Improving Exploratory Search Experience through Hierarchical Knowledge Graphs

Bahareh Sarrafzadeh
Cheriton School of Computer Science
University of Waterloo, Canada
bsarrafz@uwaterloo.ca

Edward Lank
Cheriton School of Computer Science
University of Waterloo, Canada
CRIStAL, University of Lille, France
lank@uwaterloo.ca

ABSTRACT

In information retrieval and information visualization, hierarchies are a common tool to structure information into topics or facets, and network visualizations such as knowledge graphs link related concepts within a domain. In this paper, we explore a multi-layer extension to knowledge graphs, hierarchical knowledge graphs (HKGs), that combines hierarchical and network visualizations into a unified data representation. Through interaction logs, we show that HKGs preserve the benefits of single-layer knowledge graphs at conveying domain knowledge while incorporating the sense-making advantages of hierarchies for knowledge seeking tasks. Specially, this paper describes our algorithm to construct these visualizations, analyzes interaction logs to quantitatively demonstrate performance parity with networks and performance advantages over hierarchies, and synthesizes data from interaction logs, interviews, and thinkalouds on a testbed data set to demonstrate the utility of the unified hierarchy+network structure in our HKGs.

CCS CONCEPTS

•Information systems →Search interfaces;

KEYWORDS

Knowledge Graphs, Hierarchies, Exploratory Search, Information Seeking, Representations of Search Results

1 INTRODUCTION

Finding information on the Web is often difficult. There are two predominant paradigms for finding information on the Web: Searching (i.e, Search by query) and Browsing (i.e, Search by Navigation) [20, 31]. While current search engines, following a "search by query" paradigm, are generally sufficient when the information need is well-defined in the searcher's mind, examining search results remains a necessary step within a larger information seeking process [23, 25]. To elaborate, Searching requires the user to translate an information need into queries, while Browsing accommodates the knowledge gap between what the user is able to communicate and what the system requires to find the desired information. This knowledge gap (also formalized as an 'anomalous state of knowledge' by Belkin [5]) is more evident when information is sought to address broad curiosities, for learning and other complex mental activities [2, 43].

SIGIR '17, August 07-11, 2017, Shinjuku, Tokyo, Japan

DOI: http://dx.doi.org/10.1145/3077136.3080829

This paper focuses on the design of tools to support browsing. Researchers note that, while search interfaces must support query formulation and an effective ranking algorithm, browsing interfaces require effective representations of search results to accommodate searcher's 'state of knowledge' and provide 'information scent' to guide the user during navigation [31, 32]. Related to this observation of 'information scent' is the observation that information seekers often express a desire for a user interface that organizes search results into meaningful structures to facilitate browsing and understanding of the retrieved results [15].

The desire for browsing support via structure has given rise to interfaces that represent the structure of information. These structures typically exist in one of two forms: hierarchies and networks [11]. First, information can be presented into hierarchies based on categories. Included in this class of search results organizations are techniques such as faceted browsing or automatic clustering. Second, alongside hierarchies, entity relationship or network representations are also used. In these representations, rather than clustering objects into labeled categories, the connections between objects (people, places, things, etc.) in a document corpus represent a relationship between two items. These network representations include knowledge graphs [18] and concept maps [29].

Recent work has explored the relative benefits of hierarchies and networks and has noted that the benefits are largely complementary: hierarchies provide users with some understanding of central topics, allowing them to develop a better overview of information; whereas networks allow people to glean concrete information from the representation rather than needing to extensively read individual documents [39]. Given the complementary advantages of knowledge graphs and hierarchies, our main research question in this paper is that whether we can algorithmically generate a seamless data structure that combines the advantages of both hierarchies and networks into a single unified structure.

In this paper, we evaluate the efficacy of hierarchical knowledge graphs (HKGs) as a combined representation of low-level entity relationships and high-level central concepts. We generate these knowledge graphs automatically using a simple parsing algorithm [37], then extract hierarchies using a dynamic thresholding approach. We evaluate these HKGs using a mixed methods approach. Quantitative data argues that HKGs preserve the transparency advantages of knowledge graphs and structural advantages of hierarchies. Qualitative data triangulates with quantitative observations and provides additional insight into the advantages and disadvantages of both hierarchical and network visualizations.

2 RELATED WORK

In this section we survey related work in classifying and understanding web search and in techniques to augment search results.

2.1 Understanding web search

There exist several characterizations of web search queries [6–8, 35, 44]. Marchionini [24] focuses specifically on a process he terms *exploratory search*. Marchionini defines three categories of web search: Look-up, Learn, and Investigate and groups Learn and Investigate tasks under the umbrella of exploratory search.

Learn and Investigate tasks involve sensemaking [14, 36]. Sensemaking is the process of searching for a representation and encoding data in that representation to answer task-specific questions [36]. Russell et al. [36] present four main cognitive stages that are involved in sensemaking: an interaction between a bottom-up 'Search for representation' phase and a top-down representation instantiation phase, a phase of shifting representations to fit newly discovered data into the current representation, and an application of the representation to the user's specific task.

The top-level representation that results from sensemaking has many different names: a "holistic cognitive structure" [42], a mind-map, a concept map [29]. Regardless, the construction of this representation must be either provided explicitly by the interface or constructed implicitly by the user [36, 42] before the user can fully "make sense" of the retrieved information.

Alongside the distinction between "look-up" and exploratory search, there are also varying levels of complexity associated with exploratory search tasks [24]. At the simplest level, one might simply wish to read a document to understand a topic. However, consider tasks such as comparing two different diets to understand health benefits (e.g. oriental and Mediterranean), comparing two different political systems to understand relative authoritarianism (e.g. President of Russia versus Iran), or comparing two different educational systems to understand relative student success (e.g. British versus Canadian). All of these, as examples of exploratory search tasks, require both searching and browsing to identify information, acquire knowledge, and contrast related data across two contexts. As the complexity of comparison increases (three or more alternatives, more abstract concepts), the information seeking task continues to increase in complexity. Searching is highly effective at identifying relevant documents to guide basic exploratory search, but browsing and reformulating are needed to fully acquire and synthesize knowledge [23, 26]. Therefore, understanding both searching and browsing behaviors are of vital importance to design effective interactive information retrieval systems.

2.2 Organizing Search Results

Because of the importance of structure in search, there have been efforts to contrast strengths and weaknesses of different spatial representations and groupings of search results. A taxonomy of techniques for organizing search results was proposed by Wilson et al. [46]. They identify two main classes of approaches: (1) Coupling results with additional metadata and classifications such that searchers can interact and control the presentation of results. (e.g. faceted browsing or categories), or (2) providing alternative or complementary representations of search results (e.g, a network representation). Wilson et al. also present four common approaches to structured classification [46]: hierarchical classifications, faceted classifications, automated clustering, and social classifications.

Looking first at structured classification, early forays into the domain of structuring search results contrasted categories with automatic clustering to support search. Hearst [16] showed that categories, because they were more interpretable for the user, captured important information about the document but became unwieldy when the document corpus was too large. Clusters, by comparison, were highly variable with respect to quality and were often less meaningful for the user.

Given the lack of intuitiveness associated with clustering [17] and a desire for understandable hierarchies in which categories are presented at uniform levels of granularity [33, 34], alongside specified hierarchies such as tables-of-contents, researchers have explored faceted categories, i.e. categories that are semantically related to the search task of the user, to organize search results. These include systems that define faceted categories [48], research that studies the use of facets to support browsing [9], and research that identifies strengths and weaknesses of faceted browsers [45]. In terms of strengths and weaknesses, faceted browsing has proven beneficial for users already clear about their search task [45]; additional information on interactions between facets (e.g. inter-facet relationships) is helpful when users are unfamiliar with a domain and need 'sensemaking'. In other words, exploratory tasks (e.g.learning or investigating [24]) are precisely those tasks where interactions between facets are needed.

The need to represent relationships between facets or concepts has given rise to the use of network structures to depict relationships between concepts or entities in a corpus. These network structures include concept maps, knowledge graphs, and other entity-relationship diagrams. Concept mapping has been widely used in education as a method for knowledge examination, sharing, and browsing [10, 29]. Knowledge graphs have been popularized by Google to represent web-based information. One drawback to network structures is it is hard both to get an overview of an information network and to navigate through the network effectively: users are easily "lost" in these systems [12, 22, 31].

A final question within this space is how competing representations fare in presenting results for exploratory search tasks. While some past research has explored using questionnaires to determine the efficacy of different knowledge representations [30], or has evaluated the efficacy of hierarchies, networks, or concept maps with respect to ordered lists (e.g. [1, 9, 10, 27, 38]), we have found little research that directly compares networks to hierarchies to understand their competing affordances. The one exception to this is recent work by Sarrafzadeh et al. [39]; they find that networks eliminate the need for reading documents – users can glean information from the networks with statistically significantly less time spent reading – and that hierarchies particularly benefit low-knowledge participants by giving them an effective overview of the domain.

Given Sarrafzadeh et al.'s observation of the complementary benefits of hierarchies and networks, one question is whether studies have examined the use of hierarchies and networks as combined – synchronized and simultaneous – representations of search results. We have found little work that explores combined networks+hierarchies. Part of the challenge may arise from the complexity of seamlessly integrating both hierarchies and networks into a single unified structure. For example, hierarchies are typically best when the structure aligns well with the user's task, but, given this alignment, entities in networks may have many multiple 'parents' within the structure, yielding a many-to-many relationship within the hierarchy, i.e. a three-dimensional graph.

3 HIERARCHICAL KNOWLEDGE GRAPHS

In this section, we describe hierarchical knowledge graphs, an extension of knowledge graphs that include hierarchical information about the lower level graphical structures. The rationale behind our proposed approach for employing hierarchical knowledge graphs to represent search results is the complementary benefits [39] of hierarchies [17, 47] and network structures [2, 29, 38] to support exploratory browsing of search results. More specifically, hierarchies provide a breadth-first exploration of the information that allows the user to iteratively reduce confusion, obtain an overview, and slowly exploit detail. They thus provide a structured way to navigate from more general concepts to more fine grained data and are valuable when people feel a need to orient themselves. In contrast network structures allow users to glean more information from the representation (document reading time is reduced), are more engaging, yield more control over exploration at the lower level of inter-concept relationships [39], and are more similar to one's mental model [3, 11, 28, 39].

Given the complimentary benefit of networks and hierarchies, the next question is how to design a representation that can seamlessly merge these two representations. We take the approach that a knowledge graph will be an appropriate low-level representation and seek to incorporate a hierarchical view of this low-level representation of corpus content. To incorporate a hierarchical view into a knowledge graph, we need to find answers to the following three design questions (DQs):

(1) How do we integrate network and hierarchical views into a single, seamless data structure?
(2) How can both the global and the local view of a knowledge graph be co-visualized?
(3) How can transitions between views be designed to maximize visualization stability?

To answer these DQs, we first focus on DQ1 and describe the design of our data structure. Next, to address DQ2 and DQ3 we describe an interface that supports interaction with the data structure. Alongside our DQs, we add one additional constraint to our design. We want to ensure that both the low-level knowledge graph and the hierarchies gleaned from that knowledge graph can be automatically generated from a targeted search performed by the user.

3.1 Visualization Design and Creation

As noted above, given that we take the approach that a knowledge graph will constitute the lower-level visualization of our data, the task becomes creating a knowledge graph and creating a hierarchy that is gleaned from and corresponds directly to the underlying knowledge graph.

Figure 1 depicts the system architecture that supports the process of automatically generating the hierarchical knowledge graph representation. To simplify hierarchy generation, we create a 3-level hierarchy for any document corpus. Beyond the base layer knowledge graph, there is an intermediate layer of central concepts gleaned from the knowledge graph. Finally, at the top-level, the documents, themselves, represent the top level of the hierarchical knowledge graphs. In Figure 1, three main steps are depicted to generate hierarchical knowledge graphs: Document Retrieval (yielding the top-level of the hierarchy), Knowledge Graph Generation (yielding the bottom level of the hierarchy), and Hierarchy-from-graph Generation (yielding an intermediate view of an individual knowledge graph, which we dub a *minimap*[1]).

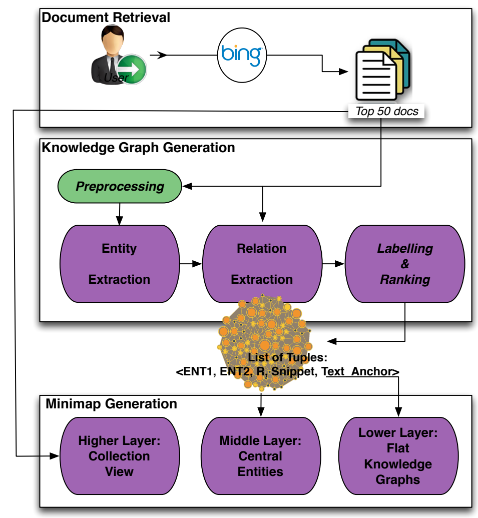

Figure 1: Generating Hierarchical Knowledge Graphs

3.1.1 Document Retrieval. The Document Retrieval component aims at creating an initial document collection based on a user's query. This collection will then be used as an input for the Knowledge Graph Generation component and will represent the top view of the target hierarchy.

To generate a document corpus, we use the Bing Search engine to retrieve the top n documents for a query while attempting to ensure a reasonable quality of information in the retrieved documents. By default, to ensure that retrieved documents are consistent in their credibility and coverage, we specify Wikipedia as the target domain. Furthermore, because it is known that searchers typically view only a few results [19] and rarely stray past the first page of results [4], we selected n=10 documents to generate collections. The target domain from which to glean documents (e.g. a user might specify webmd for medical documents, 'gov' for public policy documents, 'bbc' for news) and the size of the initial collection can be specified by the user at the time of query submission. Finally,

[1]The term minimap is drawn from the gaming literature. It represents a less detailed overview of a gaming world, allowing the user to orient themselves.

since most exploratory search tasks require multiple queries to retrieve documents for different aspects of the information need, this component assigns one partition per query so the user can narrow down the retrieved collection further.

3.1.2 Knowledge Graph Generation. Inspired by past work [37], we create knowledge graphs for an individual document or set of documents as follows. (1) Entity taggers [2] are used to extract entities from text. (2) Sentences that contain at least two entities are selected and parsed using the Stanford Dependency Parser. For each sentence, we extract meaningful relations between the entities by finding the shortest path in the corresponding parse tree. (3) Finally, labels are automatically generated for the extracted relations. The labeled relations are ranked based on relevance to the query and the informativeness of the extraction [37].

The outcome is a set of tuples in the form of <entity1, entity2, relation, snippet, document_anchor>. These tuples collectively correspond to a knowledge graph representation of retrieved documents where *entity* is usually a term or a noun phrase in text that corresponds to a concept in the domain, *relation* corresponds to a simplified sentence that is semantically complete and describes how entity1 and entity2 are connected, *snippet* is a short portion of text from which the corresponding entity pair and the relationship is derived, and *text_anchor* is an HTML anchor that links the extracted tuple to the corresponding portion of the source document in the collection. For example, from a paragraph on powers and responsibilities of a president the following tuple can be extracted: <president, parliament, "President nominates the Cabinet members to the Parliament", snippet, [URL][anchor]>.

These tuples are visualized as a knowledge graph where nodes are the entities and edges are the relationships between them. This visualization constitutes the lowest layer of the hierarchy.

3.1.3 Minimap Generation. The final component of this system generates a hierarchical representation of the search results by extracting a middle layer from the input Knowledge graph tuples and provides bidirectional mappings between all three layers. As noted earlier, we call this layer the minimap layer.

A natural result of the entity-relationship tuples extracted above is that some entities have a higher number of edges, i.e. are of higher degree. A higher edge count implies a larger number of connections to other entities in the graph; in other words, those entities with higher edge counts were more frequently linked with other entities in the document. We call these higher degree vertices *central concepts* and hypothesize that one alternative to hierarchical faceted structures is to consider a multi-level view of a knowledge graph around central concepts. The multilevel view focusing on central concepts simply introduces information seekers to those entities or objects that are most frequently linked to other entities within the corpus. Generating the hierarchy becomes a thresholding task to appropriately scope the intermediate level of the visualization. Algorithm 1 describes this process more formally.

3.2 Prototype Development

Given our hierarchical representation (DQ1), we must support mechanisms for viewing and interacting with the visualization

[2] https://cogcomp.cs.illinois.edu/page/software_view/NETagger

Algorithm 1 Extracting Central Concepts

Require: *Nodes*: array of nodes in the knowledge graph, *min_degree*: a pre-specified threshhold for the minimum degree of node to be considered as a central concept (starting value = 3), *max_count*: an experimentally derived threshold for the maximum number of Central Concepts to be included in the middle layer (default value = 15).

```
function ExtractCC(Nodes, min_degree, max_count)
    while true do
        CentralNodes ← []
        for all node in Nodes do
            if node.degree ≤ min_degree then
                CentralNodes.add(node)
        if CentralNodes.size() ≤ max_count then
            return CentralNodes
        min_degree++
```

(DQ2 and DQ3). In information retrieval, it is difficult to separate any visualization for representing search results from the interface that contains that visualization [17]. We iteratively designed an interface to support navigation of our hierarchical knowledge graphs via a series of pilot studies.

Based on established literature and pilot studies we found that knowledge graphs can become overwhelming or confusing for participants [12, 22, 31, 39]. The overwhelming nature of the full knowledge graph leads to a need to create filtered views of our graph. These filtered views draw inspiration from the "expand-from-known" paradigm in information visualization [41]. Specifically, at the top level of the full corpus, a user selects a document, then a central concept from the minimap visualization. While preserving the entire knowledge graph, we alpha-blend all nodes in the knowledge graph *except* those nodes directly related to the central concept from the minimap. Recall that the central concept is simply a high-degree vertex from the knowledge graph; therefore, the central concept and all its linked nodes are shown saturated. As a result, users can identify the central concept, linked entities, and can see closely related additional entities. Together, this focused detailed view seems to effectively support expand-from-known at the knowledge graph level.

As well, for the Hierarchical View, the biggest challenge to address was the disorientation among the participants during transitions between collection, minimap, and knowledge graph views, a common problem in interfaces that show multiple levels of abstraction. To address this disorientation (DQ3), we maintained the connection between the hierarchical view and the graph view in two ways. First, the user can move between the layers of Collection View and the Document View smoothly through a zooming functionality that changes the focus of the UI (see Figure 2). Second, the interplay between the Document View and the Detailed View is designed such that the overview of the document is present at all times, in terms of a callout on the left side of the screen, an actual *minimap* as in computer gaming, which allows the user to maintain a sense of where he or she is while manipulating the fine-grained nodes and edges in the Detailed View.

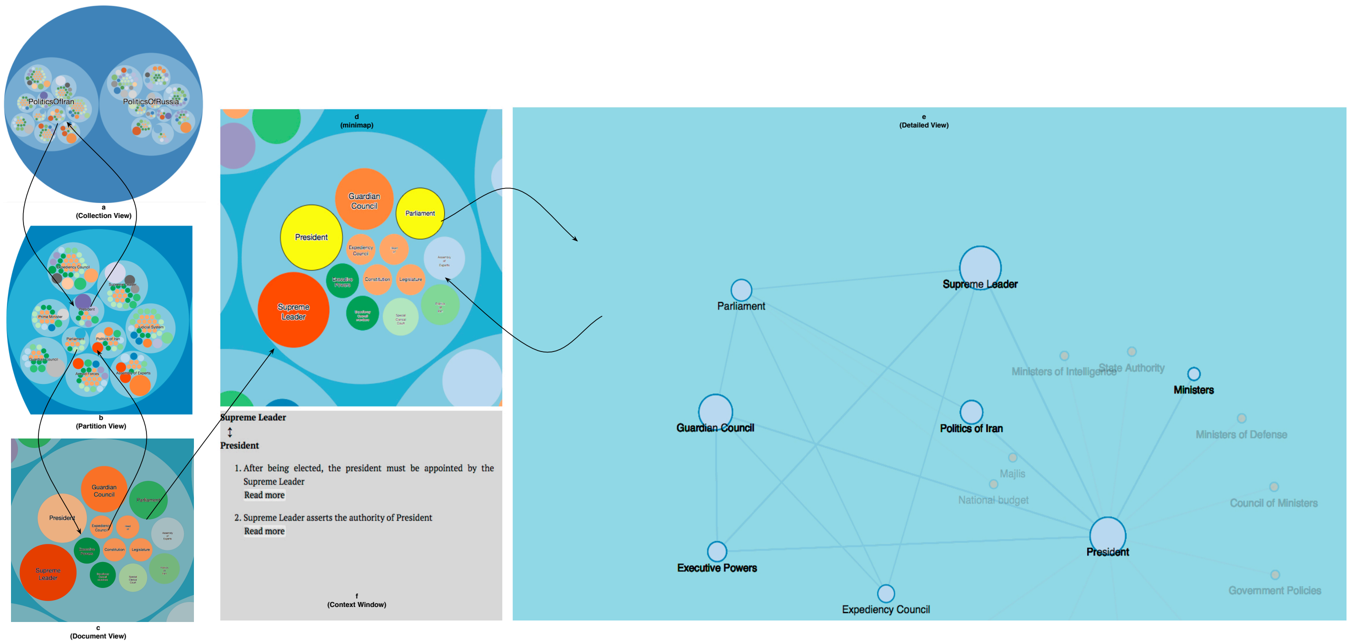

Figure 2: MultiLayer Graph Interface: (a) Collection View; (b) Partition View; (c) Document View; (d) Minimap (i.e. Global View); (e) Detailed View (Local View); (f) Snippet Window

The iterative process culminated in the final prototype shown in Figure 2.

In this interface, we see an initial overview, the Collection View that presents an overview of the underlying documents' structure in the collection (Figure 2-a). The collection view can potentially provide multiple partitions on the documents. Figure 2-b illustrates one partition of a collection. As an information seeker drills down on each document, the view is altered (Figure 2-c) such that an overview of the document is presented. The Document View provides a Global View of the corresponding document in terms of its central concepts. In this overview, the salient concepts in that article are visualized as circles of different sizes, where size indicates the frequency of occurrence in that article. We used force and pack layouts (as part of the D3 library[3]) to visualize the different layers of the knowledge graph representation.

The lowest layer of our representation is the Detailed View (Figure 2-e). This view is a knowledge graph that represents entities and relationships between them. The Detailed View, similar to Sarrafzadeh et al.'s graph interface [39], contains labeled nodes and unlabeled links between nodes. Nodes that represent entities with low frequency are hidden in the initial view, and only appear once a higher-frequency, connected node is clicked, ensuring that the graph does not become too cluttered. Once the user hovers over a node, that node and all connected nodes are highlighted, while the remainder of the graph is alpha-blended into the background. Clicking on a node can expand it by adding in its related nodes. Alternatively, clicking on a node can collapse its neighbours if they are expanded already. Nodes can also be dragged and placed at different parts of the canvas. This functionality can help with organizing the graph structure in a way that is more meaningful to the user and it can help with minimizing label overlap in the graph.

Edges can similarly be highlighted by hovering. By clicking on any edge, the user can see the relationship(s) between the two corresponding nodes (linked by this edge) in the context window located on the lower left side of the interface (Figure 2-f). For each relationship in the context region, a hyperlink allows users to view the corresponding web page.

[3]http://d3js.org/

4 EXPERIMENTAL DESIGN

Given that we have designed hierarchical knowledge graphs, a related question is how hierarchical knowledge graphs compare to hierarchies and/or knowledge graphs with respect to information seeking tasks. To evaluate this question, we need a set of control interfaces (reference interfaces that can be compared to hierarchical knowledge graphs, HKGs) and a reference data set. These can then be leveraged to design an experiment. As well, experimental design should replicate, as closely as possible, past work to ensure experimental validity.

In recent work, Sarrafzadeh et al. [39] developed two interfaces for exploratory search: one knowledge graph interface and one hierarchical tree interface. To preserve experimental validity, we use identical interfaces as control interfaces. We also leverage the identical data sets, ensuring that topic is eliminated as a confound. Finally, we use exactly the same experimental task, ensuring that performance numbers are representative between experiments.

4.1 Control Interfaces

- The first interface, a knowledge graph interface, functions as follows: As the interface starts, nodes that represent entities with low frequency are hidden in the initial view, and only appear once a higher-frequency, connected node is clicked. Users can also filter the knowledge graph by clicking on a node; when a user clicks on an edge, snippets and links associated with that edge are shown in a preview pane on the left side of the interface.
- The second interface utilized a hierarchy (or a tree) structure to organize headings and sub-headings of the articles, as observed in each page's table-of-contents. When the user launches the application, the user is presented with a fully expanded tree. By clicking on any node within the tree, that portion of the Wikipedia document corresponding to the node is presented in the preview area at the left of the interface.

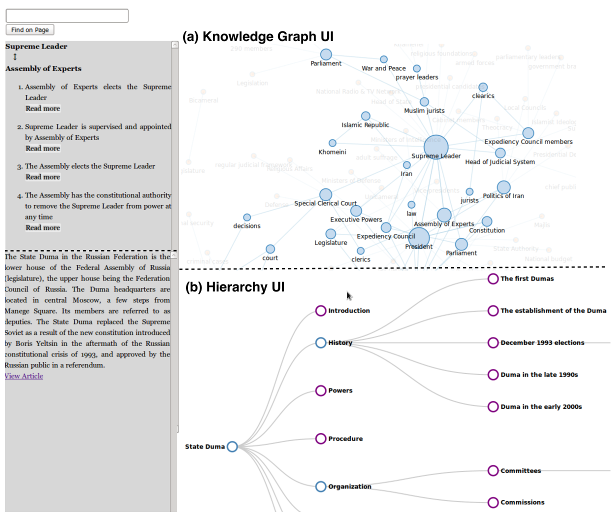

Figure 3: Control interfaces for Knowledge graph and Hierarchy. More details in Sarrafzadeh et al. [39].

Figure 3 depicts these two interfaces. Contrasting these interfaces with Figure 2 shows a similar preview pane for snippets. Links within the snippets function identically across all three interfaces.

4.2 Data Set

We leveraged the two data sets from Sarrafzadeh et al.'s previous study: A history data set, specifically a corpus of Wikipedia articles describing the historical locations of the capital city of Canada; And a global politics data set, a Wikipedia corpus representing governmental structures in Iran and Russia.

4.3 Search Tasks

We used the same two exploratory search tasks [24], a simple and a complex exploratory search task, as follows:

Simple Politics: What governmental body or bodies are involved in the impeachment of the President of Iran and of Russia? (sample question)

Complex Politics: Imagine you are a high school student who is going to write an essay on the Political Systems of Iran and Russia. Knowing little about the presidents of these two countries, you wish to determine which president has more power. Find at least 3 arguments to justify your answer.

Simple History: As a result of which act were Upper and Lower Canada formed? (sample question)

Complex History: Imagine you are a high school student who is going to write an essay on the History of Canada. Knowing little about Canadian History, you wish to know which cities have served as a capital for Canada. You would also like to understand the reasons behind moving the capital from one city to another.

4.4 Study Design

Our study design was a $3 \times 2 \times 2$ [interface, topic, complexity] mixed design. For Knowledge graph and hierarchy, we leverage the data set of Sarrafzadeh et al. [39], available from the researchers in anonymized form. We add additional participants for our HKGs to yield our mixed design as follows.

For HKGs, each participant performed two different tasks, one simple and one complex. The topic area (history or politics) differed for each of these tasks. More formally, for these participants, our design was a 2×2 full factorial mixed design, with topic and complexity as within subjects factors and complexity to topic assignment as a between subject factor. We counter-balanced the order in which the tasks were assigned to the participants.

Alongside the HKG participants, leveraging data from Sarrafzadeh et al. [39] adds two additional levels of Interface (hierarchical tree or knowledge graph) as a between subject factor. Combining the data sets yield the $3 \times 2 \times 2$ mixed design [interface, topic, complexity] with interface as a between subjects factor, and topic and complexity as within subjects factors.

4.5 Participants

In total we analyze data from forty seven participants. Twenty six participants, thirteen female, used hierarchies and knowledge graphs, the control interfaces. An additional twenty-one participants (4 female) used HKGs, the experimental condition, as a between subjects factor. All participants use the Internet on a regular basis to search for information. Participants were aged between 18 and 45 years old (62% were between 20 and 29 years old). Participants received a \$15 incentive for their participation.

4.6 Procedure

After introducing the study, participants were presented with an experimental interface (populated with an unrelated data set), and were given time to familiarize themselves with the interface and data structure. Once participants had developed some comfort with the features of the interface (~ 3 minutes), participants completed a questionnaire assessing their familiarity with the topic used for the first task. They were then given the description of their task (see above), and were asked to complete the task using the interface (15 minutes per task). Participants completed a post-task questionnaire that evaluated the experience; we used questionnaires provided by

TREC-9 Interactive Searching track [4] modified to fit our experiment. The same process was repeated for the second task.

At the end of the second task, a semi-structured interview explored participants' experience using the interface. Interviews explored the conceptual usability of the visualization, the technical usability of the application and the efficacy of the interface for different types of search tasks. Feedback on competing interfaces was also collected from participants.

4.7 Data Collection

Alongside a mixed design of within subject and between subject factors, we perform a mixed methods analysis of both quantitative and qualitative data [13]. Data was captured as follows:

(a) The interface was instrumented with a logger which monitored movement on the computer screen and participants' interactions with the system. Interactions collected included node or edge clicks, snippets read, articles viewed, and time spent reading the articles. In HKGs, the transition between the layers and switches between the MiniMap and the knowledge graph were captured.

(b) Two assessors evaluated the quality of answers provided by the participants for each of the search tasks independently. Simple queries were rated as either correct or incorrect. Complex questions were rated on a scale. Scores for all queries were normalized to reflect a value in the range [0, 1]. Inter-assessor reliability was evaluated using Pearson coefficient and an overall value of 0.97 for simple queries and 0.94 for complex queries was found.

(c) We captured field notes during participant interactions, audio recorded all sessions, transcribed final interviews, and collected questionnaire data. This data was analyzed collectively using open coding to extract low-level themes and axial coding to identify thematic connections between elements. Coding was performed incrementally as each participant's data was collected, and saturation was found after coding qualitative data from field notes and transcripts for 15 of our 21 participants.

4.8 Hypotheses and Research Questions

Quantitative data allows us to test the following hypotheses:

- Hierarchical knowledge graphs result in fewer document views and less time spent reading documents than do hierarchical trees.
- Hierarchical knowledge graphs exhibit statistically similar behaviors to Knowledge Graphs.

Alongside hypothesis testing, our log data provides insight into whether hierarchies are used in hierarchical knowledge graphs and on whether task complexity affects the use of hierarchies. As well, to triangulate quantitative data, we leverage our qualitative data to compare and contrast the nature of the hierarchies between the tree interface and the hierarchical knowledge graphs and to understand whether the hierarchies provide similar affordances.

5 RESULTS

5.1 Quantitative Analysis

Scoring of participant responses by independent evaluators and log file analysis produced the quantitative measures in Table 1 for Hierarchical Knowledge Graphs (H. Graphs), Hierarchical Trees (H. Trees), and Knowledge Graphs (K. Graphs). Rows represent measures for Marks (MK), Nodes clicked (NK), Edges Clicked (EC), Document Views (V) and Document View Time (VT). We break each measurement out by two query levels, Simple and Complex, as described previously.

		H. Graphs	H. Trees	K. Graphs
Simple	MK	0.43 (0.21)	0.32 (0.20)	0.37 (0.14)
	NC	11.4 (8.6)	19.0 (10.04)	11.38 (9.4)
	EC	18.3 (8.9)	NA	27.15 (12.9)
	V	**2.38 (1.61)**	**6.08 (2.49)**	**2.38 (3.00)**
	VT	**145.6 (153.7)**	**1430.9 (2302.8)**	**211.6 (228.0)**
Complex	MK	0.62 (0.18)	0.57 (0.28)	0.58 (0.16)
	NC	13.38 (9.2)	20.09 (17.7)	26.23 (19.12)
	EC	23.09 (12.7)	NA	41.07 (19.4)
	V	**2.15 (2.13)**	**4.38 (2.24)**	**4.38 (2.24)**
	VT	**103.4 (97.6)**	**985.38 (1848.3)**	**78.76 (131.5)**

Table 1: Hierarchical (H.) Graphs vs. Hierarchical Trees and Knowledge (K.) Graphs: Mean (Standard Deviation) values for marks (MK - average independent evaluator scores), clicks on nodes (NC) and edges (EC), document views (V), and document view time (VT). Bolded dependent variables exhibited significant differences in post-hoc testing.

5.1.1 Hypotheses Testing. Multivariate analysis of variance with respect to interface (tree versus graph versus hierarchical graph), topic (history versus politics), and task (simple versus complex) for Marks (MK), Views (V), and View Time (VT) shows a statistically significant effect of interface ($F_{6,172} = 7.126, p < 0.001, \eta^2 = 0.2$) and task ($F_{3,86} = 12.22, p < 0.001, \eta^2 = 0.3$) on dependent variables. Post-hoc factor analysis using Tukey correction indicates that the tree interface exhibited statistically significantly higher numbers of document views than both hierarchical graphs and knowledge graphs. As well, the tree exhibited statistically longer reading times than hierarchical graphs ($p < 0.05$), but not than knowledge graphs (p = 0.064) in our analysis. Hierarchical graphs and knowledge graphs did not differ significantly in their effects on any dependent variables. Task significantly impacted the marks but no other variables.

Clicks are not directly comparable between H. Trees, H.Graphs, and K.Graphs, as edges are not clickable in hierarchies (NA value in Table 1). Performing pairwise comparison between H.Graphs and K.Graphs, our analysis showed no statistically significant effect on dependent variables ($F_{3,30} = 0.752, p > 0.5, \eta^2 = 0.70$), including node click and edge click behavior.

Given the above analyses, we reject both null hypotheses and conclude that our hypotheses are supported by our data set. Hierarchical Knowledge Graphs preserve the advantages of Knowledge graphs over hierarchical trees in both reading time and in document views. Focusing specifically on our hierarchical graph, we find that our hierarchical graph has statistically lower document views (61% fewer document views, on average) and time reading (90% less time reading documents) than does hierarchical trees and that its

[4] www-nlpir.nist.gov/projects/t9i/qforms.html

behavior is statistically indistinguishable from the prior observations of knowledge graph interfaces. Furthermore, the effect size measures, η^2, are significantly above the threshold (0.14) typically considered to be a large effect, lending support to these differences being sufficiently large to be meaningful. In summary, our quantitative results support our hypothesis that our hierarchical knowledge graphs fully preserve the quantitative advantages identified by Sarrafzadeh et al [39] for knowledge graphs over hierarchies.

5.1.2 Additional Quantitative Analysis. Given the statistically indistinguishable nature of HKGs and Knowledge Graphs, one question is if (and whether) intermediate hierarchical representations are used. It is possible that Hierarchical Knowledge Graphs are indistinguisable from Knowledge Graphs because users ignore the hierarchy and simply leverage the knowledge graph.

	GlobalView	MiniMap	DetailedView
Simple Task	27.03%	14.61%	58.0%
Complex Task	23.83%	17.24%	58.90%

Table 2: Percentage of Time spent on each of Global View, Minimap and Detailed View

To specifically explore this question, we looked at how much time users spent on each of the provided views in our HKG interface. Overall, our data indicated that participants took advantage of all three layers relatively similarly across both Simple and Complex tasks. Further, while the the time spent on detailed view dominates other views (58% for the simple task and 59% for the complex task), over 40% of time was spent on additional views in the hierarchy (Table 2). Looking specifically at how participants spent their time in different layers of the hierarchy (i.e. utilizing different views of the data) for different tasks we see that the time spent at the detailed view is similar for both levels of complexity. On the other hand, participants seem to spend less time in MiniMap than Global for the simple task (Pairwise t-tests with Tukey correction yields statistical significance, $p < 0.01$). For Complex task, however, time in Global versus mid-level are not statistically different ($p > 0.1$). Essentially, in the complex task, sensemaking is split between global and minimap views of the hierarchy more equitably, i.e., the minimap is particularly useful during our complex tasks.

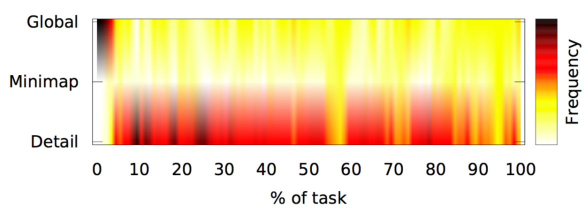

Figure 4: Heatmap visualizing the patterns of users navigating views in HKG for intervals of 1% of task length.

We also explored usage patterns of views. Figure 4 is a heatmap that visualizes use of different views for intervals of 1% of task length. Early in the task, we see frequent use of the global view. While difficult to see, MiniMap usage peaks just after the halfway point in the task, but there is no strong concentration of use. The hierarchy, and particularly the MiniMap visualization, seems to be used throughout the task.

5.2 Qualitative Analysis

The next question we explore involves participant perspectives on hierarchical knowledge graphs as a representation of search results. We were particularly interested in the overviews knowledge graphs provide for the information space and their contrast with Table-of-content-based hierarchies.

To address these questions, we performed open-coding of observations, transcripts, and questionnaire data. We coded incrementally, and saturation occurred after fifteen participants were coded. We coded all participants for completeness. Once open coding was complete, axial coding and thematic analysis was performed collaboratively by the researchers. We present three themes arising from our qualitative data analysis: Supporting Exploratory Search Tasks, Imposing a Structure versus Open Exploration and the Self-Orienting nature of HKGs.

5.2.1 Supporting Exploratory Search Tasks. As noted in our study design, we incorporate two exploratory information seeking tasks with different levels of complexity. In post-experiment interviews the participants were able to compare how different task complexities are supported by the assigned interface.

The hierarchical graph representation was found to provide more support for the Complex Task (i.e., more open ended and exploratory tasks such as essay writing or learning) versus Simple tasks (such as question answering and specific knowledge finding). This observation seems to be true for any multi-level structure which provides an overview and allows a gradual immersion into details: Finding a specific piece of information to satisfy a simple query is best done using a traditional search engine.

Looking specifically at HKGs and complex tasks, the overview allowed participants to identify the central concepts of a domain at a glance and the size of the circles indicates their prominence in the corresponding article. As many participants noted, 'relevance' or 'prominence' of a concept with respect to the main topic or the domain they are exploring is an important asset in Complex Search tasks. This qualitative observation may explain the more equitable use of the MiniMap representation for complex search tasks noted in our quantitative analysis. Complex tasks required synthesizing, rationalizing, and comparing, which seem to require more awareness of the entire data set.

This identification of central concepts was also linked to a perception of value of the MiniMap as a starting or entry point into the topic of the document being examined. Several participants articulated a belief that the overview provided by Central Concepts helped with "going from knowing nothing to having a plan", "learning terminology", "relevance, importance, or prominence", and "objectively learning about a domain". In particular, the *objective* nature of central concepts was cited by many participants as key to their utility.

As White and Roth [43] point out, exploratory search is motivated by complex information problems, poor understanding of terminology and information space structure, and often a 'desire to learn'. Vakkari [40] also argues "more support is needed in the initial stages of a task", when users have an unstructured mental model. Inspired by Kim [21], Sarrafzadeh et al. [39] found that hierarchical trees provide this benefit in unfamiliar domains. A strength of our design of hierarchical knowledge graphs is that it enables the user to engage in two alternative navigation paradigms. Users can exploit overview layers to explore the collection at a higher level followed by targeted immersion in the detailed view.

5.2.2 Imposing a Structure versus Open Exploration. While most participants were unanimous that the hierarchical representation imposes a [subjective] [rigid] structure onto the information space, their attitude towards this phenomenon varied. The level of domain knowledge and the complexity of the search tasks were found to be the major factors affecting their attitude.

When the searcher is dealing with a domain where he has limited knowledge, he is more open to accepting the structure that the representation imposes. Both hierarchical trees and hierarchical knowledge graphs incorporate imposed structures. Participants articulated a variety of advantages to structures: it was "easier to follow", "contained important aspects" that "simplified focus", and guided participants in "where to go" or "what steps to follow". With respect to hierarchical trees, some participants simply "trusted" the designer of the hierarchy (e.g. the author of an article) to be "logical" or "rational" in the way he broke down things. This was particularly true for participants with limited knowledge of a topic domain and replicates findings by Sarrafzadeh et al. [39] and Amadieu et al. [1] that low knowledge learners benefited from hierarchical structures in free recall performance and exhibited reduced disorientation.

In the case of higher domain knowledge, our participants were split in their preferences and attitudes. Some still trusted the logic behind the layout of a hierarchical trees and the fact that their knowledge of the domain can guide them to find what they want using this hierarchy. They trusted the designer to place items in close proximity to where the item should be. Other participants strongly opposed the rigid structure of a hierarchy, feeling it was "not the way I think", "based on the mindset of the author", or "did not match the domain structure".

One interesting perspective of the multi-layer graph representation which presents central concepts of a domain as an overview for each document is that it reflects the knowledge graph concepts. This reflection made it, for many participants, more flexible and exploratory, a window into the knowledge graph. Many participants commented on this phenomenon, noting it was "guiding but not imposting", "more open", "sparked interest" in the lower level structure, or was "visually appealing" and "fun".

5.2.3 Self-Orienting or Relative Positioning. One main advantage of a the Hierarchical Tree visualization in Sarrafzadeh et al. [39] was the explicit connections between nodes (categories or headings) in the representation. These edges help in two ways:

(1) At a glance, you can tell why a concept appeared in this overview, or in this domain. To whit, the hierarchical structure exists the way it does because of a human author's decision.
(2) The Path from the root to each of these nodes in the Tree Layout can provide useful information on where a concept is positioned relative to the topic.

Sarrafzadeh et al. [39] note that participants may perceive a domain to have a derivative/hierarchical structure or a multi-faceted structure. If salient relationships are viewed as derivative or hierarchical (e.g. 'is-a' relationships), then a tree can best capture this view of data, whereas if salient relationships are more heterogeneous and resist structure as a hierarchy, that disadvantages the hierarchies.

This is not the case in our MiniMap, where the connection between each of these main concepts and the main topic is unknown at first glance. Central concepts are simply extracted based on their high connectivity with other concepts within a specific document within a corpus. However, it is also true that it would be quite surprising if highly linked concepts were not, somehow, important components of any individual document. The more pervasively they link, the more they interconnect with other concepts, the more important it is to understand them and their relationship. In this way, HKGs become self-orienting for out participants.

6 LIMITATIONS

Any study has limitations. Because we leverage the research methodology and data sets of Sarrafzadeh et al. [39], we inherit the limitations of that study, including topic and implementation issues which may bias the study. Despite this, there is also a strength in replication: if interfaces are redesigned, data sets differ, and tasks are unique it becomes difficult to ensure a lack of confound in experimental design. We address this by preserving, to the limit possible, all aspects of a similar study within this space contrasting data structures.

Our mixed design of within and between subject factors is a particular strength to our study design. Because topic (history/politics) and task complexity are within-subject factors, they are controlled across participants. Because we are most interested in interface and it is a between subject factor, to observe statistical significance we need good separation of dependent variables between the two data sets, reducing the likelihood of a type-one error in our analysis.

7 CONCLUSION

The primary goal of our research was to explore whether we could combine benefits from both knowledge graphs and hierarchies into one data structure for visualizing search results. We note that our hierarchical graphs significantly reduce documents read and reading time as compared to hierarchical trees and perform on par with knowledge graphs. We also provide evidence that the hierarchy is used by participants via analysis of interaction logs.

Qualitative data from our participants does indicate that hierarchies grounded in tables-of-contents are more familiar, easier to follow, and more focused. This in turn helps users orient themselves in the data. The vetted nature of hierarchical tables-of-contents was also perceived to be an asset absent from our hierarchical knowledge graphs. The hierarchies in our knowledge graph were viewed slightly differently, as noted above, with a more quantitative perspective giving them a certain cachet with respect to the unbiased nature of topic selection.

A final issue to consider is whether any hierarchy might provide benefits. While it may, one advantage of the hierarchy in our HKGs is its tight connection to the entities contained in a knowledge graph and the ease of automatically extracting the hierarchy through thresholding. Another advantage is flexibility: while we currently leverage only three levels – corpus, central concept, and knowledge graph – it is easy to generalize the hierarchy to an arbitrary number of thresholds depending on the complexity of the domain. We do not generalize the hierarchy in this paper because, for a first experimental validation, there are a limited number of factors that can be assessed. However, future work can address more detailed inquiries into scalability to larger corpora, scalability to multi-level hierarchies, and contrasts with other hierarchies such as automatic clusters or user-specified facets.

In summary, we find that our hierarchical knowledge graphs preserve many of the previously observed advantages of traditional knowledge graphs, i.e. fewer document views and reduced reading time. Alongside this, hierarchical knowledge graphs introduce an effective hierarchical representation into knowledge graphs.

ACKNOWLEDGEMENTS

Funding for this research was provided by the Natural Science and Engineering Research Council of Canada (NSERC). Authors also thank Saeed Nejati for his contributions in implementing the interface.

REFERENCES

[1] Franck Amadieu, André Tricot, and Claudette Mariné. 2010. Interaction between prior knowledge and concept-map structure on hypertext comprehension, coherence of reading orders and disorientation. *Interacting with computers* 22, 2 (2010), 88–97.
[2] Anne Aula and Daniel M Russell. 2008. Complex and exploratory web search. In *Information Seeking Support Systems Workshop (ISSS 2008), Chapel Hill, NC, USA.*
[3] David Paul Ausubel, Joseph Donald Novak, Helen Hanesian, and others. 1968. Educational psychology: A cognitive view. (1968).
[4] Steven M Beitzel, Eric C Jensen, Abdur Chowdhury, David Grossman, and Ophir Frieder. 2004. Hourly analysis of a very large topically categorized web query log. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 321–328.
[5] Nicholas J Belkin. 1980. Anomalous states of knowledge as a basis for information-retrieval. *Canadian Journal of Information Science-Revue Canadienne Des Sciences De L Information* 5, MAY (1980), 133–143.
[6] Andrei Broder. 2002. A taxonomy of web search. In *ACM Sigir forum*, Vol. 36. ACM, 3–10.
[7] Katriina Bystr̀Om. 2002. Information and information sources in tasks of varying complexity. *Journal of the American Society for information Science and Technology* 53, 7 (2002), 581–591.
[8] Donald J Campbell. 1988. Task complexity: A review and analysis. *Academy of management review* 13, 1 (1988), 40–52.
[9] Robert Capra, Gary Marchionini, Jung Sun Oh, Fred Stutzman, and Yan Zhang. 2007. Effects of structure and interaction style on distinct search tasks. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries.* ACM, 442–451.
[10] Mary Carnot, Paul Feltovich, Robert Hoffman, Joan Feltovich, and Joseph Novak. 2003. A summary of literature pertaining to the use of concept mapping techniques and technologies for education and performance support. (2003).
[11] Hsinchun Chen, Andrea L Houston, Robin R Sewell, and Bruce R Schatz. 1998. Internet browsing and searching: User evaluation of category map and concept space techniques. *Journal of the American Society for Information Science, Special Issue on AI Techniques for Emerging Information Systems Applications* (1998).
[12] Andy Cockburn and Steve Jones. 1996. Which way now? Analysing and easing inadequacies in WWW navigation. *International Journal of Human-Computer Studies* 45, 1 (1996), 105–129.
[13] John W Creswell. 2013. *Research design: Qualitative, quantitative, and mixed methods approaches.* Sage publications.
[14] Brenda Dervin. 1998. Sense-making theory and practice: an overview of user interests in knowledge seeking and use. *Journal of knowledge management* 2, 2 (1998), 36–46.
[15] Marti Hearst. 2009. *Search user interfaces.* Cambridge University Press.
[16] Marti A Hearst. 1999. The use of categories and clusters for organizing retrieval results. In *Natural language information retrieval.* Springer, 333–374.
[17] Marti A Hearst. 2006. Clustering versus faceted categories for information exploration. *CACM* 49, 4 (2006), 59–61.
[18] P James. 1991. Knowledge graphs. *Order* 501 (1991), 6439.
[19] Bernard J Jansen and Amanda Spink. 2003. An Analysis of Web Documents Retrieved and Viewed.. In *International Conference on Internet Computing.* Citeseer, 65–69.
[20] Sussane Jul and George W Furnas. 1997. Navigation in electronic worlds: a CHI 97 workshop. *SIGCHI bulletin* 29 (1997), 44–49.
[21] Kyung-Sun Kim. 1999. Searching the Web: Effects of Problem Solving Style on Information-Seeking Behavior. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Vol. 1999. 1541–1542.
[22] Anita Komlodi, Gary Marchionini, and Dagobert Soergel. 2007. Search history support for finding and using information: User interface design recommendations from a user study. *Information processing & management* 43, 1 (2007), 10–29.
[23] Bill Kules and Robert Capra. 2008. Creating exploratory tasks for a faceted search interface. *Proc. of HCIR 2008* (2008), 18–21.
[24] Gary Marchionini. 2006. Exploratory search: from finding to understanding. *CACM* 49, 4 (2006), 41–46.
[25] Gary Marchionini and Ben Shneiderman. 1988. Finding facts vs. browsing knowledge in hypertext systems. *Computer* 21, 1 (1988), 70–80.
[26] Gary Marchionini and Ben Shneiderman. 1993. 3.1 Finding facts vs. browsing knowledge in hypertext systems. *Sparks of innovation in human-computer interaction* (1993), 103.
[27] Kent L Norman and John P Chin. 1988. The effect of tree structure on search in a hierarchical menu selection system. *Behaviour & Information Technology* 7, 1 (1988), 51–65.
[28] Joseph D Novak. 1990. Concept mapping: A useful tool for science education. *Journal of research in science teaching* 27, 10 (1990), 937–949.
[29] Joseph D Novak and Alberto J Cañas. 2008. The theory underlying concept maps and how to construct and use them. *FIHM Fl* 284 (2008).
[30] Laura R Novick and Sean M Hurley. 2001. To matrix, network, or hierarchy: That is the question. *Cognitive Psychology* 42, 2 (2001), 158–216.
[31] Christopher Olston and Ed H Chi. 2003. ScentTrails: Integrating browsing and searching on the Web. *ACM Transactions on Computer-Human Interaction (TOCHI)* 10, 3 (2003), 177–197.
[32] Peter Pirolli, Stuart K Card, and Mija M Van Der Wege. 2000. The effect of information scent on searching information: visualizations of large tree structures. In *Proceedings of the working conference on Advanced visual interfaces.* ACM, 161–172.
[33] Wanda Pratt, Marti A Hearst, and Lawrence M Fagan. 1999. A Knowledge-Based Approach to Organizing Retrieved Documents.. In *AAAI/IAAI.* 80–85.
[34] Kerry Rodden, Wojciech Basalaj, David Sinclair, and Kenneth Wood. 2001. Does organisation by similarity assist image browsing?. In *Proceedings of the SIGCHI conference on Human factors in computing systems.* ACM, 190–197.
[35] Daniel E Rose and Danny Levinson. 2004. Understanding user goals in web search. In *Proceedings of the 13th international conference on World Wide Web.* ACM, 13–19.
[36] Daniel Russell, Mark Stefik, Peter Pirolli, and Stuart Card. 1993. The cost structure of sensemaking. In *Proc. of INTERACT'93 and CHI'93.* ACM, 269–276.
[37] Bahareh Sarrafzadeh, Rakesh Guttikonda, Kaheer Suleman, Jack Thomas, and Olga Vechtomova. March 2013. *Automatic discovery of related concepts.* Technical Report.
[38] Bahareh Sarrafzadeh, Olga Vechtomova, and Vlado Jokic. 2014. Exploring knowledge graphs for exploratory search. In *Proc. of IIiX.* ACM, 135–144.
[39] Bahareh Sarrafzadeh, Alexandra Vtyurina, Edward Lank, and Olga Vechtomova. 2016. Knowledge Graphs versus Hierarchies: An Analysis of User Behaviours and Perspectives in Information Seeking. In *Proc. of the 2016 ACM on Conference on Human Information Interaction and Retrieval.* ACM, 91–100.
[40] Pertti Vakkari. 2000. Relevance and contributing information types of searched documents in task performance. In *Proc. of SIGIR.* ACM, 2–9.
[41] Frank Van Ham and Adam Perer. 2009. "Search, Show Context, Expand on Demand": Supporting Large Graph Exploration with Degree-of-Interest. *Visualization and Computer Graphics, IEEE Transactions on* 15, 6 (2009), 953–960.
[42] Charles K West, James A Farmer, and Phillip M Wolff. 1991. *Instructional design: Implications from cognitive science.* Prentice Hall Englewood Cliffs, NJ.
[43] Ryen W White and Resa A Roth. 2009. Exploratory search: Beyond the query-response paradigm. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 1, 1 (2009), 1–98.
[44] Barbara M Wildemuth and Luanne Freund. 2012. Assigning search tasks designed to elicit exploratory search behaviors. In *Proc. of HCIR 2012.* ACM, 4.
[45] Max L Wilson and others. 2009. The importance of conveying inter-facet relationships for making sense of unfamiliar domains. (2009).
[46] Max L Wilson, Bill Kules, Ben Shneiderman, and others. 2010. From keyword search to exploration: Designing future search interfaces for the web. *Foundations and Trends in Web Science* 2, 1 (2010), 1–97.
[47] Yi-fang Brook Wu, Latha Shankar, and Xin Chen. 2003. Finding more useful information faster from web search results. In *Proceedings of the twelfth international conference on Information and knowledge management.* ACM, 568–571.
[48] Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. 2003. Faceted metadata for image search and browsing. In *Proc. of CHI.* ACM, 401–408.