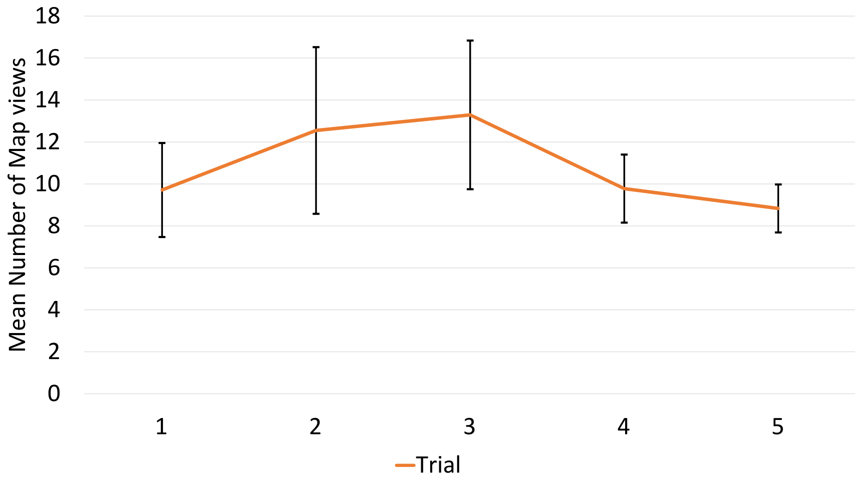


Figure 39: Average number of map views for each subsequent search task (or trial). The decrease in variance may be attributed to users optimizing their own search strategy after an intermediate trial and error phase with higher numbers of navigation interactions.

The order of each task was rotated for each participant in an ascending Latin Square design in order to avoid any bias due to varying difficulties of each task. After completion, participants were asked to rate seven statements targeting the overall usability of the prototype on a 5-point Likert scale. Finally, participants were asked about any comments or remarks. A detailed study description is attached in Appendix A.3.

Study Results

In total, 10 male and 5 female participants completed the user study. All participants were students of computer science. None reported any deficiencies in colour vision. On average, users required 10.8 clicks in order to navigate to the target images, which is slightly more than the average of 7.7 clicks required in the study described in Section 11.1. However, since other start and target images were chosen, which could have resulted in more difficult search tasks, a direct comparison is not conclusive.

Since it was known from prior studies that users are likely to improve their performance from one task to the next, gaze and pupil parameters were analysed for each subsequent task separately. Figure 39 and 40 illustrate the number of seed selections (clicks, or map views) and the time spend on each map (or between each click) for each subsequent search task (or trial). Based on the the time spend on each map, which shows a significant decrease from the first to second task, it can be concluded that most participants quickly got familiar to the prototype and followed a strategy of quick map changes to efficiently find the target image. In the last task, user spent slightly more time on each map, which suggests that users might have reversed their strategy and followed a more thoughtful approach when choosing subsequent seed objects. A similar yet inverted trend can be found in

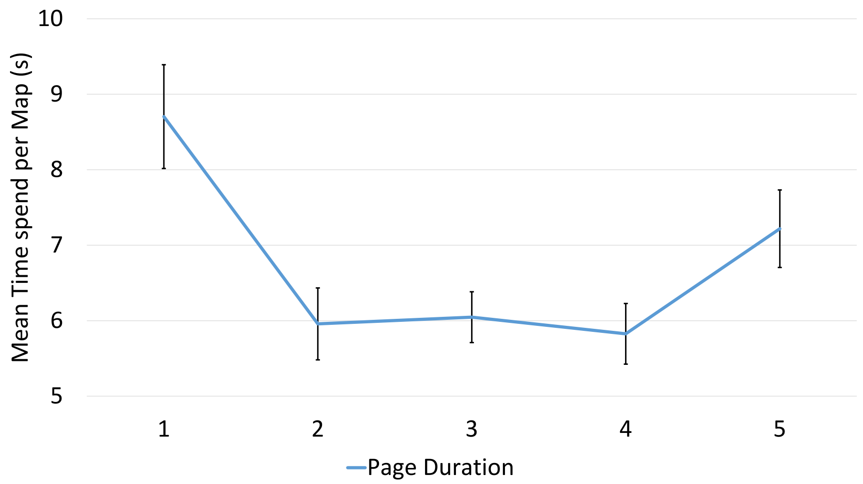


Figure 40: Time spend on each map averaged over all study participants for each search task. The intermittent decrease at task 2, 3 and 4 may be attributed to a trial and error learning phase.

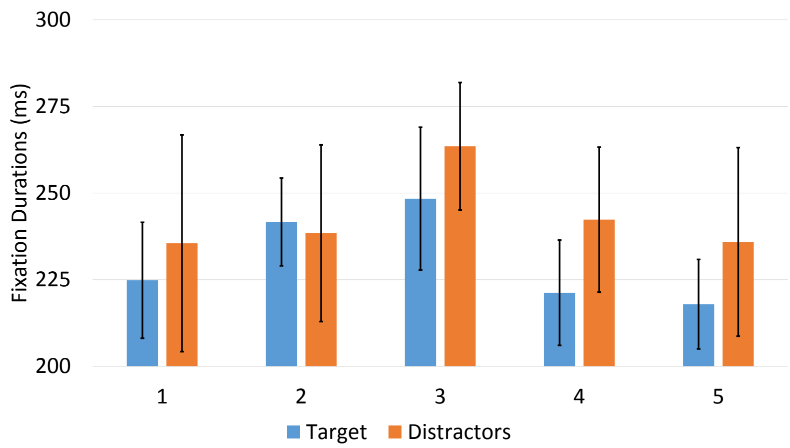


Figure 41: Average gaze fixation duration for both the clicked image (target) and not clicked images (distractors) for all five search tasks.

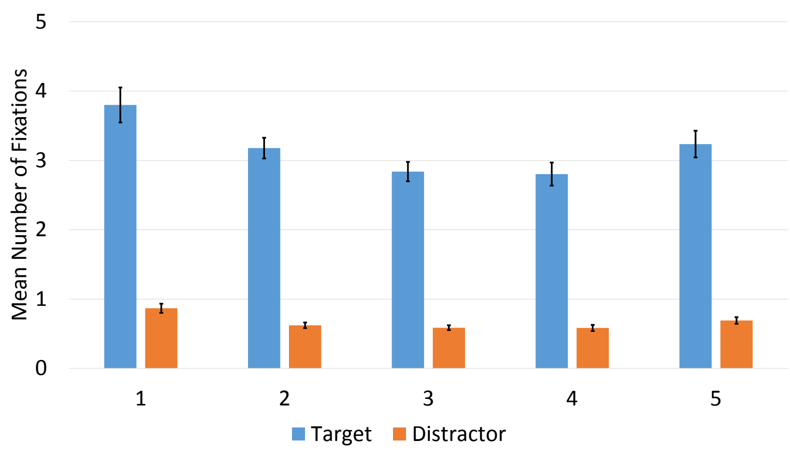


Figure 42: Average number of gaze fixations on both clicked images (target) and not clicked images (distractors) for all five search tasks.

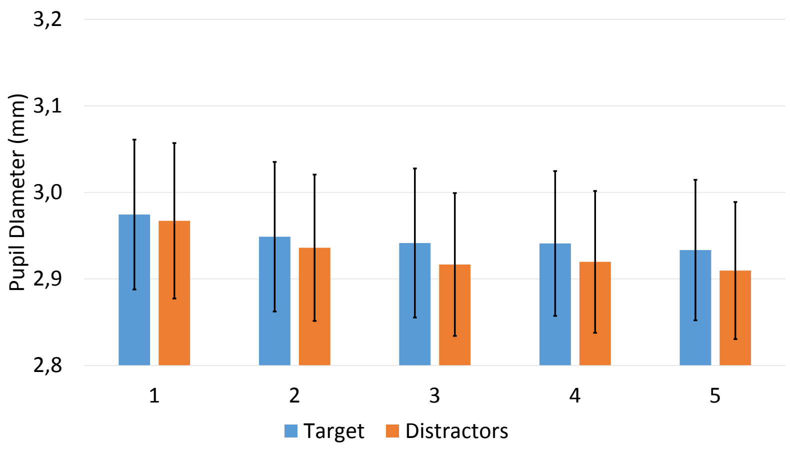


Figure 43: Average pupil diameter for fixations on both clicked images (target) and not clicked images (distractors) for all five search tasks.

Figure 39, where users initially selected fewer (approx. 10 at the first task), than more (approx. 13 at the third task) and finally again fewer maps (approx. 9 at the fifth task).

In order to find suitable predictive detectors that allow to draw conclusions about the user's information need, the difference of gaze and pupil parameters between clicked images and not-clicked images are analysed. Clicked and not-clicked images are referenced as targets and distractors in the following Figures 41, 42 and 43, which illustrate the average number of fixations, duration of fixations and pupil diameter when looking at a target or distractor image for each subsequent search task. Since clicked images are evidently perceived as more promising to users, it is expected that distractor images are on average less often fixated on, that fixation durations are on average shorter, and that pupil diameters are on average smaller in comparison to clicked images. Figures 41, 42 and 43 confirm this intuitive expectation.

On average, users fixated 19.1 times ($\sigma = 10.2$) on distractors, which means that many of the 29 distractors were not fixated on at all. This supports the hypothesis that users are able to draw conclusions about the similarity-based map arrangement of images, which allows them to skip inspecting some images deemed irrelevant solely based on their position on the map.

In the future, gaze and pupil parameters need to be analysed with respect to image features, such that meaningful conclusions about the user's information need can be drawn. For example, there might be significant differences in fixation duration or pupil dilation for images that are more similar to the search task target image with respect to a single feature or a combination of features, which would allow to form a hypothesis about which features the user is currently more interested in and adapt feature weights dynamically during the search process.

Summary

In summary, an exploratory user study was conducted, which aims at identifying potential correlations between gaze and pupil parameters and the user's information need, in order to be able to adapt interaction parameters during an evolving search process. A prototype was developed that allows to merge interaction logs of an exploratory search user interface with eye tracking data, and in the future, EEG recordings. First results indicate that there are correlations between clicked image thumbnails, i.e., temporary relevant information, and gaze and pupil parameters, e.g., clicked images are more often fixated on than non-clicked images. Although these simple correlations are not surprising, they indicate that further research might reveal additional correlations that may be used as passive relevance feedback in exploratory search scenarios. Further analysis of interaction logs and eye tracking data is required.

12 TOWARDS PERSONALIZATION BY DIRECT MANIPULATION

Adapting information systems by analysing user behaviour as proposed in Section 11.4 may be viewed as invasive by some users, since biometric indicators can not be deliberately controlled, and will directly affect what information is shown and how users interact with the system. Most users dislike a loss of control, especially if the rule set that determines how a system is affected by their own behaviour is beyond their comprehension. Of course, the approach discussed in Section 11.4 can always be designed to be subject to user consent, or only generate suggestions and require a final confirmation by users. In the following, a non-invasive personalization approach is proposed, where users have direct control over how their interactions affect system parameters. User are enabled to provide feedback in terms of direct manipulations of information objects in their respective task context, e.g., by specifying the desired rank or score of a search result given a query, or by moving objects in a similarity-based map. The latter is the focus of this section.

Aim

The primary aim of this work is to develop a method for personalized map generation by direct manipulation of objects. Users are enabled to interactively restructure a map according to their own individual preferences. A secondary focus of this work is to investigate the alternative approach to visualize all available information objects of an information space at the same time, which has been deemed computationally infeasible for very large data sets according to the discussion in Section 9.1. This allows to compare the local map-based exploration approach described in Chapter 10 as visual berrypicking with a global map-based approach. Visualizing all objects at the same time benefits the goal of supporting users in exploring information spaces, since users will be able to better discern individual clusters if they are visualized more clearly due to the higher density of examples. Unfortunately, since this work was done only recently, the results were not yet published or scientifically peer-reviewed.

Related Work: Personalized Map Visualizations

Both [116] and [106] present ideas for personalizing map-based visualizations of information spaces. In [116] different layouts can be interactively discovered by moving examples inside the visualization space. Depending on the position of two or more moved objects, suitable axis-semantics are inferred, and all other information objects are rearranged accordingly. Unfortunately, the focus of this paper lays mainly on generating packing layouts for images with different extend such that they are nicely distributed on the screen. Accordingly, the information space was limited to only a few dozens of objects, and maps are generated only by combining two interval-based features, e.g., the size and brightness.

In [106] a different approach is presented. Objects can be moved on a map, and are used to infer a two-dimensional embedding via a semi-supervised least squares projection [102]. In contrast to [116], a linear combination of features can be learned. As a consequence, similar to MDS, there is no specific meaning for both coordinate axes, which impedes users in interpreting the exact position of each object. On the other hand, more comprehensive maps are achievable, which is beneficial when working with high dimensional information objects, e.g., full-text documents.

Semi-Supervised Map Generation via Autoencoder

In the following, even non-linear embeddings are considered. Due to recent advances in hardware accelerated training of large and deep neural networks, maps are proposed to be generated and personalized by learning and optimizing an autoencoder network. It is expected that a multi-layer autoencoder network not only generates a 2-dimensional embedding, but also extracts relevant features from an otherwise low level feature space of high-dimensional information objects, e.g., tf-idf vectors of full-text documents. Therefore, the resulting similarity-based map is assumed to provide detailed insights into the global structure of a data collection. In contrast to the method of visual berrypicking as proposed in Chapter 10, a single map of the full data set is generated. This allows users to observe larger clusters and other global structures in the data. In addition, subsets of the data, e.g. search results, can be directly visualized by highlighting them inside a map. User feedback is integrated by defining a specialized loss function. Users will be able to move a number of document thumbnails on the screen according to their own preferences and perceived similarities. Documents that have been moved, in the following called landmarks, are then considered additional training data that will guide the optimization process towards an embedding that matches the landmarks defined by a user. At the same time, the optimization process is visualized by an animation that illustrates how the map projection evolves over time. User can immediately see how their manipulations affect the map projection. A detailed discussion of this personalization concept is provided shortly.

German Law Dataset

As data set, a collection of 6427 German laws provided by *Gesetze-im-Internet.de*[1] is used. Each law corresponds to a full-text document. Some laws contain only a few, others many hundred of sentences and paragraphs. Additional resources attached to a document are omitted, e.g., images and meta data. Each document is pre-processed by removing German stop words and calculating tf-idf vectors of the 2000 most frequent words.

1 Gesetze-im-Internet.de: `http://www.gesetze-im-internet.de/`

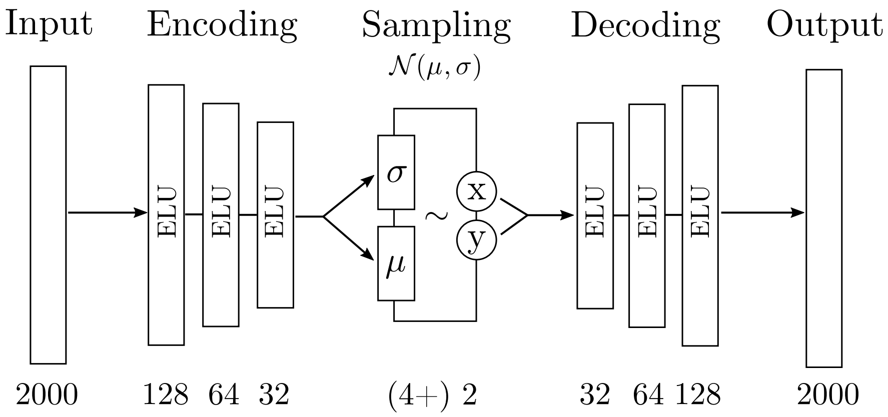


Figure 44: Fully connected autoencoder network architecture: input vectors of size 2000 are encoded and decoded by 3 layers of Exponential Linear Units (ELU); map projection to 2d is modelled as Gaussian sampling process.

12.1 AUTOENCODER NETWORK DESIGN

The autoencoder network was designed on a trial and error basis, and consists of 9 fully connected layers, see Figure 44. The input layer provides the 2000-dimensional tf-idf vectors. Layers 2 to 4 compress each document via 128, 64 and 32 neurons to a 32-dimensional feature vector. Each layer uses a combination of an exponential and linear activation functions, also known as Exponential Linear Unit (ELU). Layer 5, or the encoding layer, consists of 2 neurons and follows a variational design, i.e., activations are sampled from Gaussian distribution whose parameters are determined by the weighted activations of the previous layer and are subject to optimization as well. After training, document positions are determined by the mean of both Gaussians. Given the small training data size of only 6427 laws, following a variational design allows to prevent overfitting in early stages of the training. Layers 6 to 8 consists of 32, 64 and 128 neurons with ELU activations. Finally, layer 9 decodes the embedded data points again to a 2000-dimensional output vector.

Network Regularization

Since screen space is generally limited, the map projection also needs to be limited to fit a fixed range that can be mapped to any screen size. Typically, a min-max normalization is applied on the final embedding. However, when animating evolving embeddings, a min-max normalization would lead to violently jumping map coordinates, since the minimum and maximum may drastically change during the training process. Instead, the activations of the embedding layer are regularized to fit a constant range of $[-1, 1]$. A regularization loss is applied when coordinates exceed the range of $[-1; 1]$. If coordinates further exceed the range of $[-1.1; 1.1]$, they are clipped to that range. This normalization method enables a fluid animation of the evolving map projection.

Optimization and Loss Functions

For optimization of the autoencoder, two loss functions are defined. The primary loss models the reconstruction error of the full autoencoder. It guides the optimization process towards generating a similarity-based map. The primary loss is defined as the mean squared error between input document vectors and reconstructed output document vectors. The secondary loss is applied only when a user has moved at least one document. It is supposed to guide to optimization process towards generating a map that embeds landmarks at their respective positions. Therefore, it is defined as the squared difference between the target position of a landmark and its embedding coordinates. The secondary loss is greater than zero, if the encoder layer activations differ from the user defined coordinates of a document. Documents that have not been moved will not add to the secondary loss.

Dealing with Small Training Size for Map Personlization

Unfortunately, since only very few documents are landmarks in comparison to the total size of the data set, the secondary loss is mostly ignored during training. Mini batches that do not contain a landmark, can no consider them when optimizing. For that reason, all available landmarks are artificially added to otherwise random mini batches. In addition, Gaussian noise is added to the target coordinates of landmarks in order to avoid overfitting.

Limitations of Personalization

In practice, even with these modifications to the training process, the personalization of maps often results in no structural changes of an already trained embedding. After a number of training batches, landmarks are usually embedded in the vicinity of their specified positions, but no structural map changes occur. Only when learning a map from scratch, the secondary loss will have some influence on the optimization process such that an adapted map structure will be generated. In the future, the training strategy needs to be further optimized by adding additional adaptation incentives, e.g., feature weighting of the reconstruction loss.

12.2 USER INTERFACE FOR PERSONALIZED MAP EXPLORATION

In order to improve and evaluate the autoencoder network design, an interactive prototype was developed, see Figure 45. It provides a traditional keyword-based search in combination with a similarity-based map visualization. Documents are shown as small grey semi-transparent discs. A selection of documents is represented via thumbnails listing the abbreviated title of the law. The selection algorithm is based on a stream sampling method that maximizes the minimum distance between thumbnails, see [91]. When clicking a thumbnail, details of the document are shown in a side panel. Users may zoom into sub areas of the map by using the scroll wheel or state-of-the-art

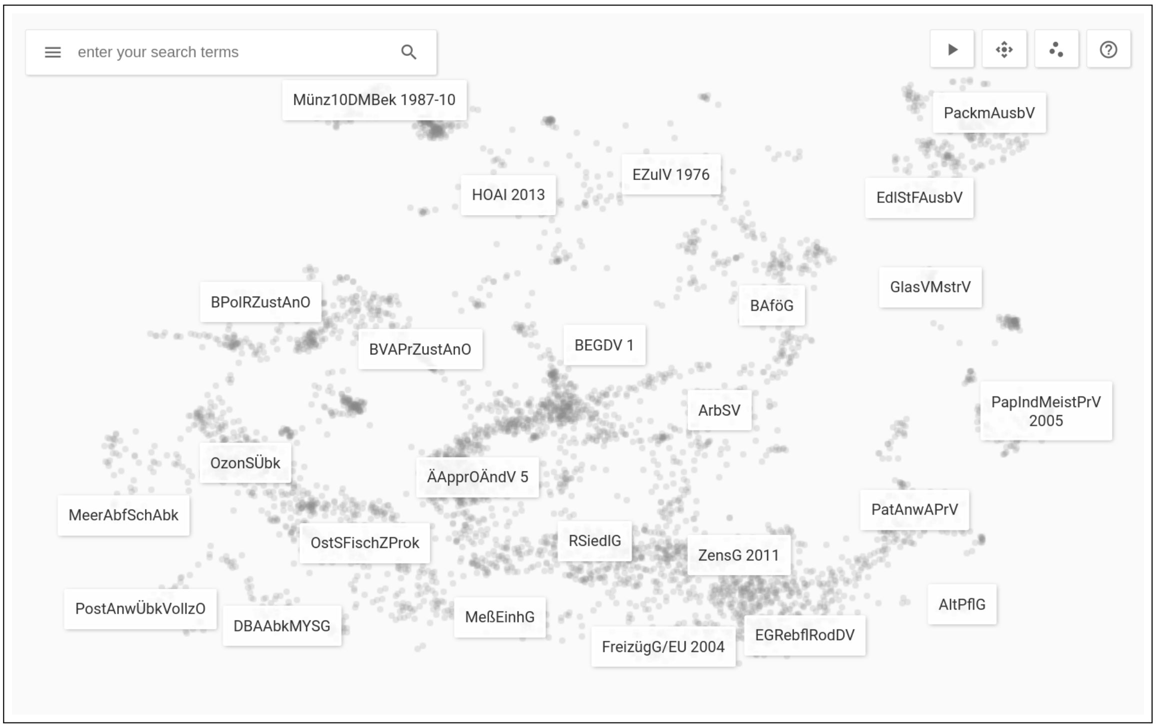


Figure 45: User Interface Prototype showing a map of German laws: each disc corresponds to one document; boxes with law abbreviations can be clicked to open a law, or moved to begin personalization phase; when entering a search query, results will be highlighted, see Figure 46; users may zoom into the map using touch gestures or mouse scroll.

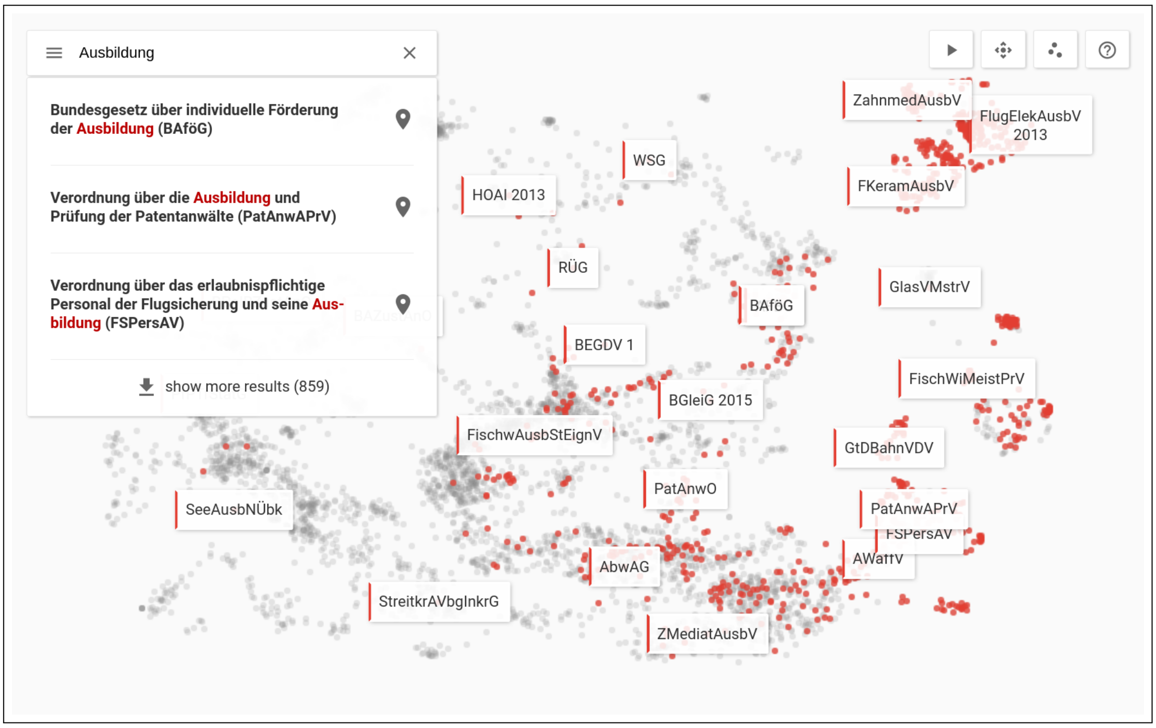


Figure 46: Prototype highlighting search results as red discs and boxes with red border. Users may discover various clusters of search results. First three search results are shown as classic search result list ordered by relevance.

pinch gestures. When zooming, new documents are automatically selected to be represented by thumbnails instead of grey discs. Eventually, when zoomed in to an extent such that only a few documents are visible, all of them will be visualized as thumbnails, and thus, all documents are accessible to the user. This constitutes a major advantage to the local map exploration approach proposed in Section 10, since it has been shown that some objects, called orphans, can not be found by navigating $k-nearestneighbour$ maps.

Animated Training Process

Most importantly, the prototype is able to visualize the learning process by displaying updated encoder activations as an animated map live during batch training. A new map is generated after each mini batch, unless a previous map has already been created within a specified amount of time, such that a certain frame rate of updates can be achieved, e.g., ten map updates each second. Users can start, stop and reset the training process by control elements shown at the top right of the user interface, see Figure 45. During development of the prototype, the animated training process allowed to efficiently optimize autoencoder network parameters via trial and error.

Interactive Map Personalization

Users can move documents by dragging one or multiple thumbnails to their preferred position on the map. After moving a document, the training process is started automatically, and updates to the map will be visualized by an animation. Unfortunately, as discussed before, the proposed training strategy is not yet able to structurally adapt an already trained map projection according to added landmarks. Only minimal changes to a map can be observed. In contrast, when resetting the training process and learning a new map from scratch, landmarks appear to have a more profound effect on the overall map structure. In the future, further investigation is required to improve the adaptability of map projections.

Combining Search and Map Exploration

Additionally, the prototype allows to highlight search results inside the current map visualization, see Figure 46. Highlighted search results will reveal clusters or areas that match the query. Accordingly, users are able to zoom into different clusters in order to inspect individual search results. The ability to both get an overview over the whole data set and inspect search results of different clusters is hypothesized to be an effective method to break out of traditional filter bubbles. In the future, comprehensive user studies are required in order to confirm this hypothesis.

Summary

In summary, this work proposes a personalization method for adapting similarity-based maps to user preferences through direct manipulation. A prototype for document search and exploration is developed, which uses an autoencoder network to generate a custom map projection. First experiments show that the proposed training strategy is not yet capable of dynamically adapting maps to user input. However, thanks to recent advances in hardware accelerated optimization of autoencoders, the prototype demonstrates that global maps showing all available information objects can be efficiently generated. Furthermore, the prototype visualizes search results of a keyword query by highlighting each result in the global map, which is a promising approach to allow users to identify different clusters of search results, and therefore, escape the traditional filter bubble induced by an otherwise linear ordering according to query relevance. In the future, more research is expected to further refine the proposed map personalization approach and evaluate the overall concept of combining global map projections with keyword search for supporting users in the scenario of exploratory search.

Part IV

OUTLOOK

13 CONCLUSIONS

This chapter presents the main contributions of this thesis by revisiting each of the six research goals of this thesis, see Section 13.1. In Section 13.2 a concise summary and final conclusions are presented.

13.1 RESEARCH GOALS

This thesis focuses on developing and evaluating novel approaches to support users during exploration and exploratory search. A total of six research questions are investigated, see Section 1.2. In the following, the results of this thesis are revisited and discussed in context of these research questions.

(RQ1) Identify Opportunities for Supporting Exploratory Search

Research question (RQ1) aims at identifying promising approaches for supporting exploratory search tasks. This goal was achieved by studying current methods of state-of-the-art search systems, analysing user behaviour with these systems in context of exploratory search tasks, identifying problems, challenges and opportunities and developing prototypical solutions. Chapter 5 analyses current search behaviour based on a log analysis of state-of-the-art keyword-based search engines. The results confirm previous insights that many users do not examine search results beyond the first result page. This usage behaviour is believed to induce a strong information bias determined by the ranking of search results. Users are likely to miss out on clusters of results which are deemed less relevant and hidden in subsequent result pages. However, exploration entails the goal to learn about the full diversity of an information space. Thus, approaches are needed that are better suited at representing this diversity of information, and consequentially, better support users during exploratory search. Based on this observation and a literature analysis of state-of-the-art approaches, see Section 3.3, two novel approaches for supporting exploratory search are proposed in Chapter 6 and Chapter 7.

Chapter 6 investigates whether search result lists can be augmented with additional exploratory components such that the user's efficiency at judging the relevance of individual results is improved. Assuming that users are able to more quickly and efficiently identify relevant results, they might be more effective when working on an exploratory search task, since more results can be examined in the same time. In order to evaluate this hypothesis, a prototype is developed that augments search result lists by representing each result not only by title, URL and excerpt, but by highlighting entities from an extendible domain ontology if they are found inside the result web page. Due to this highlighting of domain entities, users can easily associate known concepts with search results, which may help to judge the relevance of a result, e.g., in case the user has already learned about that particular concept and is looking for new perspectives instead. In order to model the user's prior and obtained knowledge, new concepts can be interactively added to the ontology during the exploratory search process. While the proposed augmentation method is considered a promising approach, it does not solve the fundamental problem that list-based visualizations of search results do not convey structural information about the information space, e.g., clusters or outliers, and thus, other visualizations might be more suitable in the scenario of exploratory search. Initial experiments have not indicated that users would significantly benefit from this approach during exploratory search tasks, which is why it was not further studied nor evaluated in detail.

Chapter 7 proposes a visualization strategy that focuses on providing an overview over search results and their relations. Search results are clustered by Formal Concept Analysis and visualized as a lattice structure using a graph visualization toolkit. This lattice structure arranges search results in a concept hierarchy, which allows users to easily distinguish results that belong to different topics or sub-topics. Unfortunately, a known problem in Formal Concept Analysis is that the lattice structure quickly grows in size with an increasing number of diverse items like documents. The resulting graph visualization becomes overwhelmingly complex. As a consequence, the approach was discarded in favour of a map-based visualization of search results, which is the focus of the remaining chapters of this thesis.

(RQ2) Large Scale Map Projections for Exploratory Search

In contrast to list-based visualizations, two-dimensional maps provide an overview over a large quantity of items, revealing their global structure including clusters, patterns and outlier. Therefore, visualizing the set of search results in a similarity-based map is considered a promising approach to support exploratory search. Research question (RQ2) aims at investigating whether there are efficient projection algorithms that would allow to generate similarity-based maps for large sets of search results in acceptable time as part of an interactive search session. In Chapter 9, a number of state-of-the-art methods for dimensionality reduction are reviewed and compared, including principal component analysis, multidimensional scaling, multiple variants of stochastic neighbour embeddings, autoencoder and growing self-organizing maps. Unfortunately, algorithms that achieve a high projection quality also suffer from high computational complexity as well as various properties unfavourable for generating maps during an interactive search session. Promising methods in terms of projection quality, e.g., t-distributed stochastic neighbour embedding or autoencoders, are often based on iterative non-deterministic optimization algorithms, which require multiple trials and a manual tuning of hyper-parameters to achieve acceptable projections, and thus, are expected to lead to user dissatisfaction due to the inconsistency of generated maps. Other methods, e.g., classical multidimensional scaling, provide a deterministic closed form solution, but suffer from even higher computational complexity. Accordingly, research question (RQ2) can be concluded with the result that there are currently no suitable projection methods for generating similarity-based maps of search results as part of an interactive search session. In the future, advances in computational efficiency or projection consistency may allow to revisit the approach to generate a global map of the set of search results.

(RQ3) Effective Map-Based Exploratory Search

In the remainder of the thesis, the potential of map-based visualizations to support exploratory search tasks is studied under the assumption that only a limited amount of items can be projected in reasonable time. Accordingly, research question (RQ3) aims at finding an effective method to support exploratory search under this assumption. Chapter 10 proposes a novel interactive approach based on the known concept of berrypicking. In information retrieval, berrypicking describes a model of exploratory search that involves an iterative learning process by examining search results, discovering and learning about new concepts, and repeatedly refining search queries based on that gained knowledge. This strategy is transferred to the idea of using similarity-based maps to support exploratory search, and called visual berrypicking. Users are enabled to examine local maps of search results, learn about new clusters and relations between items, and interactively navigate to neighbouring areas. Multiple incremental prototypes are implemented and evaluated in various scenarios. Chapter 11 presents three user studies targeted at evaluating whether the proposed approach effectively supports exploratory search. All results indicate that users are quickly able to comprehend the proposed interaction design and utilize it to solve classic image retrieval tasks, see Section 11.1, and find interesting links when exploring scientific research papers, see Section 11.2. A large comparative online study revealed that users are able to effectively discover movies, see Section 11.3. A novel evaluation strategy for measuring the exploration performance based on the concept of a personal movie watch list did not show a significant improvement in comparison to a state-of-the-art movie portal website. However, comparing and evaluating highly complex and highly interactive information systems is an open research problem. Current evaluation methods are mostly based on user satisfaction measures, which may be biased towards known and familiar design patterns. In summary, research question (RQ3) can be concluded with the result that visual berrypicking represents a competitive method for supporting exploratory search using local similarity-based maps, although no significant improvement could be shown.

(RQ4) Map Consistency under Data Changes

Research question (RQ4) investigates the problem that most projection methods are highly sensitive to changes in input parameters, which leads to inconsistent maps if the underlying data changes. However, many real world information spaces evolve over time. Information objects or even clusters of objects may be added, removed or modified. On the one hand, map projections should reflect these changes as accurate as possible, integrating new items and clusters. On the other hand, user might get confused if item positions drastically change, and thus, are not able to transfer knowledge gained from examining previous maps. Section 9.2 compares a selection of projection algorithms in the scenario of a growing music collection. Several projection methods are evaluated based on objective measures targeting projection accuracy and average position change of items. In addition, a user study investigates the user's ability to follow animated map changes when adding new songs. The results show that multidimensional scaling in combination with Procrustes analysis is preferred by users, and achieves a good balance between projection accuracy and map consistency. As a result, research question (RQ4) can be concluded with the recommendation to use multidimensional scaling in combination with Procrustes analysis, which was implemented in the previously mentioned prototypes for evaluating visual berrypicking.

(RQ5) Limitations of Map-Based Exploratory Search

Every new method is inherently accompanied by some disadvantages or limitations. Research question (RQ5) aims at identifying additional limitations of the proposed map-based approach to exploratory search. Some limitations are well known or have already been discussed, e.g., the limitation on the number of items that can be efficiently projected in an interactive search session due to high computational costs, or the limitation that high-dimensional data can not be projected to two-dimensional space without loss of information. However, Section 10.3 investigates the open problem referred to as the Hubness Phenomenon, whose origin has not yet been fully understood, but is relevant to the proposed visual berrypicking method, since local maps are generated based on a k-nearest neighbour search. It is shown that hubs, i.e., items that occur unusually often amongst the k-nearest neighbours of other items, are not an artefact of high dimensionality, but can be associated with density gradients. Hubs do not occur in high-dimensional data that exhibits no density gradients, e.g., data equally distributed on the surface of a sphere. Also, it is demonstrated that hubs can be observed in two-dimensional data. The proposed map-based exploration method is affected in the way that users will likely notice hubs as repeating items, since they are contained in more local maps than average. Also, due to this asymmetric neighbourhood distribution, there will be items, called orphans, that do not occur amongst the k-nearest neighbours of other items, and thus, can not be discovered by users at all. In contrast, a global map approach would not suffer from this limitation, since all items of an information space can be embedded. In summary, research question (RQ5) can be concluded with the result that there are additional limitations related to the Hubness Phenomenon that need to be studied further and have a profound impact on the proposed visual berrypicking method.

(RQ6) Personalized Map Exploration

Exploration is a highly user-specific process, which depends on many factors, e.g., prior knowledge, user abilities and even mood. Accordingly, research question (RQ6) investigates the possibility of personalizing exploratory search using either invasive or non-invasive methods. Section 11.4 discusses whether biometric features such as gaze or pupil parameters can be exploited to adapt a map-based exploration process to user preferences, which is considered an invasive method since users do not have control over how their behaviour affects the adaptation. An exploratory user study discovered that there are indeed correlations between gaze parameters and map navigation clicks, most notably fixation duration and pupil diameter. Unfortunately, the results of this initial study could not yet be transformed into a novel personalization method. Further research is needed in order to infer user preferences from gaze and pupil tracking in the scenario of exploratory search. Chapter 10 discusses non-invasive personalization methods as part of the proposed visual berrypicking method. Since map-based projections are based on a notion of similarity, they can be adapted using manual features weighting, which was implemented in two prototypes discussed in Section 10.4 and 11.3. Alternatively, the method of metric learning can be applied, but was not investigated in this thesis. Chapter 12 proposes to use the interaction design of direct manipulations to guide the optimization process of an Autoencoder towards learning a map projection that conforms with user-defined item positions. The autoencoder network is optimized by combining a classic reconstruction loss with an additional error function that measures the squared difference between training examples and predicted embedding coordinates. Since users are expected to only manipulate or move very few information objects at a time, there is an extreme imbalance in training examples. This imbalance is mitigated by artificially introducing examples with know item positions in otherwise random mini batches. First experiments indicate that an already learned map can not be effectively adapted by adding new training examples in an interactive exploration session. However, when learning a new map from scratch, user-specified item positions will affect the learned map structure. Further research is required to improve the adaptability of map projections learned from autoencoder networks. As a consequence, research question (RQ6) can be concluded with the result that there are both promising invasive and promising non-invasive personalization approaches, which need to be studied further to be able to draw conclusions about their effectiveness.

In summary, all of the posed research questions could be answered or at least partially answered. As is to be expected, even more research questions have been discovered throughout this thesis. Further research is required to fully understand how users can be effectively supported during exploratory search.

13.2 SUMMARY

In this thesis, novel approaches that support users during exploration and exploratory search tasks could be developed and validated. Therefore, the main research goal of this thesis is achieved. The most impactful contribution of this thesis is considered the concept of visual berrypicking as an extension to the classic berrypicking model, which combines various methods from the research areas of Human Computer Interaction, Machine Learning and Information Retrieval. Visual Berrypicking uses similarity-based maps of k-nearest neighbourhoods as a means to navigate complex information space. Map-based visualizations are considered a very promising approach to support exploratory search due to their ability to provide an overview, visualize cluster structures and outliers. Recent advances in methods for dimensionality reduction will further allow to improve the usefulness of map visualization when exploring large-scale data. Besides, many additional opportunities for improvement have been identified in this thesis, e.g., considering the adaptability of map visualizations. In the future, more research is required to refine such methods that support exploration and exploratory search tasks. In the following chapter, two approaches for future work are discussed in detail.

14 FUTURE RESEARCH

As is to be expected, this thesis leaves many research questions open that could be studied in the future. A selection of future research ideas are discussed in more detail in the following Sections 14.1 and 14.2. Many of these questions have been shortly touched in previous chapters of this thesis. Unfortunately, the open nature of exploratory search tasks requires detailed and costly evaluation strategies, which could not be performed for all approaches proposed in this thesis. Finding effective evaluation methods for exploratory tasks and systems can be considered an ongoing research problem in itself. The lack of standardized evaluation methods significantly handicaps research and developments in this area. Therefore, further research in the area of evaluating exploratory search systems is considered a vital step in order to improve current and future approaches.

Besides evaluation challenges, there are also a number of opportunities to extend the approaches discussed in this thesis. In the following, two perspectives are discussed in more detail.

14.1 ADVANCED MAP INTERACTIONS

An important part of designing information systems is how users can interact with them. Given that exploratory search is a highly subjective task, a tight integration between users and the exploration system is needed, which can only be achieved by advanced interaction designs. Similarity-based map visualizations provide many opportunities to achieve an efficient and effective cooperation between users and the information system. In the following, a number of advanced interaction techniques are presented, that have already been proposed in other research areas, but whose influence on the effectiveness of supporting exploratory search is not yet understood. Therefore, further research is required to carefully evaluate how these interaction designs can be utilized.

Touch Gestures for Locally Aligned Maps

Due to widespread hardware support in professional and consumer devices, touch gestures are a well-known approach to human computer interaction in many scenarios. For map-based exploration and exploratory search, touch gestures represent an ideal interaction modality, since they closely resemble natural interactions with physical maps. In Chapter 10, which describes the method of visual berrypicking, information spaces are suggested to be explored by subsequently navigating between local maps. The user clicks on a new seed object to jump to a new neighbourhood. The transition between one map and the next is animated, which creates the impression of navigating a large map. However, this panning-like animation is only the result of overlappings between two subsequent maps and can not be naively implemented as a true panning animation. Still, consistently panning locally aligned maps via touch gestures is considered a very promising extension and expected to significantly improve the user's impression of navigating a large map. The same applies to other established digital map interaction gestures, e.g., zooming via pinch gestures, or rotation. The goal would be to achieve fluid animations and interactions when working with local maps.

Tangible Interactions for Map-based Exploration

Another promising interaction metaphor that has not yet reached widespread adoption are tangible user interfaces. Tangible user interfaces use physical objects to enable interactions with digital information systems. Users benefit from interacting with physical objects, since they can represent additional state information, can be naturally manipulated, and are readily accessible. For example, a simple wooden cube and its coloured faces could represent different choices. Users can interact with the system by rotating the cube to choose the option associated with each face. The current state, i.e., the current choice is always visible and easily accessible. Tangible interaction design has been previously applied to map interactions as well. For example, [133] propose to use a simple piece of paper as an interactive zoom lens. Users position a sheet of paper above a tabletop showing a map. A camera detects the location and orientation of the paper. And a projector throws an image of the zoomed-in section of the map onto the paper in real-time. Based on the position and distance between the tabletop and paper, a different zoom levels can be achieved. User get the impression of using an optical lens to zoom into a digital map. Besides this optical metaphor, other information can be shown as well. However, as mentioned before, there is no research yet how such an interaction design would influence user behaviour when performing exploratory search tasks.

Freehand Gesture Interactions

Today, information systems are used in a wide variety of situations and by a wide variety of users. Depending on the situation and user, classic interaction modalities might be available all the time. Some users may be temporarily handicapped, others might be struck by permanent disabilities. In order to support as many users as possible, it is important to consider many different usage scenarios. For example, if a user is not able to directly interact with an information system via mouse or touch gestures, a freehand alternative might be helpful. In prior work as part of collaboration with the University of Rostock, a method for gestures spotting was developed and published in [pub:2]. Gesture spotting describes the problem of detecting gestures from a continuous stream of sensor data without specific start and stop information. The goal is to develop a hand gesture detection method that is unobtrusive, i.e., it does not disturb the user with false positive interactions, but also responsive and intuitive, i.e., hand gestures are reliably recognized when a users actually intents to interact with the system. This can be especially useful, if other interaction methods are not available, e.g., when a worker needs to interact with a touch-enabled device, but work gloves prevent touch gestures from being recognized. Each gesture is described by a Hidden Markov Model. The continuous sensor stream is segmented by assuming that gestures are preceded and followed by short idle states. Unfortunately, the system was not evaluated for map interactions. However, other researchers have investigated freehand gestures for zooming and panning maps, see Stellmach et al. [135].

Novel map interaction approaches may not only allow for more rich and intuitive interactions, but also enable additional exploration use cases and scenarios, e.g., collaborative exploration or exploratory data analysis. Since similarity-based maps are capable of visualizing a large quantity of information objects at the same time, advanced interactions for selecting multiple objects or clusters via the lasso method are conceivable. Users would be be able to select custom subsets or clusters of objects, which then could be analysed and compared for their statistical properties. In contrast, when visualizing search results as a list, selecting a large number of search results would require a lot of manual effort. In order to support collaborative exploration, map visualizations could be extended with the ability to annotate objects, clusters or regions with freehand drawings or text. Users would be able to highlight certain clusters by drawing a bubble around them, communicate interests by marking certain objects, or communicate other information by adding text annotations.

Many more interaction ideas are promising extensions to map-based exploration tools. Unfortunately, given limited time and resources, only a fraction of ideas can be thoroughly studied within a single thesis. More research is required to enable map-based exploration tools to reach their full potential.

14.2 EXPLORING THE SPACE OF VIEWS

When visualizing data, e.g., the distribution of a specific feature in a histogram, only a single view on the information space is revealed. A view, in this case, refers to the combination of specific data and visualization method, which is then presented to the user for interpretation. Similarity-based map visualizations as discussed in Part iii of this thesis are no exception. They only highlight one aspect of the information space, e.g., by grouping movies based on their overall similarity. However, high-dimensional information spaces can be observed from many different perspectives, resulting in many different views. When considering linked data, i.e., highly intertwined large information spaces, there are a vast number of potential views. Depending on various factors, e.g., the usage scenario, user preferences and more, one or multiple views are of particular interest. In many cases, but especially when considering exploration tasks, it is not clear what views on an information space are most relevant.

In most information system designs, the designer of the system selects a single view or a fixed number of views that are considered most relevant given the anticipated usage scenario of the system. For example, search results are often only visualized as a vertical list ordered by relevance. Users are often left without any means to actively adapt view parameters, e.g., the weights of a relevance scoring function. In the following, it is proposed that users should be able to control or influence which view is selected and presented, given that they are the judges of how helpful a certain view is given the current usage scenario. Since direct adaptation of view parameters often requires expert knowledge, an intuitive method is required for users to find and select one or more suitable views.

When enabling users to actively adapt or personalize a view, e.g., through direct manipulation of a similarity-based map projection as described in Chapter 12 or [116], users are already able to influence that particular view. However, personalized similarity-based maps are considered only a fraction of all available views on an information space. When imagining the set of all possible views as a space, it is argued that this space should be made accessible to users are much as possible. It is hypothesized that developing an intuitive method for users to explore, discover and select relevant views on information spaces will enable users to work more efficiently with information spaces. On the one hand, users are able to select more relevant views given their current information need or exploration scenario. On the other hand, custom views allow to define custom operators on the data, e.g., filtering operations that follow the definition of a view.

In order to draft such a method, the space of all available views needs to be characterized. Since there are a huge number of advanced visualization methods, unfortunately, only fundamental approaches can be considered in the following. Table 8 shows a selection of four types of views highlighting four types of relations between information objects. For example, an ordering between items can be visualized as a list of said items. However, objects can be ordered in many permutations. There are a small number of natural permutations, e.g., sorting people by age, or scientific papers by the number of citations. There are also personal permutations that only have a meaning to the current user, e.g., sorting movies by a user's personal rating. The goal is to allow users to explore the space of orderings, groupings, similarities, and connections in various views through direct manipulation of items. Table 8 also illustrates how views can be manipulated by users through interactions with individual items. In case of or-

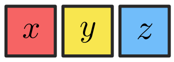
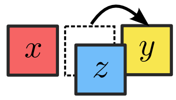
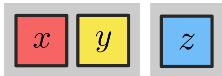
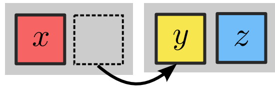
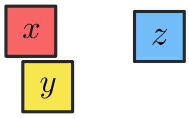
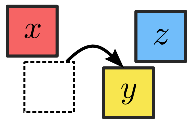
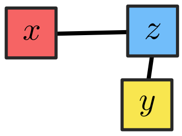
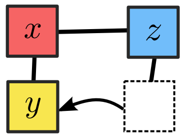
Type of Relation	**Data Model** (Data Level)	**Example View** (Visualization Level)	**Manipulation** (Interaction Level)
Ordering	$x < y < z$		
Grouping	$\{\{x, y\}, \{z\}\}$		
Similarity	$d(x, y)$, $d(x, z)$, $d(y, z)$		
Connection	$\{(x, z), (y, z)\}$		

Table 8: A selection of relation types and their representation on the data, visualization and interaction level. Views are adapted through user feedback provided as direct manipulations.

derings, users could move a particular item to another position in the list to provide feedback in terms of constraints. Based on these constraints, which describe item relations as perceived by the user, the best matching ordering could be selected for visualization. Given that many permutations will match a set of constraints, a bias needs to be defined, which describes which orderings are more likely to be relevant to users. This bias could build on the principle of Occam's razor and, e.g., prefer natural permutations before permutations that do not correlate with the data.

Previous research has focused on personalization in specific usage scenarios. In case of keyword search, the ranking of results can be adapted based on binary relevance feedback [120], relative order feedback [13], or implicit feedback [143, 130, 64]. Map-based projections can be adapted by allowing the user to move items [137, 116] or by changing weights of the underlying similarity metric [136]. Clusterings can be adapted to respect user-specific constraints [155, 5, 10]. Unfortunately, there is no approach yet that unifies personalization methods of multiple types of views in one exploration system using a common feedback strategy, e.g., direct manipulation of items.

A combination of personalization methods for various types of visualizations based on the common feedback strategy of direct manipulation is envisioned to result in a fluid dialogue between users and the information spaces. This dialogue is illustrated as a circle in Figure 47. Users start by interpreting the current visualization, which may not provide the desired information. Then, users provide feedback through direct manipulations of the visualization. Deduced constraints are used to build a model that determines the most likely

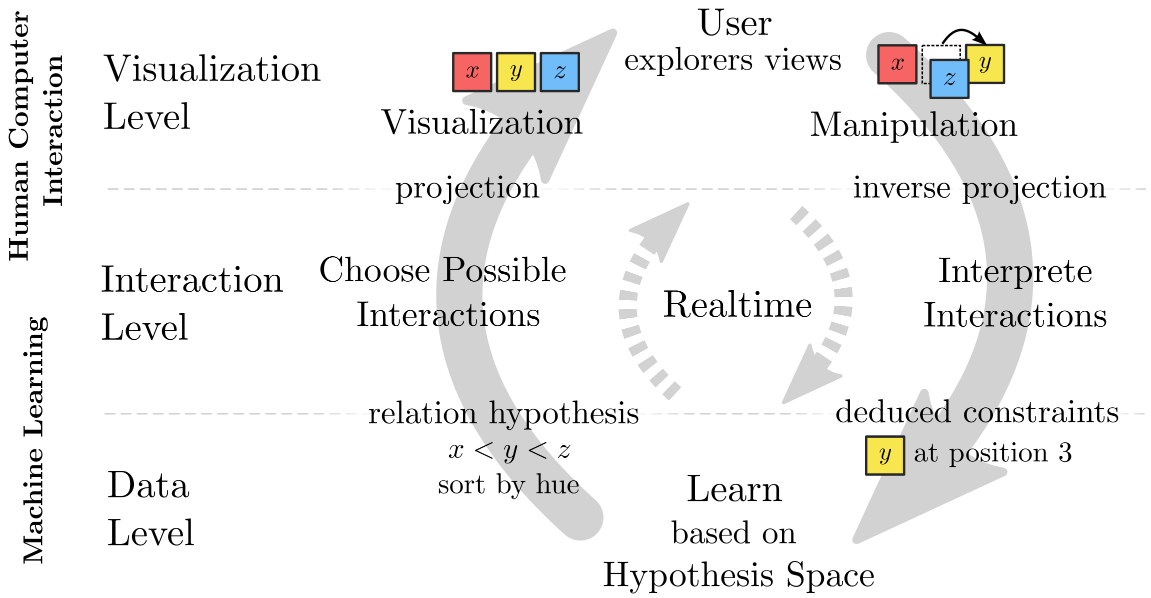


Figure 47: View Exploration Cycle: a user manipulates the visualization through direct manipulation. Interactions are interpreted as constraints, which allow to search the hypothesis space for relevant relation hypotheses, and determine the next view. Efficient algorithms allow for real-time interactions between user and the information system.

relation hypothesis, e.g., a specific permutation of items. Based on the relation hypothesis, visualizations are updated. Ideally, this process can be achieved in real time such that multiple quick iterations are possible.

Assuming that users can effectively navigate the space of available views, interesting new search and exploration scenarios are conceivable. For example, following the idea of the movie exploration tool proposed in Section 11.3, users could incrementally define a complex information need by going through the adaptation cycle illustrated in Figure 47 multiple times. First, a hypothetical user could order known movies based on personal liking by moving items in a list. Assuming that the deduced permutation reflects the user's preferences, the user could then add a filter operation such that only a certain fraction of preferred movies remain. After that, the user could arrange movies based on a custom notion of similarity, and use a lasso selection tool to filter additional movies. Finally, the remaining movies could be visualized as multiple lists grouped by genre.

In summary, the combination of multiple personalizable data representations, e.g., lists, categories, similarity-based maps, and more, is envisioned as an highly effective tool for exploring complex information spaces. In the future, more effort should be invested in studying methods that allow to adapt visualizations through direct manipulation, which is considered an effective and intuitive approach for personalization.

BIBLIOGRAPHY

Books

Publications related to this thesis can be found on page v.

[book:1] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., 1999 (cit. on pp. 14, 15).

[book:2] R. E. Bellman. *Adaptive Control Processes - A Guided Tour*. Princeton University Press, 1961, p. 255 (cit. on p. 21).

[book:3] C. M. Bishop et al. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995 (cit. on p. 44).

[book:4] I. Borg and P. J. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer Science & Business Media, 2005 (cit. on pp. 41, 42, 86).

[book:5] P. J. Clarkson, R. Coleman, S. Keates and C. Lebbon. *Inclusive design: Design for the Whole Population*. Springer Science & Business Media, 2013 (cit. on p. 19).

[book:6] A. Dix, A. J. Dix, J. Finlay, G. D. Abowd and R. Beale. *Human-Computer Interaction*. Pearson Education, 2003 (cit. on p. 19).

[book:7] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer Science & Business Media, 2012 (cit. on pp. 35, 69).

[book:8] I. Goodfellow, Y. Bengio and A. Courville. *Deep Learning*. MIT Press, 2016 (cit. on p. 46).

[book:9] I. T. Jolliffe. *Principal Component Analysis*. Springer Verlag, 1986 (cit. on p. 38).

[book:10] J. Pearsall, P. Hanks and A. Stevenson, eds. *Oxford Dictionary of English*. Oxford University Press, 2010 (cit. on p. 11).

[book:11] P. Salembier and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Interface*. Ed. by B. Manjunath. John Wiley & Sons, Inc., 2002 (cit. on p. 114).

Articles

Publications related to this thesis can be found on page v.

[1] C. C. Aggarwal, A. Hinneburg and D. A. Keim. 'On the Surprising Behavior of Distance Metrics in High Dimensional Space'. In: *Int. Conf. on Database Theory*. ICDT. 2001, pp. 420–434 (cit. on pp. 22, 24).

[2] J.-W. Ahn, P. Brusilovsky, D. He, J. Grady and Q. Li. 'Personalized Web Exploration with Task Models'. In: *Proc. of the 17th Int. Conf. on World Wide Web*. WWW. 2008, pp. 1–10 (cit. on p. 32).

[3] A. Aqle, D. Al-Thani and A. Jaoua. 'Can Search Result Summaries Enhance the Web Search Efficiency and Experiences of the Visually Impaired Users?' In: *Universal Access in the Information Society* (2020), pp. 1–22 (cit. on p. 69).

[4] W. Basalaj. *Proximity Visualisation of Abstract Data*. Tech. rep. University of Cambridge, Computer Laboratory, 2001 (cit. on pp. 96, 97).

[5] S. Basu, A. Banjeree, E. Mooney, A. Banerjee and R. J. Mooney. 'Active Semi-Supervision for Pairwise Constrained Clustering'. In: *Proc. of the SIAM Int. Conf. on Data Mining*. SDM. 2004, pp. 333–344 (cit. on p. 171).

[6] M. J. Bates. 'The Design of Browsing and Berrypicking Techniques for the Online Search Interface'. In: *Online Information Review* 13.5 (1989), pp. 407–424 (cit. on pp. 16, 100).

[7] J. L. Bentley. 'Multidimensional Binary Search Trees used for Associative Searching'. In: *Communications of the ACM* 18.9 (1975), pp. 509–517 (cit. on pp. 49, 102).

[8] K. Beyer, J. Goldstein, R. Ramakrishnan and U. Shaft. 'When Is "Nearest Neighbor" Meaningful?' In: *Int. Conf. on Database Theory*. ICDT. 1999, pp. 217–235 (cit. on p. 22).

[9] D. Bilal and J. Kirby. 'Differences and Similarities in Information Seeking: Children and Adults as Web Users'. In: *Information Processing & Management* 38.5 (2002), pp. 649–670 (cit. on pp. 57, 59).

[10] M. Bilenko, S. Basu and R. J. Mooney. 'Integrating Constraints and Metric Learning in Semi-Supervised Clustering'. In: *Proc. of the 21th Int. Conf. on Machine Learning*. ICML. 2004, pp. 11–18 (cit. on p. 171).

[11] F. L. Bookstein. 'Landmark Methods for Forms Without Landmarks: Morphometrics of Group Differences in Outline Shape'. In: *Medical Image Analysis* 1.3 (1997), pp. 225–243 (cit. on p. 47).

[12] C. Brewster, H. Alani, S. Dasmahapatra and Y. Wilks. 'Data Driven Ontology Evaluation'. In: *Proc. of the 4th Int. Conf. on Language Resources and Evaluation*. LREC. 2004 (cit. on p. 63).

[13] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton and G. Hullender. 'Learning to Rank using Gradient Descent'. In: *Proc. of the 22nd Int. Conf. on Machine Learning*. ICML. 2005, pp. 89–96 (cit. on p. 171).

[14] P. Butka and J. Pócs. 'Generalization of One-Sided Concept Lattices'. In: *Computing and Informatics* 32.2 (2013), pp. 355–370 (cit. on pp. 36, 70).

[15] J. E. Camargo, J. C. Caicedo and F. A. Gonzalez. 'A Kernel-Based Framework for Image Collection Exploration'. In: *Journal of Visual Languages & Computing* 24.1 (2013), pp. 53–67 (cit. on pp. 29, 86).

[16] R. G. Capra and G. Marchionini. 'The Relation Browser Tool for Faceted Exploratory Search'. In: *Proc. of the 8th ACM/IEEE-CS Joint Conf. on Digital Libraries*. JCDL. 2008, pp. 420–420 (cit. on p. 31).

[17] C. Carpineto, G. Romano and F. U. Bordoni. 'Exploiting the Potential of Concept Lattices for Information Retrieval with CREDO'. In: *Journal of Universal Computer Science* 10.8 (2004), pp. 985–1013 (cit. on p. 69).

[18] M. Cavallo and Ç. Demiralp. 'A Visual Interaction Framework for Dimensionality Reduction based Data Exploration'. In: *Proc. of the 2018 Conf. on Human Factors in Computing Systems*. CHI. 2018, pp. 1–13 (cit. on p. 45).

[19] J. C. Chang, N. Hahn, A. Perer and A. Kittur. 'SearchLens: Composing and Capturing Complex User Interests for Exploratory Search'. In: *Proc. of the 24th Int. Conf. on Intelligent User Interfaces*. IUI. 2019, pp. 498–509 (cit. on p. 32).

[20] A. Cockburn, A. Karlson and B. B. Bederson. 'A Review of Overview + Detail, Zooming, and Focus + Context Interfaces'. In: *ACM Computing Surveys* 41.1 (2009), pp. 1–31 (cit. on p. 19).

[21] M. Crampes and S. Ranwez. 'Ontology-Supported and Ontology-Driven Conceptual Navigation on the World Wide Web'. In: *Proc. of the 11th ACM on Hypertext and Hypermedia*. HYPERTEXT. 2000, pp. 191–199 (cit. on p. 61).

[22] J. V. Davis, B. Kulis, P. Jain, S. Sra and I. S. Dhillon. 'Information-Theoretic Metric Learning'. In: *Proc. of the 24th Int. Conf. on Machine Learning*. ICML. 2007, pp. 209–216 (cit. on p. 51).

[23] M. De Gemmis, P. Lops, G. Semeraro and C. Musto. 'An Investigation on the Serendipity Problem in Recommender Systems'. In: *Information Processing & Management* 51.5 (2015), pp. 695–717 (cit. on p. 18).

[24] C. De Maio, G. Fenza, V. Loia and S. Senatore. 'Hierarchical Web Resources Retrieval by Exploiting Fuzzy Formal Concept Analysis'. In: *Information Processing & Management* 48.3 (2012), pp. 399–418 (cit. on pp. 31, 35, 69).

[25] V. De Silva and J. B. Tenenbaum. 'Global Versus Local Methods in Nonlinear Dimensionality Reduction'. In: *Advances in Neural Information Processing Systems* (2003), pp. 721–728 (cit. on pp. 42, 86, 90).

[26] P. DeCamp, A. Frid-Jimenez, J. Guiness and D. Roy. 'Gist Icons: Seeing Meaning in Large Bodies of Literature'. In: *Proc. of the IEEE Symposium on Information Visualization*. InfoVis. 2005 (cit. on p. 78).

[27] T. Deselaers, T. Gass, P. Dreuw and H. Ney. 'Jointly Optimising Relevance and Diversity in Image Retrieval'. In: *Proc. of the ACM Int. Conf. on Image and Video Retrieval*. CIVR. 2009, p. 39 (cit. on p. 78).

[28] M. Dörk, C. Williamson and S. Carpendale. 'Navigating Tomorrow's Web: From Searching and Browsing to Visual Exploration'. In: *ACM Trans. on the Web*. TWEB 6.3 (2012), p. 13 (cit. on p. 25).

[29] S. Duarte Torres, D. Hiemstra and P. Serdyukov. 'Query Log Analysis in the Context of Information Retrieval for Children'. In: *Proc. of the 33rd Int. Conf. on Research and Development in Information Retrieval*. SIGIR. 2010, pp. 847–848 (cit. on p. 55).

[30] S. Duarte Torres and I. Weber. 'What and How Children Search on the Web'. In: *Proc. of the 20th Int. Conf. on Information and Knowledge Management*. CIKM. 2011, pp. 393–402 (cit. on p. 56).

[31] S. Duarte Torres, I. Weber and D. Hiemstra. 'Analysis of Search and Browsing Behavior of Young Users on the Web'. In: *ACM Trans. on the Web*. TWEB 8.2 (2014), pp. 1–54 (cit. on pp. 56, 57).

[32] T. Falke and I. Gurevych. 'GraphDocExplore: A Framework for the Experimental Comparison of Graph-Based Document Exploration Techniques'. In: *Proc. of the 2017 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations*. 2017, pp. 19–24 (cit. on p. 26).

[33] L. Fei-Fei, R. Fergus and P. Perona. 'One-Shot Learning of Object Categories'. In: *IEEE Trans. on Pattern Analysis & Machine Intelligence* 28.4 (2006), pp. 594–611 (cit. on p. 114).

[34] B. Fritzke. 'Growing Grid - A Self-Organizing Network with Constant Neighborhood Range and Adaptation Strength'. In: *Neural Processing Letters* 2.5 (1995), pp. 9–13 (cit. on p. 46).

[35] V. Ganti, R. Ramakrishnan, J. Gehrke, A. Powell and J. French. 'Clustering Large Datasets in Arbitrary Metric Spaces'. In: *Proc. of the 15th IEEE Int. Conf. on Data Engineering*. ICDE. 1999, pp. 502–511 (cit. on p. 104).

[36] A. W. Gatian. 'Is User Satisfaction a Valid Measure of System Effectiveness?' In: *Information & Management* 26.3 (1994), pp. 119–131 (cit. on p. 20).

[37] E. J. Gibson. 'Exploratory Behavior in the Development of Perceiving, Acting, and the Acquiring of Knowledge'. In: *Annual Review of Psychology* 39.1 (1988), pp. 1–42 (cit. on p. 11).

[38] A. Gionis, P. Indyk, R. Motwani et al. 'Similarity Search in High Dimensions via Hashing'. In: *Proc. of the 25th Int. Conf. on Very Large Data Bases*. Vol. 99. VLDB. 1999, pp. 518–529 (cit. on p. 134).

[39] J.-E. Golenia, M. Wenzel and B. Blankertz. 'Live Demonstrator of EEG and Eye-Tracking Input for Disambiguation of Image Search Results'. In: *Int. Workshop on Symbiotic Interaction*. Springer. 2015, pp. 81–86 (cit. on p. 144).

[40] G. Golovchinsky and M. H. Chignell. 'The Newspaper as an Information Exploration Metaphor'. In: *Information Processing & Management* 33.5 (1997), pp. 663–683 (cit. on pp. 25, 31).

[41] G. Golovchinsky, A. Diriye and T. Dunnigan. 'The Future is in the Past: Designing for Exploratory Search'. In: *Proc. of the 4th Information Interaction in Context Symposium*. IIIX. 2012, pp. 52–61 (cit. on p. 33).

[42] C. Görg, Z. Liu, J. Kihm, J. Choo, H. Park and J. Stasko. 'Combining Computational Analyses and Interactive Visualization for Document Exploration and Sensemaking in Jigsaw'. In: *IEEE Trans. on Visualization and Computer Graphics* 19.10 (2012), pp. 1646–1663 (cit. on p. 25).

[43] J. Gower. 'Generalized Procrustes Analysis'. In: *Psychometrika* 40.1 (1975), pp. 33–51 (cit. on pp. 47, 91, 102).

[44] J. C. Gower. 'Some Distance Properties of Latent Root and Vector Methods used in Multivariate Analysis'. In: *Biometrika* 53.3-4 (1966), pp. 325–338 (cit. on p. 41).

[45] M. Grachten, M. Schedl, T. Pohle and G. Widmer. 'The ISMIR Cloud: A Decade of ISMIR Conferences at Your Fingertips'. In: *Proc. of the 10th Int. Conf. on Music Information Retrieval*. ISMIR. 2009, pp. 63–68 (cit. on p. 126).

[46] M. R. Grossman, G. V. Cormack and A. Roegiest. 'TREC 2016 Total Recall Track Overview'. In: *Text Retrieval Configerence*. TREC. 2016 (cit. on p. 15).

[47] S. Günter, N. Schraudolph, S. Vishwanathan et al. 'Fast Iterative Kernel Principal Component Analysis'. In: *Journal of Machine Learning Research* 8 (2007), pp. 1893–1918 (cit. on p. 85).

[48] A. Guttman. 'R-trees: A Dynamic Index Structure for Spatial Searching'. In: *Proc. of the 1984 ACM SIGMOD Int. Conf. on Management of Data*. 1984, pp. 47–57 (cit. on p. 49).

[49] J. Gwizdka and Y. Zhang. 'Differences in Eye-Tracking Measures between Visits and Revisits to Relevant and Irrelevant Web Pages'. In: *Proc. of the 38th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*. SIGIR. 2015, pp. 811–814 (cit. on p. 144).

[50] S. Hadlak, H. Schumann and H.-J. Schulz. 'A Survey of Multi-faceted Graph Visualization'. In: *Eurographics Conf. on Visualization*. EuroVis. 2015, pp. 1–20 (cit. on p. 25).

[51] C. Harte, M. Sandler, S. Abdallah and E. Gómez. 'Symbolic Representation of Musical Chords: A Proposed Syntax for Text Annotations'. In: *Proc. of the 6th Int. Society for Music Information Retrieval*. ISMIR. 2005, pp. 66–71 (cit. on p. 92).

[52] A. Hassan Awadallah, R. W. White, P. Pantel, S. T. Dumais and Y.-M. Wang. 'Supporting Complex Search Tasks'. In: *Proc. of the 23rd Int. Conf. on Information and Knowledge Management*. CIKM. 2014, pp. 829–838 (cit. on p. 31).

[53] S. Haun, T. Gossen, A. Nürnberger, T. Kötter, K. Thiel and M. R. Berthold. 'On the Integration of Graph Exploration and Data Analysis: The Creative Exploration Toolkit'. In: *Bisociative Knowledge Discovery*. 2012, pp. 301–312 (cit. on p. 26).

[54] S. Haun, A. Nürnberger, T. Kötter, K. Thiel and M. R. Berthold. 'CET: A Tool for Creative Exploration of Graphs'. In: *Joint European Conf. on Machine Learning and Knowledge Discovery in Databases*. ECML PKDD. 2010, pp. 587–590 (cit. on p. 71).

[55] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen and K.-P. Yee. 'Finding the Flow in Web Site Search'. In: *Communications of the ACM* 45.9 (2002), pp. 42–49 (cit. on p. 30).

[56] G. Hinton and S. T. Roweis. 'Stochastic Neighbor Embedding'. In: *Proc. of the 15th Int. Conf. on Neural Information Processing Systems*. Vol. 15. NIPS. 2002, pp. 833–840 (cit. on pp. 42, 90).

[57] G. E. Hinton and R. R. Salakhutdinov. 'Reducing the Dimensionality of Data with Neural Networks'. In: *Science* 313.5786 (2006), pp. 504–507 (cit. on p. 44).

[58] E. Hoque, O. Hoeber and M. Gong. 'CIDER: Concept-Based Image Diversification, Exploration, and Retrieval'. In: *Information Processing & Management* 49.5 (2013), pp. 1122–1138 (cit. on p. 31).

[59] M. Houle, H.-P. Kriegel, P. Kröger, E. Schubert and A. Zimek. 'Can Shared-Neighbor Distances Defeat the Curse of Dimensionality?' In: *Int. Conf. on Scientific and Statistical Database Management*. Vol. 6187. SSDBM. 2010. Chap. 34, pp. 482–500 (cit. on p. 24).

[60] P. Indyk and R. Motwani. 'Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality'. In: *Proc. of the 30th Annual ACM Symposium on Theory of Computing*. 1998, pp. 604–613 (cit. on pp. 50, 102).

[61] P. Isenberg, U. Hinrichs, M. Hancock and S. Carpendale. 'Digital Tables for Collaborative Information Exploration'. In: *Tabletops - Horizontal Interactive Displays*. 2010, pp. 387–405 (cit. on p. 25).

[62] S. Jabbari and K. Stoffel. 'FCA-Based Ontology Learning From Unstructured Textual Data'. In: *Int. Conf. on Mining Intelligence and Knowledge Exploration*. Springer. 2018, pp. 1–10 (cit. on p. 69).

[63] T. Jankun-Kelly and K.-L. Ma. 'MoireGraphs: Radial Focus+Context Visualization and Interaction for Graphs with Visual Nodes'. In: *IEEE Symposium on Information Visualization*. INFOVIS. 2003, pp. 59–66 (cit. on p. 26).

[64] R. Jin, H. Valizadegan and H. Li. 'Ranking Refinement and its Application to Information Retrieval'. In: *Proc. of the 17th Int. Conf. on World Wide Web*. WWW. 2008, pp. 397–406 (cit. on p. 171).

[65] C. F. Julià and S. Jordà. 'SongExplorer: A Tabletop Application for Exploring Large Collections of Songs'. In: *Proc. of the 10th Int. Society for Music Information Retrieval*. ISMIR. 2009, pp. 675–680 (cit. on pp. 28, 90).

[66] A. Kanazawa, M. J. Black, D. W. Jacobs and J. Malik. 'End-to-End Recovery of Human Shape and Pose'. In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*. CVPR. 2018, pp. 7122–7131 (cit. on p. 47).

[67] S. Kaski, J. Nikkilä, M. Oja, J. Venna, P. Törönen and E. Castrén. 'Trustworthiness and Metrics in Visualizing Similarity of Gene Expression'. In: *BMC Bioinformatics* 4.1 (2003), pp. 1–13 (cit. on pp. 39, 90).

[68] D. P. Kingma and M. Welling. 'Auto-Encoding Variational Bayes'. In: *Proc. of the 2nd Int. Conf. on Learning Representations*. ICLR. 2014 (cit. on p. 45).

[69] K. Klouche, T. Ruotsalo, D. Cabral, S. Andolina, A. Bellucci and G. Jacucci. 'Designing for Exploratory Search on Touch Devices'. In: *Proc. of the 33rd Conf. on Human Factors in Computing Systems*. CHI. 2015, pp. 4189–4198 (cit. on p. 33).

[70] P. Knees, M. Schedl, T. Pohle and G. Widmer. 'Exploring Music Collections in Virtual Landscapes'. In: *IEEE Multimedia* 14.3 (2007), pp. 46–54 (cit. on pp. 27, 30, 46, 90).

[71] T. Kohonen. 'Self-Organized Formation of Topologically Correct Feature Maps'. In: *Biological Cybernetics* 43.1 (1982), pp. 59–69 (cit. on pp. 27, 46).

[72] T. Kohonen. 'Exploration of Very Large Databases by Self-Organizing Maps'. In: *Proc. of the Int. Conf. on Neural Networks*. Vol. 1. ICNN. 1997, PL1–PL6 (cit. on pp. 27, 29).

[73] M. Kotzyba, T. Gossen, J. Schwerdt and A. Nürnberger. 'Exploration or Fact-Finding: Inferring User's Search Activity Just in Time'. In: *Proc. of the 2017 Conf. on Human Information Interaction and Retrieval*. CHIIR. 2017, pp. 87–96 (cit. on p. 18).

[74] A. Krizhevsky and G. E. Hinton. 'Using Very Deep Autoencoders for Content-Based Image Retrieval'. In: *Proc. of the 19th European Symposium on Artificial Neural Networks*. Vol. 1. ESANN. 2011, p. 2 (cit. on p. 45).

[75] J. Kruskal. 'Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis'. In: *Psychometrika* 29 (1964), pp. 1–27 (cit. on pp. 38, 90, 102).

[76] B. Kulis et al. 'Metric Learning: A Survey'. In: *Foundations and Trends in Machine Learning* 5.4 (2012), pp. 287–364 (cit. on p. 51).

[77] R. Langner, T. Horak and R. Dachselt. 'VisTiles: Coordinating and Combining Co-located Mobile Devices for Visual Data Exploration'. In: *IEEE Trans. on Visualization and Computer Graphics* 24.1 (2017), pp. 626–636 (cit. on p. 25).

[78] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer et al. 'DBpedia – A Large-Scale, Multilingual Knowledge Base Extracted from Wikipedia'. In: *Semantic Web* 6.2 (2015), pp. 167–195 (cit. on p. 65).

[79] Q. Li, M. Tian, J. Liu and J. Sun. 'An Implicit Relevance Feedback Method for CBIR with Real-Time Eye Tracking'. In: *Multimedia Tools and Applications* 75.5 (2016), pp. 2595–2611 (cit. on p. 144).

[80] H. Liu, X. Xie, X. Tang, Z.-W. Li and W.-Y. Ma. 'Effective Browsing of Web Image Search Results'. In: *Proc. of the 6th ACM SIGMM Int. Workshop on Multimedia Information Retrieval*. MIR. 2004, pp. 84–90 (cit. on p. 96).

[81] T. Liu, A. W. Moore, A. Gray and C. Cardie. 'New Algorithms for Efficient High-Dimensional Nonparametric Classification'. In: *Journal of Machine Learning Research* 7.6 (2006) (cit. on p. 49).

[82] Y. Liu, S. Barlowe, Y. Feng, J. Yang and M. Jiang. 'Evaluating Exploratory Visualization Systems: A User Study on How Clustering-Based Visualization Systems Support Information Seeking from Large Document Collections'. In: *Information Visualization* 12.1 (2013), pp. 25–43 (cit. on p. 20).

[83] S. Lohmann, P. Heim, T. Stegemann and J. Ziegler. 'The RelFinder User Interface: Interactive Exploration of Relationships between Objects of Interest'. In: *Proc. of the 15th Int. Conf. on Intelligent User Interfaces*. IUI. 2010, pp. 421–422 (cit. on p. 26).

[84] J. Lokoč, G. Kovalčík, T. Souček, J. Moravec and P. Čech. 'VIRET: A Video Retrieval Tool for Interactive Known-Item Search'. In: *Proc. of the 2019 on Int. Conf. on Multimedia Retrieval*. ICMR. 2019, pp. 177–181 (cit. on p. 133).

[85] P. Majaranta and A. Bulling. 'Eye Tracking and Eye-Based Human Computer Interaction'. In: *Advances in Physiological Computing*. 2014, pp. 39–65 (cit. on p. 144).

[86] M. Mandel and D. Ellis. 'Song-Level Features and Support Vector Machines for Music Classification'. In: *Proc. of the 6th Int. Society for Music Information Retrieval*. ISMIR. 2005, pp. 594–599 (cit. on pp. 27, 92).

[87] G. Marchionini. 'Exploratory Search: From Finding to Understanding'. In: *Communications of the ACM* 49.4 (2006), pp. 41–46 (cit. on p. 17).

[88] B. McFee, L. Barrington and G. Lanckriet. 'Learning Content Similarity for Music Recommendation'. In: *IEEE Trans. on Audio, Speech, and Language Processing* 20.8 (2012), pp. 2207–2218 (cit. on p. 51).

[89] B. McFee and G. R. Lanckriet. 'Metric Learning To Rank'. In: *Proc. of the 27th Int. Conf. on Machine Learning*. ICML. 2010, pp. 775–782 (cit. on p. 51).

[90] S. M. McNee, J. Riedl and J. A. Konstan. 'Being Accurate is not Enough: How Accuracy Metrics have Hurt Recommender systems'. In: *Extended Abstracts on Human Factors in Computing Systems*. CHI. 2006, pp. 1097–1101 (cit. on p. 18).

[91] E. Minack, W. Siberski and W. Nejdl. 'Incremental Diversification for Very Large Sets: A Streaming-Based Approach'. In: *Proc. of the 34th Int. Conf. on Research and Development in Information Retrieval*. SIGIR. 2011, pp. 585–594 (cit. on pp. 104, 137, 154).

[92] A. Mitschick, F. Nieschalk, M. Voigt and R. Dachselt. 'Icicle-Query: A Web Search Interface for Fluid Semantic Query Construction'. In: *Proc. of the 3rd Int. Workshop on Visualization and Interaction for Ontologies and Linked Data at the Int. Semantic Web Conf.* VOILA@ISWC. 2017, pp. 99–110 (cit. on p. 61).

[93] O. R. Musin. 'The Kissing Number in Four Dimensions'. In: *Annals of Mathematics* 168 (2008), pp. 1–32 (cit. on p. 110).

[94] M. Nefkens and W. Hürst. 'The MovieWall: A New Interface for Browsing Large Video Collections'. In: *Proc. of the 27th Int. Conf. on Multimedia Modeling*. MMM. 2021, pp. 170–182 (cit. on pp. 133, 137, 141).

[95] N. Nikitina, S. Rudolph and B. Glimm. 'Interactive Ontology Revision'. In: *Journal of Web Semantics* 12-13 (2012), pp. 118–130 (cit. on p. 67).

[96] M. Nitsche and A. Nürnberger. 'QUEST: Querying Complex Information by Direct Manipulation'. In: *Human Interface and the Management of Information. Information and Interaction Design*. Vol. 8016. LNCS. 2013, pp. 240–249 (cit. on p. 32).

[97] M. Nitsche and A. Nürnberger. 'Trailblazing Information: An Exploratory Search User Interface'. In: *Int. Conf. on Human Interface and the Management of Information*. HIMI. 2013, pp. 230–239 (cit. on p. 32).

[98] A. Odlyzko and N. Sloane. 'New Bounds on the Number of Unit Spheres that can Touch a Unit Sphere in n Dimensions'. In: *Journal of Combinatorial Theory, Series A* 26.2 (1979), pp. 210–214 (cit. on p. 110).

[99] K. A. Olsen, R. R. Korfhage, K. M. Sochats, M. B. Spring and J. G. Williams. 'Visualization of a Document Collection: The VIBE system'. In: *Information Processing & Management* 29.1 (1993), pp. 69–81 (cit. on pp. 31, 32).

[100] H. Osipyan, M. Kruliš and S. Marchand-Maillet. 'A survey of cuda-based multidimensional scaling on gpu architecture'. In: *Imperial College Computing Student Workshop*. ICCSW. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2015 (cit. on p. 42).

[101] H. Osipyan, A. Morton and S. Marchand-Maillet. 'Fast interactive information retrieval with sampling-based MDS on GPU architectures'. In: *Information Retrieval Facility Conference*. Springer. 2014, pp. 96–107 (cit. on p. 42).

[102] J. G. S. Paiva, W. R. Schwartz, H. Pedrini and R. Minghim. 'Semi-Supervised Dimensionality Reduction based on Partial Least Squares for Visual Analysis of High Dimensional Data'. In: *Computer Graphics Forum*. Vol. 31. 2012, pp. 1345–1354 (cit. on p. 152).

[103] E. Pampalk, A. Rauber and D. Merkl. 'Content-based Organization and Visualization of Music Archives'. In: *Proc. of the 10th ACM Int. Conf. on Multimedia*. MULTIMEDIA. 2002, pp. 570–579 (cit. on p. 92).

[104] T. Partala and V. Surakka. 'Pupil Size Variation as an Indication of Affective Processing'. In: *Int. Journal of Human-Computer Studies* 59.1-2 (2003), pp. 185–198 (cit. on p. 144).

[105] C. L. Paul, J. Chang, A. Endert, N. Cramer, D. Gillen, S. Hampton, R. Burtner, R. Perko and K. A. Cook. 'TexTonic: Interactive Visualization for Exploration and Discovery of Very Large Text Collections'. In: *Information Visualization* 18.3 (2019), pp. 339–356 (cit. on pp. 29, 126).

[106] D. Paurat and T. Gärtner. 'InVis: A Tool for Interactive Visual Data Analysis'. In: *Machine Learning and Knowledge Discovery in Databases*. Vol. 8190. LNCS. 2013, pp. 672–676 (cit. on pp. 151, 152).

[107] J. Poelmans, P. Elzinga, S. Viaene and G. Dedene. 'Formal Concept Analysis in Knowledge Discovery: A Survey'. In: *Int. Conf. on Conceptual Structures*. Springer. 2010, pp. 139–153 (cit. on p. 35).

[108] U. Priss. 'Lattice-Based Information Retrieval'. In: *Knowledge Organization* 27.3 (2000), pp. 132–142 (cit. on p. 69).

[109] G.-J. Qi, J. Tang, Z.-J. Zha, T.-S. Chua and H.-J. Zhang. 'An Efficient Sparse Metric Learning in High-Dimensional Space via l1-Penalized Log-Determinant Regularization'. In: *Proc. of the 26th Int. Conf. on Machine Learning*. ICML. 2009, pp. 841–848 (cit. on p. 51).

[110] J. Raad and C. Cruz. 'A Survey on Ontology Evaluation Methods'. In: *Proc. of the Int. Joint Conf. on Knowledge Discovery, Knowledge Engineering and Knowledge Management*. IC3K. 2015, pp. 179–186 (cit. on p. 63).

[111] M. Radovanović, A. Nanopoulos and M. Ivanović. 'Nearest Neighbors in High-Dimensional Data: The Emergence and Influence of Hubs'. In: *Proc. of the 26th Int. Conf. on Machine Learning*. ICML. 2009, pp. 865–872 (cit. on p. 23).

[112] M. Radovanović, A. Nanopoulos and M. Ivanović. 'Hubs in Space: Popular Nearest Neighbors in High-Dimensional Data'. In: *Journal of Machine Learning Research* (2010), pp. 2487–2531 (cit. on pp. 23, 24, 106, 109).

[113] M. A. Ranzato, Y.-L. Boureau and Y. LeCun. 'Sparse Feature Learning for Deep Belief Networks'. In: *Proc. of the 20th Int. Conf. on Neural Information Processing Systems*. NIPS. 2007, pp. 1185–1192 (cit. on p. 45).

[114] P. E. Rauber, A. X. Falcão, A. C. Telea et al. 'Visualizing Time-Dependent Data Using Dynamic t-SNE.' In: *Proc. of the Eurographics / IEEE VGTC Conf. on Visualization*. EuroVis. 2016, pp. 73–77 (cit. on p. 90).

[115] K. Ravi Kanth, D. Agrawal and A. Singh. 'Dimensionality Reduction for Similarity Searching in Dynamic Databases'. In: *ACM SIGMOD Record* 27.2 (1998), pp. 166–176 (cit. on p. 89).

[116] B. Reinert, T. Ritschel and H.-P. Seidel. 'Interactive By-example Design of Artistic Packing Layouts'. In: *ACM Trans. on Graphics* 32.6 (2013), 218:1–218:7 (cit. on pp. 29, 96, 151, 152, 170, 171).

[117] R. S. Rele and A. T. Duchowski. 'Using Eye Tracking to Evaluate Alternative Search Results Interfaces'. In: *Proc. of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 49. 15. 2005, pp. 1459–1463 (cit. on p. 75).

[118] T. Repke and R. Krestel. 'Visualising Large Document Collections by Jointly Modeling Text and Network Structure'. In: *Proc. of the ACM/IEEE Joint Conf. on Digital Libraries*. JCDL. 2020, pp. 279–288 (cit. on pp. 30, 43, 87, 100, 126).

[119] T. Repke and R. Krestel. 'Robust Visualisation of Dynamic Text Collections: Measuring and Comparing Dimensionality Reduction Algorithms'. In: *Proc. of the SIGIR Conf. on Human Information Interaction and Retrieval*. CHIIR. 2021, pp. 1–4 (cit. on p. 90).

[120] J. Rocchio. 'Relevance Feedback in Information Retrieval'. In: *The SMART Retrieval System: Experiments in Automatic Document Processing*. 1971, pp. 313–323 (cit. on p. 171).

[121] F. J. Rohlf and D. Slice. 'Extensions of the Procrustes Method for the Optimal Superimposition of Landmarks'. In: *Systematic Biology* 39.1 (1990), pp. 40–59 (cit. on p. 47).

[122] S. Roweis and L. Saul. 'Nonlinear Dimensionality Reduction by Locally Linear Embedding'. In: *Science* 290.5500 (2000), pp. 2323–2326 (cit. on p. 38).

[123] Y. Rubner, L. Guibas and C. Tomasi. 'The Earth Mover's Distance, Multi-Dimensional Scaling, and Color-Based Image Retrieval'. In: *Proc. of the ARPA Image Understanding Workshop* (1997), pp. 661–668 (cit. on pp. 100, 101).

[124] T. Ruotsalo, J. Peltonen, M. J. Eugster, D. Głowacka, P. Floréen, P. Myllymäki, G. Jacucci and S. Kaski. 'Interactive Intent Modeling for Exploratory Search'. In: *ACM Trans. on Information Systems* 36.4 (2018), pp. 1–46 (cit. on p. 32).

[125] R. Salakhutdinov and G. Hinton. 'Semantic Hashing'. In: *Int. Journal of Approximate Reasoning* 50.7 (2009), pp. 969–978 (cit. on pp. 45, 50).

[126] M. Schedl, M. Mayr and P. Knees. 'Music Tower Blocks: Multi-Faceted Exploration Interface for Web-Scale Music Access'. In: *Proc. of the 2020 Int. Conf. on Multimedia Retrieval*. ICMR. 2020, pp. 388–392 (cit. on pp. 29, 43, 87).

[127] D. Schnitzer, A. Flexer, M. Schedl and G. Widmer. 'Using Mutual Proximity to Improve Content-Based Audio Similarity'. In: *3rd Int. Conf. on Music Information Retrieval*. ISMIR. 2011, pp. 79–84 (cit. on pp. 23, 24).

[128] J. Schwerdt, M. Kotzyba and A. Nürnberger. 'Inferring User's Search Activity using Interaction Logs and Gaze Data'. In: *Proc. of the 2017 Int. Conf. on Companion Technology*. ICCT. 2017, pp. 1–6 (cit. on p. 144).

[129] J. Schwerdt, M. Kotzyba and A. Nürnberger. 'Fact-Finding or Exploration: Identifying Latent Behavior Clusters in User's Search Activities'. In: *Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics*. SMC. 2019, pp. 1465–1471 (cit. on pp. 56, 59).

[130] A. Sieg, B. Mobasher and R. Burke. 'Web Search Personalization with Ontological User Profiles'. In: *Proc. of the 16th ACM Conf. on Information and Knowledge Management*. CIKM. 2007, pp. 525–534 (cit. on p. 171).

[131] J. Singh, W. Nejdl and A. Anand. 'Expedition: A Time-Aware Exploratory Search System Designed for Scholars'. In: *Proc. of the 39th Int. Conf. on Research and Development in Information Retrieval*. SIGIR. 2016, pp. 1105–1108 (cit. on p. 31).

[132] P. Sorg and P. Cimiano. 'Exploiting Wikipedia for Cross-Lingual and Multilingual Information Retrieval'. In: *Data & Knowledge Engineering* 74 (2012), pp. 26–45 (cit. on p. 61).

[133] M. Spindler, S. Stellmach and R. Dachselt. 'PaperLens: Advanced Magic Lens Interaction Above the Tabletop'. In: *Proc. of the ACM Int. Conf. on Interactive Tabletops and Surfaces*. ITS. 2009, pp. 69–76 (cit. on p. 168).

[134] A. Spink and B. J. Jansen. 'A Study of Web Search Trends'. In: *Webology* 1.2 (2004), p. 4 (cit. on pp. 1, 57, 59).

[135] S. Stellmach, M. Jüttner, C. Nywelt, J. Schneider and R. Dachselt. 'Investigating Freehand Pan and Zoom'. In: *Mensch & Computer 2012: Interaktiv Informiert – Allgegenwärtig und Allumfassend!?* (2012), pp. 303–312 (cit. on p. 169).

[136] S. Stober, C. Hentschel and A. Nürnberger. 'Multi-Facet Exploration of Image Collections with an Adaptive Multi-Focus Zoomable Interface'. In: *Proc. of 2010 IEEE World Congress on Computational Intelligence*. WCCI. 2010, pp. 2780–2787 (cit. on pp. 28, 171).

[137] S. Stober and A. Nürnberger. 'Towards User-Adaptive Structuring and Organization of Music Collections'. In: *Int. Workshop on Adaptive Multimedia Retrieval*. AMR. 2008, pp. 53–65 (cit. on pp. 46, 51, 88–90, 171).

[138] S. Stober and A. Nürnberger. 'Analyzing the Impact of Data Vectorization on Distance Relations'. In: *IEEE Int. Conf. on Multimedia and Expo*. 2011, pp. 1–6 (cit. on p. 91).

[139] S. Stober and A. Nürnberger. 'MusicGalaxy: A Multi-focus Zoomable Interface for Multi-facet Exploration of Music Collections'. In: *Exploring Music Contents*. Vol. 6684. LNCS. 2011, pp. 273–302 (cit. on pp. 24, 28, 90, 100).

[140] H. Stuckenschmidt, A. De Waard, R. Bhogal, C. Fluit, A. Kampman, J. van Buel, E. van Mulligen, J. Broekstra, I. Crowlesmith, F. van Harmelen et al. 'A Topic-Based Browser for Large Online Resources'. In: *Int. Conf. on Knowledge Engineering and Knowledge Management*. EKAW. 2004, pp. 433–448 (cit. on p. 31).

[141] O. Suominen, K. Viljanen and E. Hyvänen. 'User-Centric Faceted Search for Semantic Portals'. In: *European Semantic Web Conference*. ESWC. 2007, pp. 356–370 (cit. on pp. 31, 61).

[142] J. Tan, X. Wan and J. Xiao. 'Abstractive Document Summarization with a Graph-Based Attentional Neural Model'. In: *Proc. of the 55th Annual Meeting of the Association for Computational Linguistics*. 2017, pp. 1171–1181 (cit. on p. 79).

[143] J. Teevan, S. T. Dumais and E. Horvitz. 'Personalizing Search via Automated Analysis of Interests and Activities'. In: *Proc. of the 28th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*. SIGIR. 2005, pp. 449–456 (cit. on p. 171).

[144] J. B. Tenenbaum, V. De Silva and J. C. Langford. 'A Global Geometric Framework for Nonlinear Dimensionality Reduction'. In: *Science* 290.5500 (2000), pp. 2319–2323 (cit. on p. 38).

[145] A. Tharwat. 'Principal Component Analysis - A Tutorial'. In: *Int. Journal of Applied Pattern Recognition* 3.3 (2016), pp. 197–240 (cit. on p. 39).

[146] W. S. Torgerson. 'Multidimensional Scaling: I. Theory and Method'. In: *Psychometrika* 17.4 (1952), pp. 401–419 (cit. on p. 41).

[147] M. Torrens, P. Hertzog and J. L. Arcos. 'Visualizing and Exploring Personal Music Libraries.' In: *Proc. of the 5th Int. Conf. on Music Information Retrieval*. ISMIR. 2004 (cit. on p. 28).

[148] D. Vallet, M. Fernández and P. Castells. 'An Ontology-Based Information Retrieval Model'. In: *The Semantic Web: Research and Applications*. 2005, pp. 455–470 (cit. on p. 61).

[149] L. Van Der Maaten. 'Accelerating t-SNE using Tree-Based Algorithms'. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 3221–3245 (cit. on pp. 29, 43, 87, 100).

[150] L. Van der Maaten and G. Hinton. 'Visualizing Data using t-SNE.' In: *Journal of Machine Learning Research* 9.11 (2008) (cit. on pp. 42, 43).

[151] L. Van Der Maaten, E. Postma and J. Van den Herik. 'Dimensionality Reduction: A Comparative'. In: *Journal of Machine Learning Research* 10.66-71 (2009), p. 13 (cit. on p. 38).

[152] J. Venna and S. Kaski. 'Nonlinear Dimensionality Reduction as Information Retrieval'. In: *Proc. of the 11th Int. Conf. on Artificial Intelligence and Statistics*. AISTATS. 2007, pp. 572–579 (cit. on p. 44).

[153] J. Venna, J. Peltonen, K. Nybo, H. Aidos and S. Kaski. 'Information Retrieval Perspective to Nonlinear Dimensionality Reduction for Data Visualization'. In: *Journal of Machine Learning Research* 11.2 (2010) (cit. on pp. 44, 90).

[154] E. F. Vernier, R. Garcia, I. d. Silva, J. L. D. Comba and A. C. Telea. 'Quantitative Evaluation of Time-Dependent Multidimensional Projection Techniques'. In: *Computer Graphics Forum*. Vol. 39. 3. 2020, pp. 241–252 (cit. on p. 90).

[155] K. Wagstaff, C. Cardie, S. Rogers and S. Schrödl. 'Constrained K-means Clustering with Background Knowledge'. In: *Proc. of the 18th Int. Conf. on Machine Learning*. ICML. 2001, pp. 577–584 (cit. on p. 171).

[156] J. A. Waterworth. 'A Pattern of Islands: Exploring Public Information Space in a Private Vehicle'. In: *Int. Conf. on Multimedia, Hypermedia, and Virtual Reality*. MHVR. 1996, pp. 265–278 (cit. on pp. 25, 46).

[157] J. A. Waterworth and M. H. Chignell. 'A Model of Information Exploration'. In: *Hypermedia* 3.1 (1991), pp. 35–58 (cit. on p. 13).

[158] R. W. White, G. Marchionini and G. Muresan. 'Evaluating Exploratory Search Systems'. In: *Information Processing and Management* 44.2 (2008), p. 433 (cit. on pp. 20, 119, 120).

[159] M. L. Wilson. 'Search User Interface Design'. In: *Synthesis Lectures on Information Concepts, Retrieval, and Services* 3.3 (2011), pp. 1–143 (cit. on p. 30).

[160] K. Wittenburg, T. Lanning, M. Heinrichs and M. Stanton. 'Parallel Bargrams for Consumer-Based Information Exploration and Choice'. In: *Proc. of the 14th ACM Symposium on User Interface Software and Technology*. UIST. 2001, pp. 51–60 (cit. on p. 25).

[161] E. P. Xing, A. Y. Ng, M. I. Jordan and S. Russell. 'Distance Metric Learning with Application to Clustering with Side-Information'. In: *Proc. of the 15th Int. Conf. on Neural Information Processing*. Vol. 15. NIPS. 2002, pp. 505–512 (cit. on p. 51).

[162] T. Yang, J. Liu, L. McMillan and W. Wang. 'A fast approximation to multidimensional scaling'. In: *IEEE workshop on computation intensive methods for computer vision*. 2006 (cit. on p. 42).

[163] S. M. Yimam, H. Ulrich, T. von Landesberger, M. Rosenbach, M. Regneri, A. Panchenko, F. Lehmann, U. Fahrer, C. Biemann and K. Ballweg. 'new/s/leak – Information Extraction and Visualization for Investigative Data Journalists'. In: *Proc. of ACL-2016 System Demonstrations*. 2016, pp. 163–168 (cit. on p. 26).

[164] M. Zhao, S. Zhang, W. Li and G. Chen. 'Matching Biomedical Ontologies Based on Formal Concept Analysis'. In: *Journal of Biomedical Semantics* 9.1 (2018), pp. 1–27 (cit. on p. 35).

Part V

APPENDIX

A USER STUDY PROCESS & QUESTIONNAIRES

For the sake of completeness, the following pages illustrate the study designs that were used in this thesis. At first, Appendix A.1 presents the questionnaire that was used to evaluate the visual berrypicking method using artificially generated images, see Section 11.1. Then, Appendix A.2 illustrates the online study interface that was used to evaluate the movie exploration tool as presented in Section 11.3. Finally, Appendix A.3 presents the questionnaire that was used to evaluate gaze and pupil parameters as potential indicators for personalization as presented in Section 11.4.

A.1 PRE-STUDY ON ARTIFICIALLY GENERATED IMAGES

User study for Visualization of and Navigation in large Image Collections

By means of this experiment, we intend to compare two different types of interfaces for image retrieval scenarios. Your task will be to use both interfaces in order to retrieve a given image from a collection of 2000 images. You will be always presented a smaller subset. By clicking on any image, the system will present you images from the collection that are most similar to the one you clicked. This will help you to explore and navigate the collection. The target image will be always visible in the top left corner of the screen. If you found the correct image (please click on it!), a popup will be shown and the task is solved! You will have to solve 4 retrieval tasks.

The images are simple geometric primitives (e.g. squares, triangles etc.) at different scale and in different colors.

We will measure the time and number of clicks you need to find the image requested.

The results of this study will be evaluated anonymously and used only for research purposes.

A small constraint: Please do not change the scaling in the Web Browser!

Statistics

Participant ID: __________ Start with: ☐ Map (odd id) ☐ List (even id)

Age: ______ years Profession/Study program (please be precise):____________

Sex: ☐ female ☐ male

Color blindness: ____________________

Figure 48: Experiment introduction, task description and demographic information.

List Visualization

a. Retrieval time:

b. Number of clicks required:

c. How understandable was the (re-)arrangement of the images during navigation?

☐ very clear ☐ clear ☐ neutral ☐ unclear ☐ very unclear

Please give a reason why so: ____________________

____

____

d. I thought the visualization was easy to use.

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

e. I found the visualization unnecessarily complex.

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

f. I found navigation using this visualization intuitive.

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

g. I thought there was too much inconsistency when new Images were shown.

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

h. I was able to efficiently complete my task using this visualization

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

i. The visualization did not support me in solving my task

strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

j. What would you change about the visualization?

____

____

____

____

Figure 49: Evaluation of list-based visualization recording retrieval time, number of clicks and several usability ratings.

Map Visualization

a. Retrieval time:

b. Number of clicks required:

c. How understandable was the (re-)arrangement of the images during navigation?
☐ very clear ☐ clear ☐ neutral ☐ unclear ☐ very unclear
Please give a reason why so: ______________________

d. I thought the visualization was easy to use.
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

e. I found the visualization unnecessarily complex.
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

f. I found navigation using this visualization intuitive.
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

g. I thought there was too much inconsistency when new Images were shown.
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

h. I was able to efficiently complete my task using this visualization
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

i. The visualization did not support me in solving my task
strongly disagree ☐ ☐ ☐ ☐ ☐ strongly agree

j. What would you change about the visualization?

Figure 50: The same evaluation form for the map-based visualization.

Other Comments

Figure 51: Some extra space for general comments of study participants.

A.2 ONLINE STUDY FOR MOVIE EXPLORATION

The following section provides a detailed illustration of every step of the movie explorer user study described in Section 11.3.

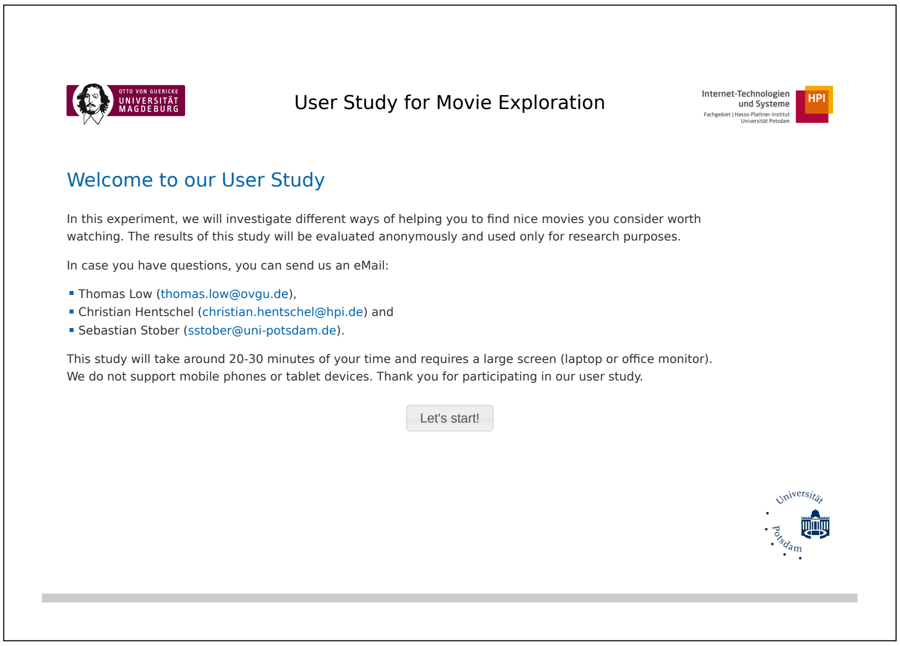


Figure 52: Start and introduction page for the movie explorer study: Participants are asked for 20-30 minutes of their continuous attention and assured that all results will be evaluated anonymously.

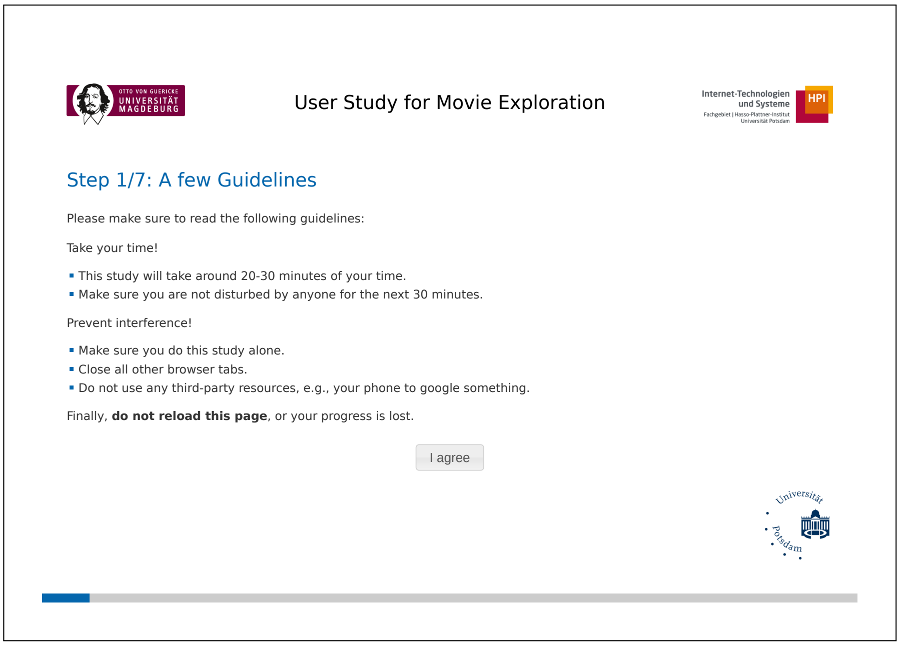


Figure 53: Participants are asked to prevent any possible disturbances while performing the study.

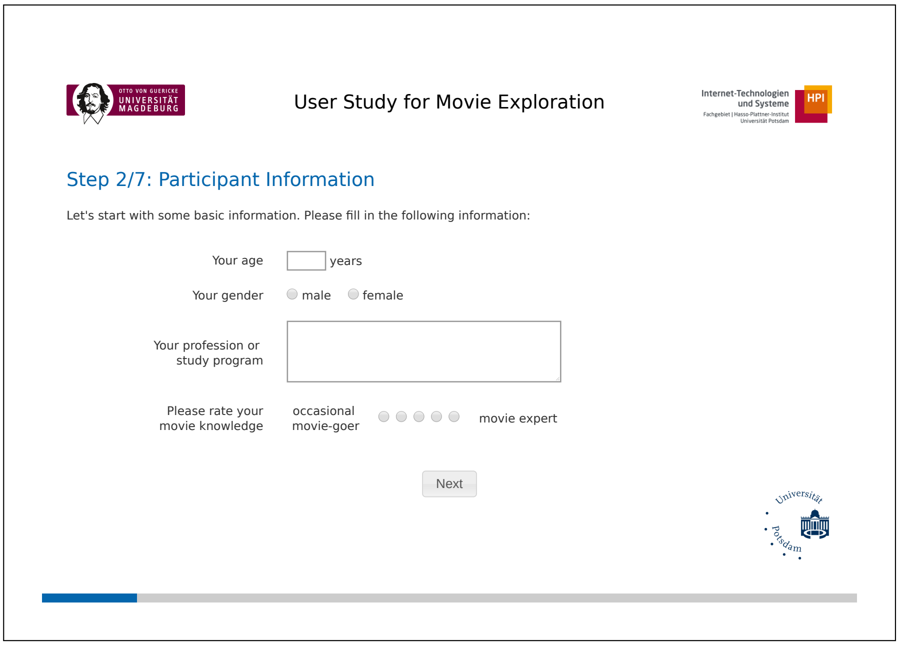


Figure 54: Participants are asked to submit demographic information and rate their own movie knowledge on a 5-point Likert scale.

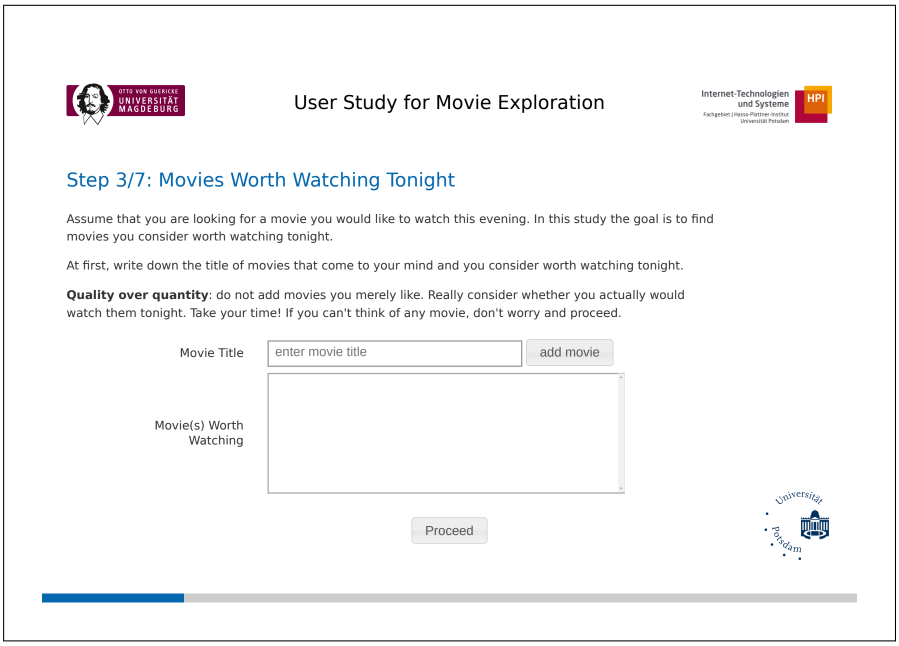


Figure 55: An introduction to the exploration task is given. At the same time, participants are asked to remember any movies they consider worth watching in order to prevent bias.

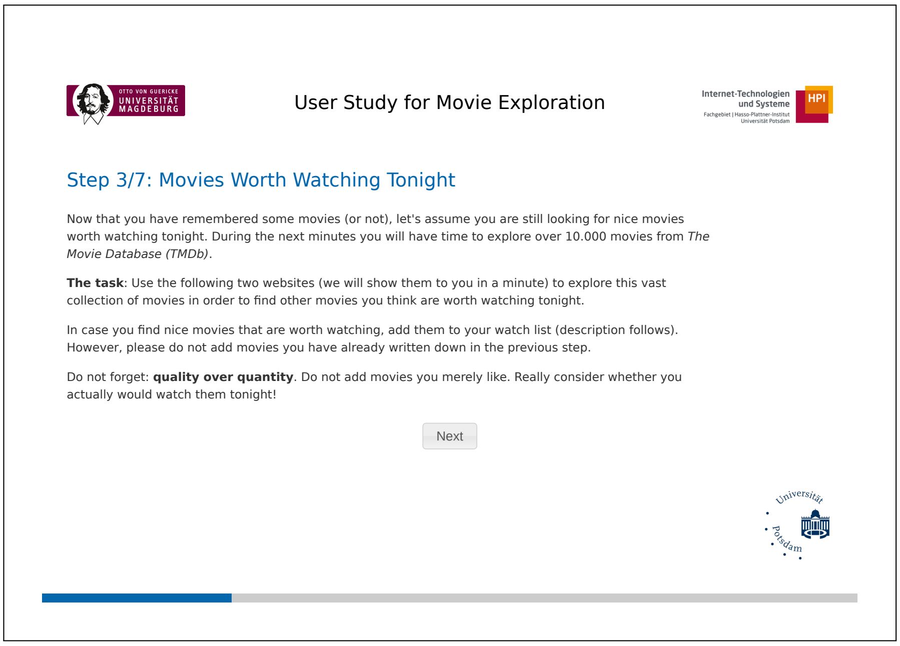


Figure 56: Participants are introduced to the task of using two different websites in order to find new movies they would consider worth watching tonight.

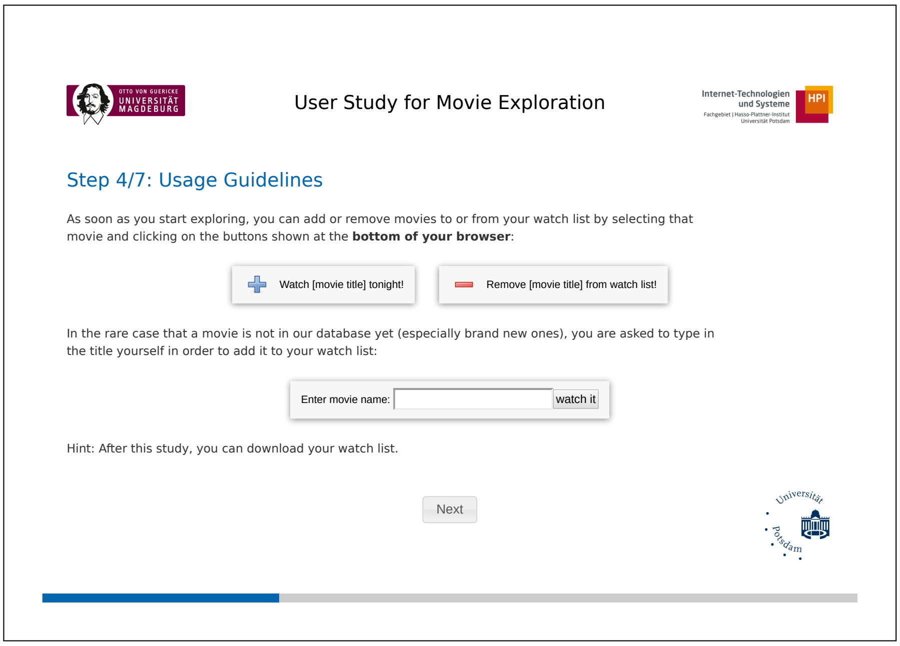


Figure 57: Study control buttons are explained: how to add or remove a movie from their watch list. In rare occasions, due to technical limitations, the movie title had to be entered manually.

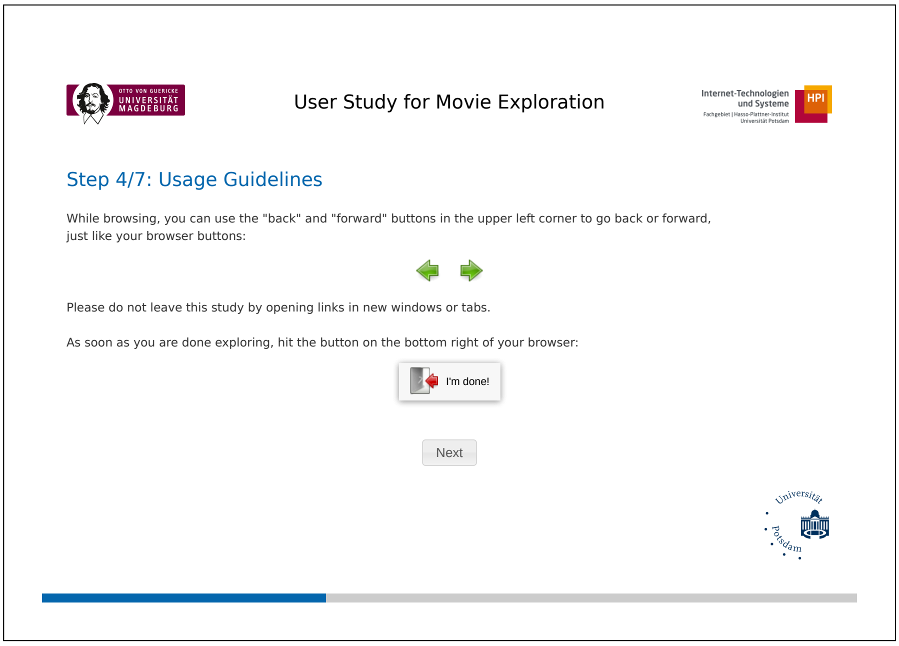


Figure 58: Additional study control buttons are introduced: traditional browser back and forward buttons as well as an exit button, which can be used to proceed with the study.

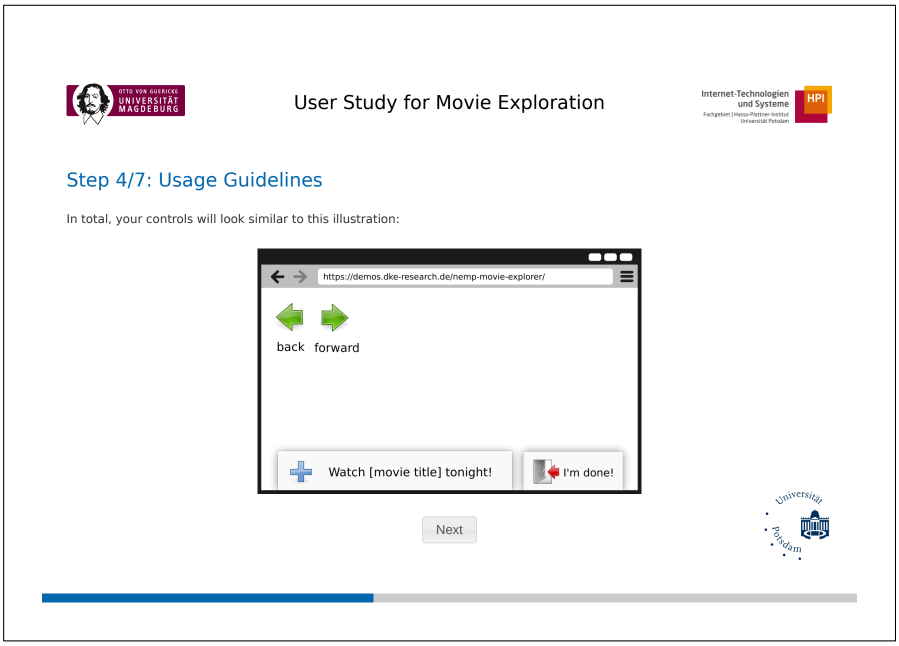


Figure 59: A summary of all study control buttons available to both the proposed prototype and as an overlay for TMDb.

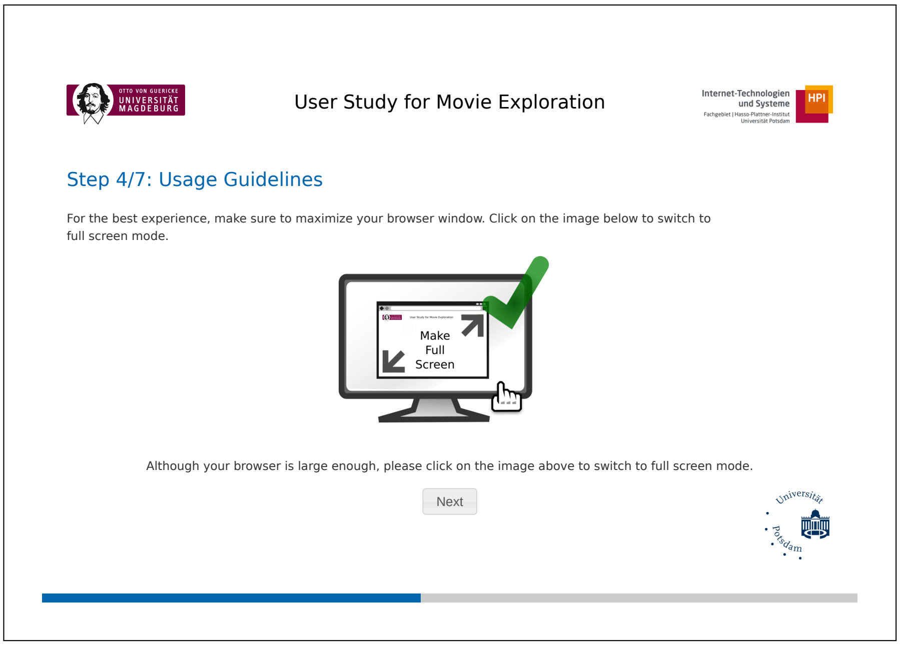


Figure 60: Participants were asked to enable the fullscreen mode of their browsers in order to avoid any distractions through open tabs or other programs.

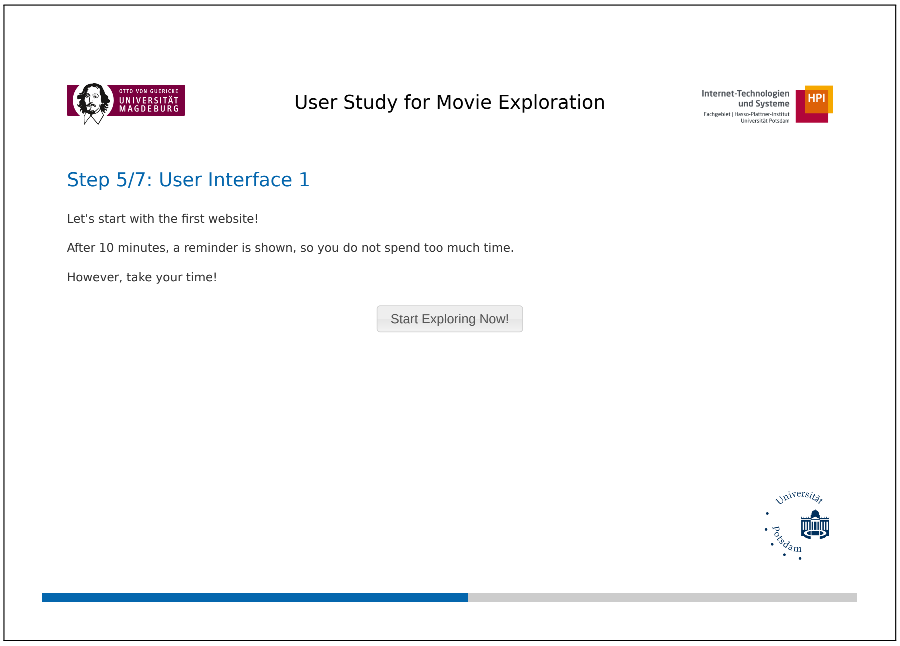


Figure 61: Participants are instructed to start exploring using the first interface, which was chosen randomly.

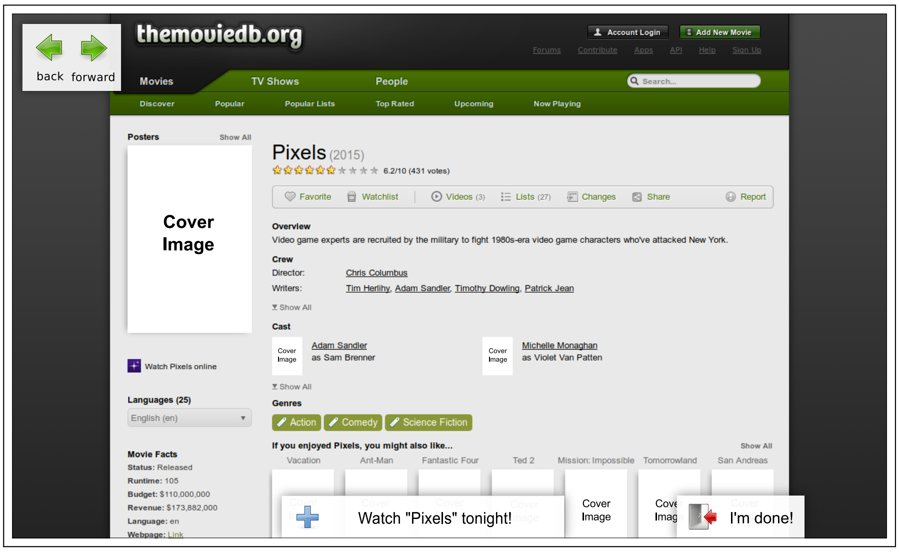


Figure 62: Participants may use the TMDb interface to find relevant movies, e.g., Pixels, and add them to their personal watch lists via the study control button on the bottom. Cover images removed due to licensing concerns.

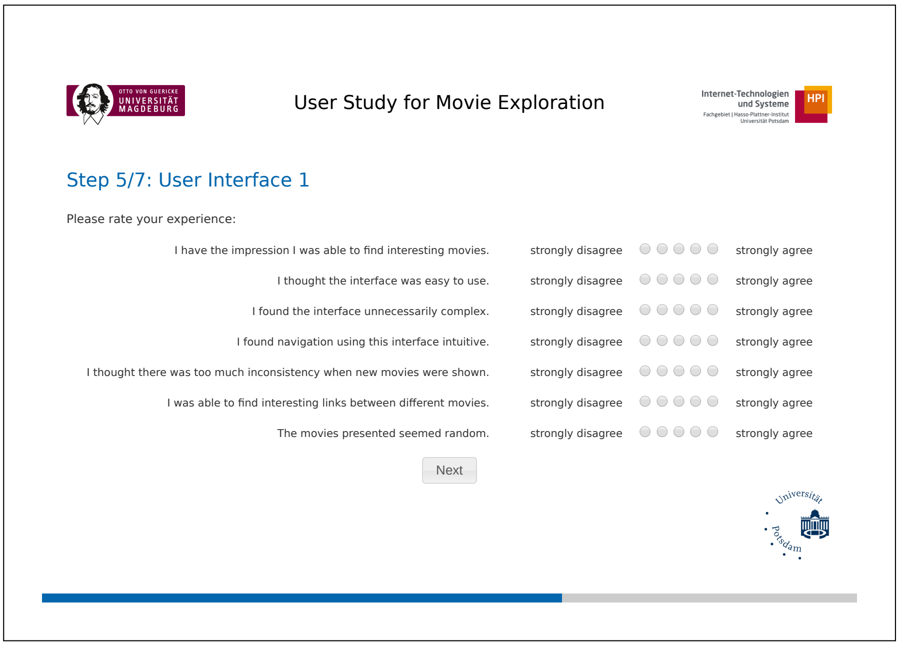


Figure 63: After exiting the exploration session, participants are asked to describe their experience by rating seven statements targeting the usability of each interface on a 5-point Likert scale.

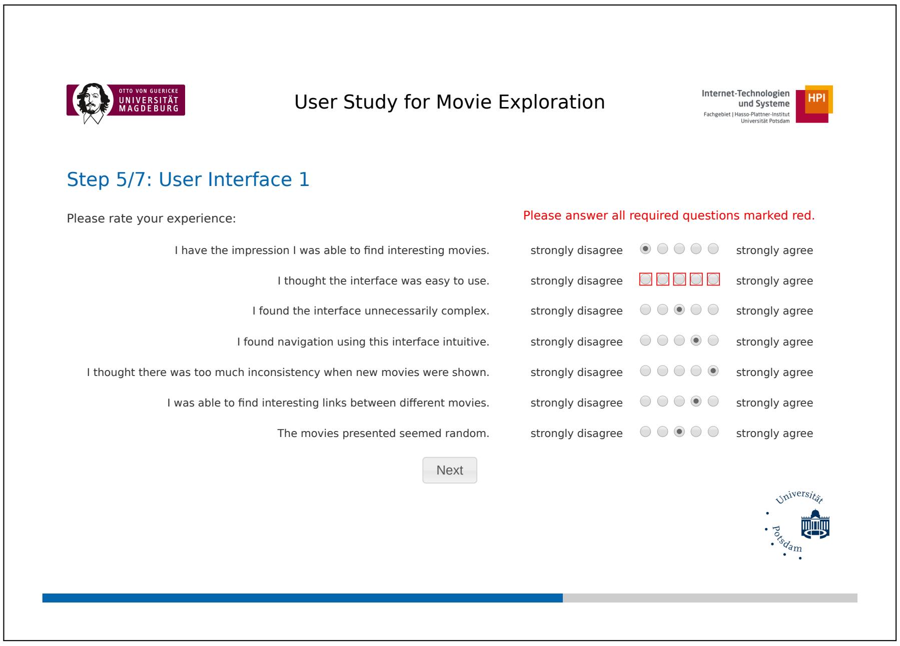


Figure 64: In case participants did forget to rate any of the statements, a short hint was shown in order to guarantee completeness of the study results.

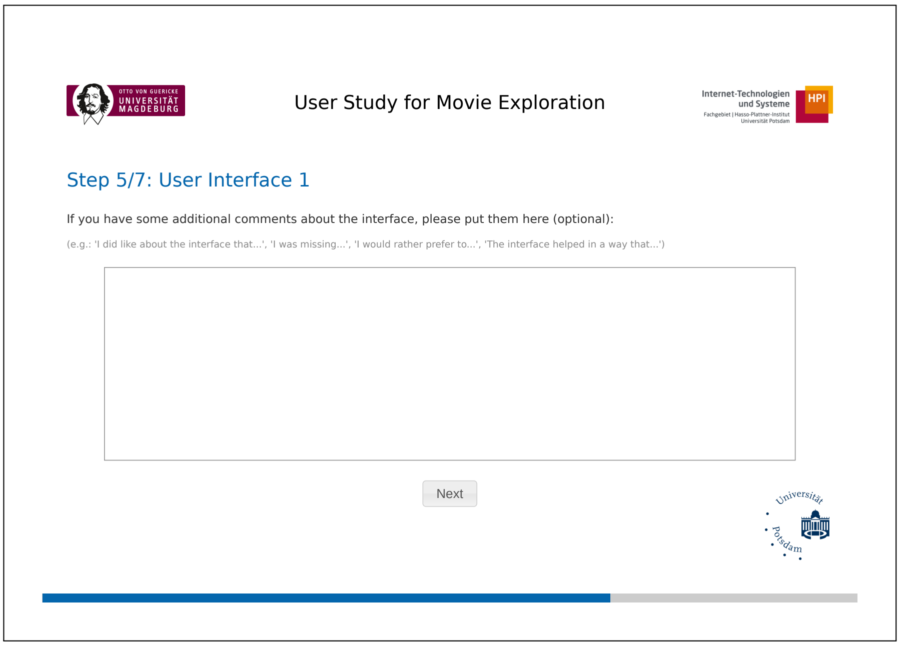


Figure 65: Participants were asked to provide any remarks or comments regarding their experience with the previous interface.

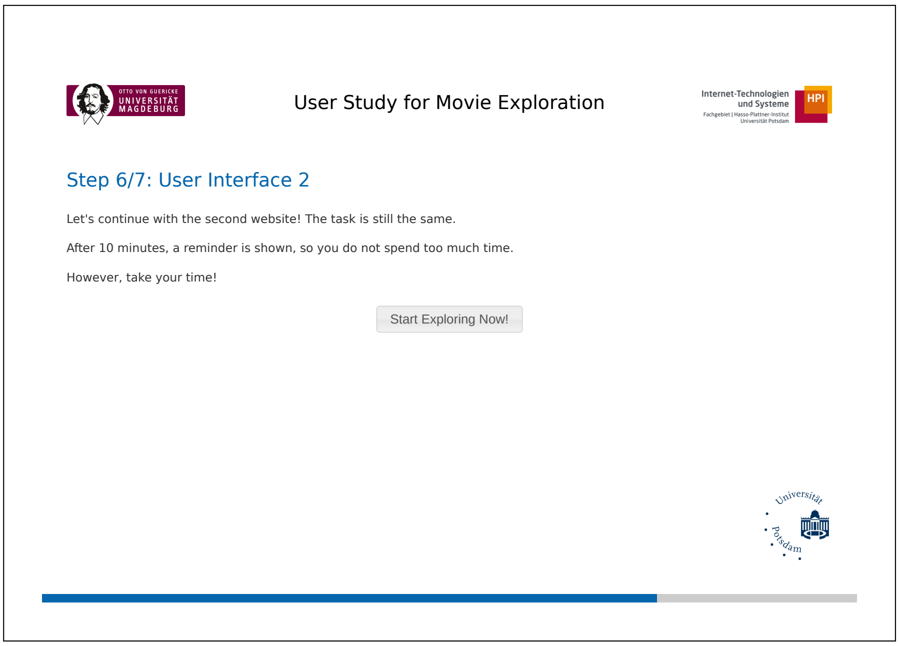


Figure 66: Participants are instructed to start exploring using the second interface.

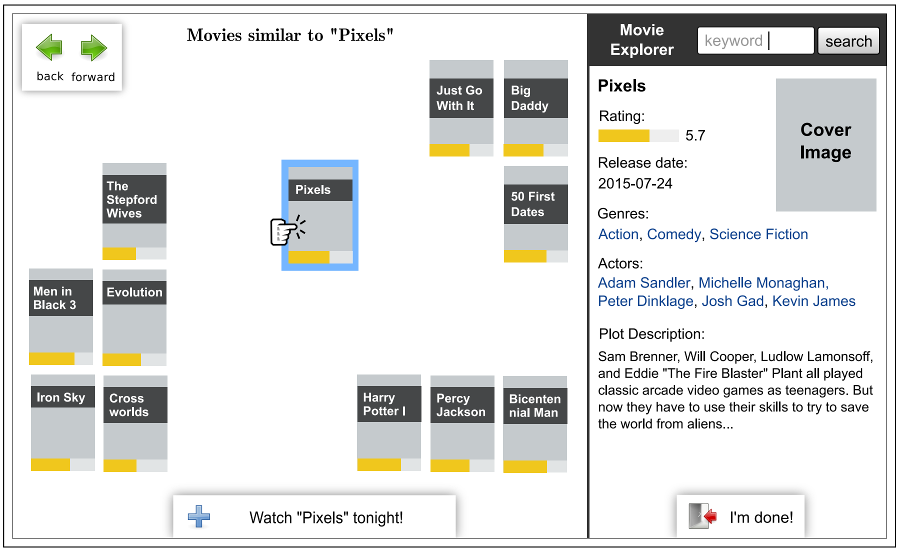


Figure 67: Similar as before, participants are enabled to discover movies and add them to their personal watch lists using the map-based interface. Movie cover image removed due to licensing concerns.

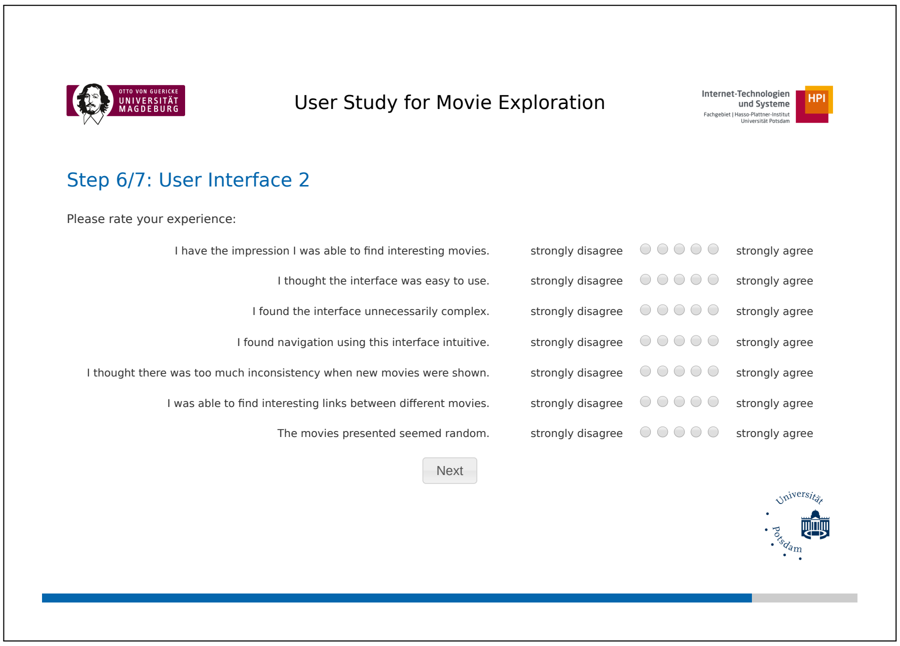


Figure 68: In the same way as before, participants are asked to rate the second interface using the same usability statements.

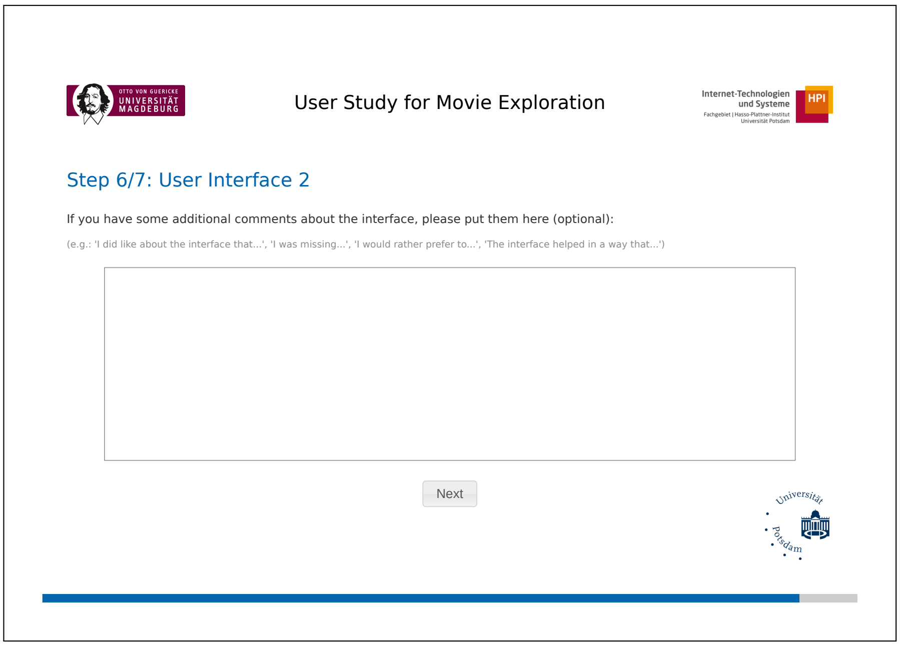


Figure 69: Again, participants are asked to provide any remarks or comments about their experience with the second interface.

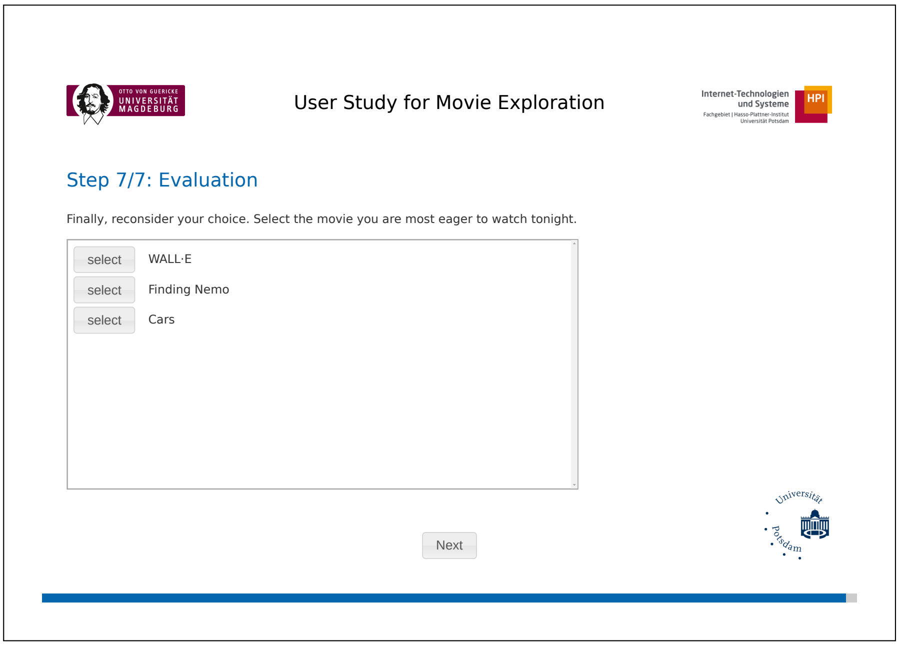


Figure 70: Participants are presented their full selection of movies they have been added to their watch lists in both interfaces, and asked to pick their final favourite.

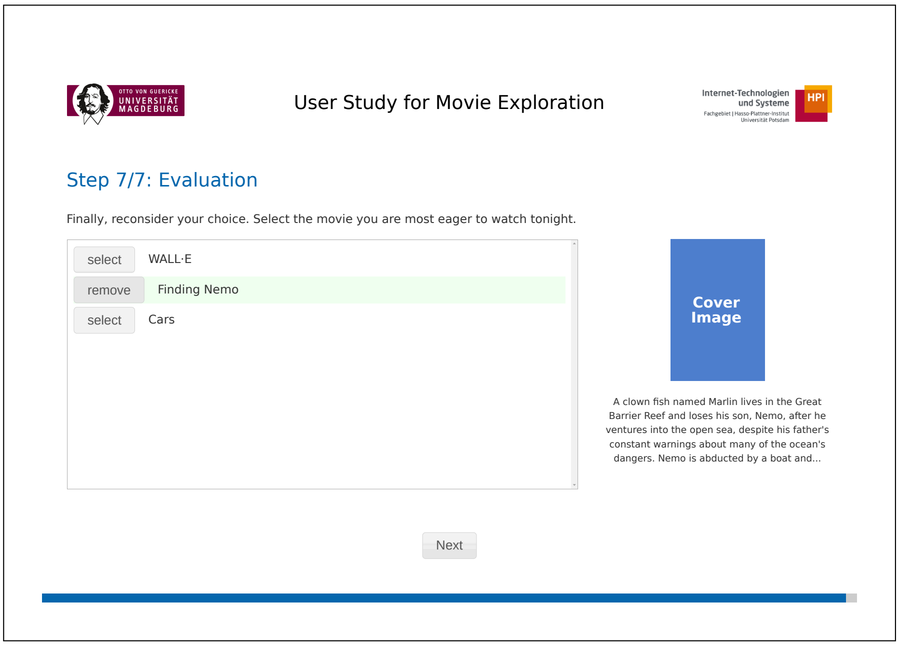


Figure 71: Selecting a final movie would again show basic information about that movie for reference. Cover image removed due to licensing concerns.

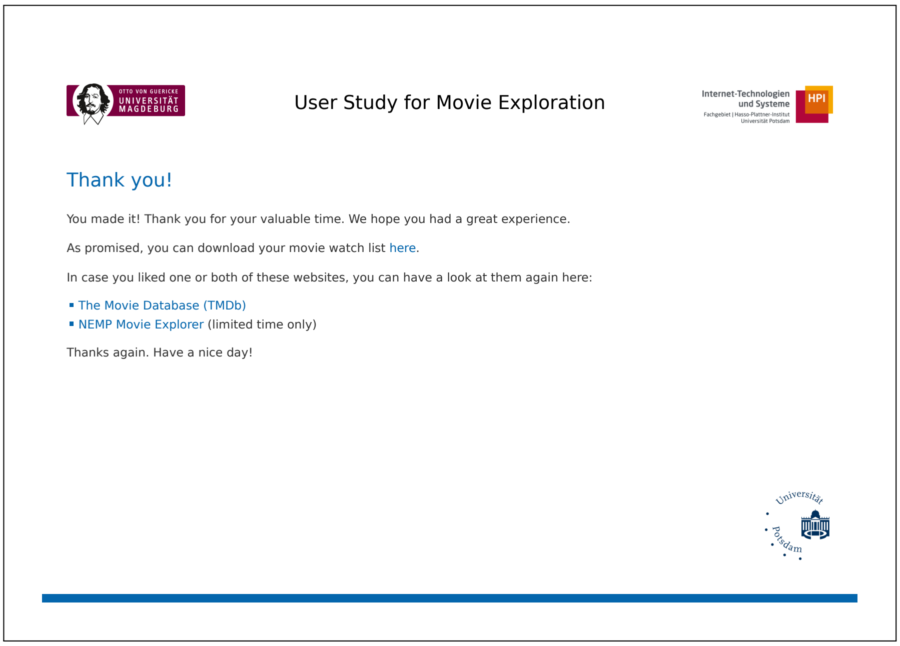


Figure 72: Finally, participants are thanked. The study is over.

A.3 TOWARDS USER INTENTION DETECTION USING GAZE TRACKING

Proband: Design: Datum:

Nutzerstudie zur Untersuchung des Blickverhaltens bei einer explorativen Informationssuche in Bildsammlungen

Die Nutzerstudie besteht aus zwei Teilen:

Teil 1: Ein Fragebogen zu demographischen Daten

Teil 2: Aufgaben zur Suche in einer Bildsammlung (Experiment)

Teil 3: Bewertung ihrer Nutzererfahrung

Die Ergebnisse der Studie werden **anonymisiert** ausgewertet, gespeichert und nur für Forschungszwecke verwendet, im Besonderen nicht für gewerbliche Zwecke.

Bitte kreuzen Sie die am ehesten zutreffende Antwort an: ☒ ja ☐ nein

Teil 1: Fragebogen zu demographischen Daten

1. Bitte geben Sie ihr Alter an: ______ Jahre
2. Geschlecht: ☐ weiblich ☐ männlich
3. Farbsehschwächen: ____
4. Welche berufliche Tätigkeit (bzw. welches Studium) üben Sie derzeit hauptsächlich aus? Bitte verwenden Sie eine genaue Bezeichnung, z.B. Masterstudium Informatik.

   ____

Teil 2: Experiment

Die von uns bereitgestellte Suchmaschine erlaubt die Exploration von großen Bildsammlungen. Dabei wird jeweils nur ein kleiner Teil der Sammlung angezeigt. Wenn Sie an einem konkreten Bild interessiert sind, können Sie es auswählen, indem Sie das Bild anklicken. Anschließend werden die Bilder der Sammlung, die dem von Ihnen ausgewählten Bild besonders ähnlich sind, dargestellt und entsprechend ihrer Ähnlichkeit neu angeordnet. So können Sie die Sammlung interaktiv erforschen bis Sie ein bestimmtes Bild gefunden haben.

Im Folgenden werden Ihnen 6 Suchaufgaben gestellt. In jeder Aufgabe wird ein Ziel-Bild dargestellt, das Sie mit Hilfe der Suchmaschine finden sollen. Das gesuchte Bild wird immer in der oberen linken Ecke des Monitors zu sehen sein. Die Aufgabe ist erfolgreich gelöst, wenn Sie das Ziel-Bild gefunden und angeklickt haben.

1/2

Figure 73: Experiment introduction, task description and demographic information.

Proband: Design: Datum:

Teil 3: Bewertung

Bitte bewerten Sie ihre Erfahrung mit unserer Suchmaschine.

a. Wie verständlich empfanden Sie die (Neu-)Anordnung der Bilder während der Navigation?

☐ sehr verst. ☐ verst. ☐ neutral ☐ etwas unverst. ☐ sehr unverst.

Bitte begründe wieso: ____

____

____

b. Ich empfand die Visualisierung als leicht zu benutzen.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

c. Ich empfand die Visualisierung als unnötig komplex.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

d. Ich empfand die Navigation mit Hilfe der Visualisierung intuitiv.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

e. Ich finde es gab zu viel Inkonsistenzen, wenn neue Bilder gezeigt wurden.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

f. Ich konnte meine Aufgabe mit Hilfe der Visualisierung sehr effizient lösen.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

g. Die Visualisierung hat mir nicht dabei geholfen, meine Aufgabe zu lösen.

trifft voll zu ☐ ☐ ☐ ☐ ☐ trifft überhaupt nicht zu

Weitere Anmerkungen

____

____

____

____

____

____

Danke für Ihre Unterstützung!

2/2

Figure 74: Usability evaluation statements and additional comments from study participants.