Tools for Supporting Online Sensemaking under Uncertainty and Evolving User Interests

Joseph Chee Chang

May 2019

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Aniket Kittur, Carnegie Mellon University, Chair
Jeffrey Bigham, Carnegie Mellon University
Adam Perer, Carnegie Mellon University
David Karger, Massachusetts Institute of Technology

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Abstract

Whether planning trips or researching medical conditions, exploring and consuming information online has become how most people make sense of the world today. Having access to this rich and diverse information can lead to better informed decisions [46], but the underlying cost of attention and cognitive capacity must also be considered [150, 154]. While modern search engines can answer factual questions in split seconds, exploratory searchers who are uncertain about their goals need to explore and learn from data in order to make sense of the available options and how to compare them. This process can be costly for users, requiring them learn the space of information through exploring many webpages, discover key aspects that were important to them personally, refine their goals throughout the process, and keep track of potential options and evidence scattered across many webpages.

This thesis explore ways to build tools that support uncertainty and evolving user interests at different stages of this sensemaking process. I will first describe my past work from data-driven and crowdsourced approaches that synthesizes overviews from multiple webpages for ad-hoc queries (Chapter 3) to user-driven approaches involving a search interface allowing users to express and evolve their interests and generated personalized visual explanations (Chapter 4), and systems that support saving information under uncertainty (Chapter 5) and foraging across multiple webpages in the browser (Chapter 6). Building on a prototype browser extension described in Chapter 6, my proposed work will expand my research two areas of sensemaking in the browser – managing information sources and structuring information collected from them.

In addition to conducting user testing and controlled lab studies throughout development, I will also attempt to publically release the browser extension to test my approaches at scale and with real-life sensemaking tasks. This thesis will contribute to HCI by generating new frameworks and interaction techniques for supporting evolving goals during online sensemaking at different stages, and the implementation and evaluation of several systems that embodied such frameworks.

Contents

List of Figures

List of Tables

Chapter 1: Introduction and Thesis Statement

Whether planning a trip, researching a medical condition, or developing a new skill, consuming and foraging information online through exploratory search has become how people make sense of the world. People now have instant access to an enormous online bazaar of information produced by experts and novices with different personal preferences, backgrounds, and assumptions. While this rich repository of diverse perspectives has the potentials to empower consumers and learners to explore and understand available options thoroughly and make better decisions [46], the underlying cost of our attention and cognitive capacity must also be considered [154] — the seemingly infinite number of options and evidence scattered across numerous information sources can be overwhelming to comprehend, prohibiting us to benefit fully from the richness of Web content, and potentially leading to indecision [41, 150]. Exploratory searches are uncertain because users do not have a clear goal and do now know what potential options were available and what signals and criteria to base their decisions on [115]. Major challenges for individuals include: 1) exploring and learning the space of information to identify potential options, 2) discovering the important aspects and criteria from data to develop a personal framework for comparing options while 3) managing many the information sources in the browser, and keeping track of options and evidence scattered across them.

Consider purchasing a desk lamp at an office supply store versus online, and the differences in scale of available options and evidence. A physical store typically offers a handful of choices that can be narrowed down to make a decision using a few simple criteria such as price, aesthetic, and brightness. Contrarily, when making the same purchase online, Amazon lists over 4,000 different options. Besides more options, online reviews provide in-depth information and experiences that were often unavailable at physical stores. Amazon lists up to thousands of reviews for each option; Google returns hundreds of "best desk lamps" listicles; and Reddit[1] lists thousands of discussions on desk lamps. While having access to more options and evidence can potentially lead to better decisions, the process of exploring and foraging from this data can be costly to the end users [134, 135]. First, to gain an overview of the space users have to examine many sources to identify a smaller set of potential options. Then, to develop a framework for comparison they have to learn and discover the different aspects and criteria important to them. Finally, to characterize the different options based on relevant aspects they have to collect all evidence scattered across different webpages [139]. These challenges are prevalent to many other online decision-making scenarios beyond shopping. For example, novices figuring out "how to grow better tomatoes" from expert tutorials and enthusiast forums; citizens learning how a proposed bill might affect them personally from the news and online debates; and researchers conducting literature surveys in unfamiliar fields. The lack of support to efficiently overview, manage, and forage from multiple online sources makes it difficult for users to understand the space of information and keep track of what they are interested in and why.

Evidence for this need can be seen in a decade of research in machine learning that ranks in-

[1] `http://reddit.com`

dividual reviews based on general usefulness [59, 93, 101] and review summarization [80, 110, 113, 184]. Anecdotal evidence can also be seen in the rise of aggregation-based websites such as Meta-critics[2] and Rotten Tomatoes[3] that calculate a weighted average rating from multiple other review sources. Yet prior work has shown that reading qualitative reviews is still a crucial process of consumer decision making even when average ratings were available [58, 127]. In response to the need for in-depth understanding in online decision making, meta-analysis websites such as Wirecutter[4] compiles comprehensive meta-analysis of evidence and opinions scattered across multiple sources for common items such as "The Best Rechargeable Batteries for Most People" to more technical items such as "The Best DLSR Cameras for Most People." While these aggregation and meta-analysis based approaches offer great starting points for exploratory searchers to overview common options and evidence to discover useful aspects and criteria, they can have difficulties scaling to ad-hoc exploratory search scenarios, especially when well-defined options and structured reviews were not available (e.g., different approaches for growing better tomatoes) or when readers have a wide variety of personal preferences (e.g., food [60] or fashion choices).

Identifying Options across Webpages for Overview

In the first part of this dissertation (Chapter 3), I explore ways to generate overviews from multiple online sources for ad-hoc exploratory search topics. For this, I built a novel system that identify potential options scattered across webpages using both crowdsourcing and machine learning techniques. Due to limited prior knowledge, exploratory searchers typically start with general and tentative queries [171, 173, 175]. This allowed for maximizing recall for exploring the space of information to identify important subtopics and aspects for further investigation. Due to the large number of available sources and options, summarizing information scattered across search results has the potential of better orienting searchers in an unfamiliar domain and identify interesting options or subtopics to investigate further. Developments in crowdsourced information structuring [42, 153] introduced new opportunities for identifying options in Web content for generating overviews for ad-hoc exploratory search scenarios. In this dissertation, I built upon these techniques, and explored ways to identify options or key subtopics across webpages and provide overviews for ad-hoc scenarios in the following ways:

- Developed a system that combined crowdsourcing and machine learning to discover useful options for a given exploratory search topic by clustering information gathered from multiple webpages into useful and coherent subtopics (Chapter 3).
- Synthesized the generated structures into overview articles and comparing them against top search results (Section 3.4).

The intuition behind this approach is that crowdsourcing can provide on-demand human computation that can generate overviews for ad-hoc online sensemaking tasks, and combining it with machine learning techniques allowed the system to generate coherent structures with lowered costs.

[2] `https://www.metacritic.com/`, a CBS company.
[3] `https://www.rottentomatoes.com/`, a Comcast/NBC company.
[4] `http://wirecutter.com`, a New York Times company.

Searching and Foraging and Under Evolving Interests

While data-driven summarization can provide general overviews to better orient users in the initial stages, the later stages of exploration can become increasingly personal, dynamic, and opportunistic [115] – as users develop a personalized framework for comparison by discovering new aspects and options, they iteratively refine their search goals, which potentially invalidates prior choices and opens new research directions [134, 135]. This intense sensemaking process under uncertainty and evolving goals, combined with the need to manage and forage from multiple webpages, can also be costly for users, require constant switching between browser tabs and note-taking software for cross-referencing and foraging. In the second part of this dissertation, I investigated user-driven approaches to better support the above activities, such as capturing user interests with a visual search interface or supporting entity-centric foraging across webpages in the browser.

While modern search engines and web browsers provide excellent efficiency for retrieving and displaying large numbers of sources, little support is available for expressing and maintaining the evolving goals and interests of the users, presenting information based on users' interests, and supporting foraging under uncertainty and from multiple webpages. For example, simple query terms rarely capture all the nuanced preferences of users, and browsers make no connections between concepts mentioned in different tabs and in users' notes. As users forage from more sources and consider more options, their workspaces could become increasingly cluttered, making the process even more difficult. In the second part of this dissertation (Chapters 4 to 6), I explored new ways to support uncertainty and evolving user goals and interests during an exploratory search during searching, saving information, and foraging across multiple webpages in the browser:

- Developed a review search system that allowed searchers to externalize and evolve their latent interests as structured queries, and generated personalized interfaces with visual explanations that promotes transparency and enables deeper exploration (Chapter 4).
- Designed a novel interaction that intentionally incorporates uncertainty when selecting and capturing information from webpages to lower users' cognitive and physical efforts (Chapter 5).
- Developed a browser extension that enabled the browser to identify common entities mentioned across open webpages for an exploratory search task (such as mentions of travel destinations in a trip planning task), allow users to efficiently foraging evidence about entity across multiple webpages (Chapter 6).

Building on my prior work and the prototype system presented in Chapter 6, I propose to further explore ways to support the next steps in online sensemaking: managing multiple information sources in the browser during foraging and organizing information collected from them. Specifically, I plan to first explore how information workers utilize browser tabs during complex sensemaking tasks and issues they encountered, and extend the current system to address these issues. Detailed study plans are described in Chapter 7.

Thesis Statement and Overview

When conducting exploratory searches, users need to explore the space of information and learn from data to better understand their choices and refine their search goals. However, useful

information is often scattered across many sources and interspersed with information not of interests to the users. This introduces many challenges to the users as described in previous sections. To investigate methods that can better address the uncertain and distributed nature of online sensemaking, I consider the following as the thesis of this dissertation:

> **By identifying the latent and nuanced structures of options and evidence scattered across multiple online sources and supporting the uncertain and evolving nature of end-user goals during exploratory search tasks, we can provide better sensemaking support in different stages of the process, allowing users to gain a broader and deeper understanding of the space of information to make informed decisions with lowered efforts.**

Specifically, my dissertation work explores this thesis in three contexts: 1) **identifying Options across Webpages for Overview**: Synthesize information scattered across search results into overview articles (Chapter 3) by identifying useful options for an exploratory search task using crowdsourcing and machine learning. 2) **Searching and Foraging and Under Evolving Interests**: Allow users to express and iteratively refine rich expressions of their nuanced personal interests and criteria during foraging and visualize information based on user interests (Chapter 4), an interaction technique that supports saving information under uncertainty with lowered efforts (Chapter 5), and supporting foraging across webpages in the browser (Chapter 6). 3) Finally, building on insights from these prior work and informed by user studies and preliminary interviews, my proposed work will focus on the two areas of the online sensemaking process — **managing multiple information sources in the browser during foraging and structuring the collected information** (Chapter 7).

Chapter 2: Background

Exploratory Search Interfaces

Summarizing search results has been an area of high research interests. Early threads of research include search results clustering [74, 185, 186] and faceted search interfaces [71]. While search results clustering allowed users to effectively identify irrelevant sources caused by ambiguous queries (i.e., Apple), automatically generated structures can be incoherent and difficult to comprehend by users [32, 45, 73]. Conversely, faceted search interfaces present pre-compiled metadata structures that allowed users to explore different subsets of options with different criteria (such as products of specific brands and within a price range). While prior work has shown faceted search interfaces to be beneficial for exploratory searchers [72, 181], they cannot easily scale to data where structured metadata was not available such as the Web [166] or scenarios where options do not have well-defined attributes (such as ways to grow better tomatoes.) More fundamentally, many early work on search results interfaces focused on providing overview of different webpages instead of their content. While this allowed for efficient navigation to sources, searchers still have to go through the relevant or partially relevant content to learn about their options and evidence. This dissertation instead focuses on identifying potential options of from content of webpages, and presents overviews of options and evidence. In chapter 3, I explored combining crowdsourcing and machine learning to structure information in learning scenarios such as figuring out "how to grow better tomatoes" or "what does a planet need to support life."

Structuring Information with Crowdsourcing

Human computation presents new opportunities to address issues with automatically generated information structures, and provide coherent structures for overviews for exploratory searchers. For example, Cascade [22, 42] generated hierarchical categories from online forum discussions, but suffers from categories at the same level having varying specificity due to the limited context of each crowdworkers. Crowd Synthesis [5] showed that simply showing more items to each crowdworker can lead to significant improvements, suggesting global context is a key element for crowd structuring. Fundamentally, most prior systems provide context by showing a small sample of items, hoping that they capture the distribution of information in the larger dataset. A complementary set of approaches has focused on the scaling through computation, applying approaches such as partial clustering [182], learning similarity metrics [162], or matrix completion [183]. While these have shown to be powerful on simple information such as visual patterns or colors using large numbers of split-second judgments, structuring complex exploratory search information can be difficult without providing novice crowdworkers with richer context or opportunities to learn from data. In chapter 3, I propose an alternative approach that builds up workers' mental models by allowing them to actively request for more context, identify discriminative keywords, and search the dataset for similar items, taking advantage of people's capacity of information foraging [134]. The resulting structures were found to be more coherent than a state-of-the-art system and at a lowered monetary cost.

Online Information Foraging

When reading the individual pages, users in exploratory search tasks often need to take notes or save information from different webpages. Due to its ubiquity and importance, researchers have also been developing systems that can better support saving and organizing information to facilitate learning and exploratory search tasks [31, 76, 97, 149, 164, 168]. However, researchers have also identified common issues users are faced with while organizing collected information. Karger et al. pointed to the fact that most information management systems failed to provide an effective structure for its users due to the long-tail distribution of information types that people have [91]. Further, Kittur et al. found that in early stages of exploration, often searchers themselves do not have enough context to come up with effective structures [97]. These studies revealed systems that force schematization too early in the exploration process stages to can be detrimental to their users. In a study closely related to my proposed work, the NoteToSelf system added a sidebar to the browser for saving free-form notes that persist as users browse different webpages [168]. However, they found that while participants created many notes during browsing, the strategies they used for re-finding and re-accessing previously saved notes may not scale to scenarios where the user need to save many notes and that users also rarely deleted unused notes. This suggests that even though allowing users to externalize their arbitrary structures can be beneficial, but the high cost of recalling and re-finding previously saved information can still be prohibitive. In my proposed work, I plan to address the information reuse problem using an entity-centric approach to build a note-taking browser extension. As the user opened a webpage, the system analyzes the content and resurfaces previously saved notes that were associated with entities also on the current webpage.

Chapter 3: Alloy

Structuring Information with Crowds and Computation for Overview

This work was previously published in ACM SIGCHI 2016 [32, 66] and has been adapted for this document.

This chapter describe the first of the two approaches explored in this dissertation for identifying potential options from Web data to generate overviews and support ad-hoc exploratory search scenarios. Crowdsourced clustering approaches present a promising way to harness deep semantic knowledge of human computation for identifying coherent categories and clustering complex information. However, existing approaches have difficulties supporting the global context needed for workers to generate meaningful categories, and are costly because all items require human judgments. We introduce Alloy, a hybrid approach that combines the richness of human judgments with the power of machine algorithms. Alloy supports greater global context through a new "*sample and search*" crowd pattern which changes the crowd's task from classifying a fixed subset of items to actively sampling and querying the entire dataset. It also improves efficiency through a two phase process in which crowds provide examples to help a machine cluster the head of the distribution, then classify low-confidence examples in the tail. To accomplish this, Alloy introduces a modular "*cast and gather*" approach which leverages a machine learning backbone to stitch together different types of judgment tasks. In an application-oriented evaluation, Alloy clustered were further synthesized into comprehensive overview articles using a workflow described in [66]. Results show that Alloy structures can lead to coherent and comprehensive overviews that out performed top Google search results published by experts in scenarios where there are a lack of authoritative sources.

3.1 Introduction

Clustering, or pulling out the patterns or themes among documents, is a fundamental way of organizing information and is widely applicable to contexts ranging from web search (clustering pages) to academic research (clustering articles) to consumer decision making (clustering product reviews) [85]. For example, a researcher may try to pull out the key research topics in a field for a literature review, or a Wikipedia editor may try to understand the common topics of discussion about a page in order to avoid or address previous conflicts. Doing so involves complex cognitive processing requiring an understanding of how concepts are related to each other and learning the meaningful differences among them [11, 100, 121].

Computational tools such as machine learning have made great strides in automating the clustering process [19, 28, 44]. However, a lack of semantic understanding to recognize the important differences between clusters leaves the difficult task of identifying meaningful concepts to the human analyst [45]. This reflects an inherent advantage for humans over machines for the complex problem of understanding unstructured data beyond merely measuring surface similarity, and a corresponding opportunity for research in combining human and computational judgments to process complex information [53, 81, 102].

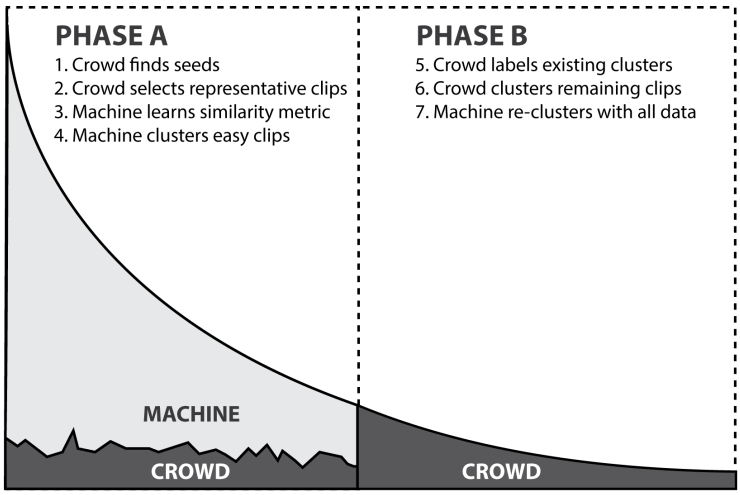

Figure 3.1: A conceptual overview of the system. In the first phase, crowd workers identify seed clips to train a machine learning model, which is used to classify the "head" of the distribution. In the second phase, crowd workers classify the more difficult items in the "tail". A machine learning backbone provides a consistent way to connect worker judgments in different phases.

One such promising avenue of research harnesses the power of crowds to identify categories and cluster rich textual data. Crowdsourcing approaches such as Cascade, Deluge, and Crowd Synthesis [5, 22, 42] have demonstrated the power of splitting up rich, complex datasets into small chunks which can be distributed across many human coders. However, all of these approaches must grapple with a fundamental problem: since each human coder is seeing only a small part of the whole dataset, a lack of global context can lead to incoherent results. For example, if the items sampled are too similar, the worker might create overly fine-grained clusters. On the other hand, if the items sampled are too dissimilar, the worker might create overly broad clusters. Clusters found in many worker segmentation sets may give rise to redundant clusters, while clusters whose items are sparsely split among segmentation sets may never be realized at all. As an example, [5] cite redundancies in Cascade's top level clusters having both "green" and "seafoam green", "blue" and "aqua", as well as the encompassing category of "pastels". While Crowd Synthesis used an iterative approach to address these redundancy problems, it trades this off with lowered robustness as issues with early workers' categories can cascade throughout subsequent workers' judgments. This suggests the design space of approaches for crowd clustering may be being critically limited by the assumption of splitting up the dataset into small, fixed pieces that prevent workers from gaining a more global context.

Another challenge with current crowd clustering approaches is that using human judgments to label each piece of data is costly and inefficient. Deluge addresses some issues with efficiency, improving on Cascade by reducing the number of human judgments elicited as the rate of new category generation slows [42]. However, these crowd clustering algorithms still require human judgments for every item, which is costly. In the real world data often follows a long-tailed distribution in which much of the data is captured by a small number of categories in the head of the distribution [172]. For such data in which many items in the head of the distribution are likely to be highly similar, once humans have identified the meaningful categories and representative examples it would be more efficient if a machine could classify the remaining items in those

categories. A danger with such an approach is that the sparse categories in the tail of the distribution with few examples may be difficult to train a machine to recognize, and so human judgments may have another important role in "cleaning up" low frequency categories.

This chapter describes Alloy, a hybrid approach to text clustering that combines the richness of human semantic judgments with the power of machine algorithms. Alloy improves on previous crowd clustering approaches in two ways. First, it supports better global context through a new "*sample and search*" crowd pattern which changes the crowd's task from classifying a fixed subset of items to actively sampling and querying the entire dataset. Second, it improves efficiency using initial crowd judgments to help a machine learning algorithm cluster high-confidence unlabeled items in the head of the distribution (prominent categories), and then uses later crowd judgments to improve the quality of machine clustering by covering the tail of the distribution (edge cases and smaller categories). To achieve these benefits, Alloy introduces a novel modular approach we call "*cast and gather*" which employs a machine learning backbone to stitch together different types of crowd judgment tasks. While we provide a particular instantiation of the cast and gather approach here (with a hierarchical clustering backbone which gathers three types of crowd tasks, or "casts"), the general framework for modularizing multiple types of human judgments with a common machine-based backbone may inspire application to other contexts as well.

3.1.1 Related Work

Document and short text classification are well researched topics in natural language processing and machine learning. With enough labeled training data, state-of-the-art algorithms can often produce good results that are useful in real world applications. Yet building such systems often requires expert analysis of specific datasets both to manually design an organization scheme and to manually label a large set of documents as training data. Unsupervised approaches, or clustering, aim to discover structures on-demand and without expert preparation [70, 86, 157]. While these data mining approaches may discover dimensions (features) that provide a good separation of the dataset, the inferred categories can be difficult for a human to interpret, and many of them may not capture the most meaningful or useful structure in a domain due to high dimensionality or sparseness in the word vector space [11, 100]. To deal with these issues, researchers have explored ways to automatically discover topical keywords that can help identify useful categories in unstructured data such as TF-IDF, latent semanic analysis, and latent Dirichlet allocation [19, 47, 89, 114]. However, even with these improvements, automatic methods often still perform poorly, especially when the number of document is small, the lengths of the documents are short, or when the information is sparse.

More recently, researchers have begun to use crowds to organize datasets without predefined categories. Cascade [42] attempts to address abstraction and sampling problems by first having multiple workers generate categories for each item and then later having workers choose between them. By providing limited context to each worker (8 items or 1 item with 5 categories), it suffers from categories that can have varying levels of specificity. As a follow up study, Deluge [22] produces comparable results, but with significantly lower cost by optimizing its workflow using machine algorithms. In another line of research, Crowd Synthesis [5] showed that providing more context by simply showing more items can lead to significant better categories, suggesting that global context is one of the key elements for crowd clustering algorithms. In general, most current systems provide context by showing a small sample of items, hoping that they captures

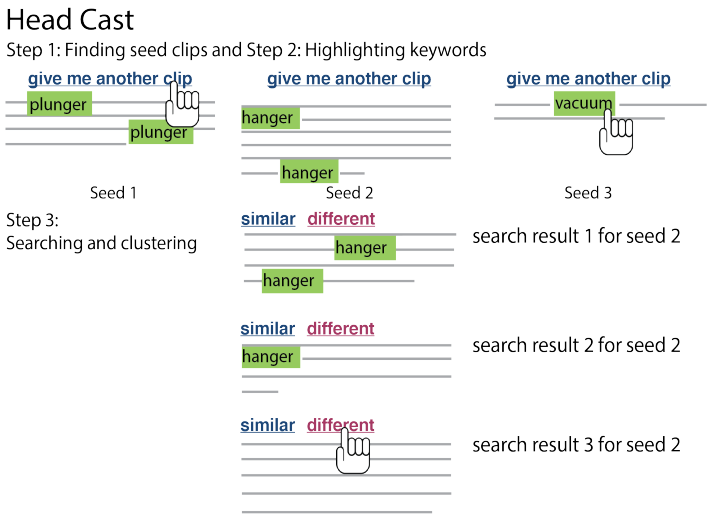

Figure 3.2: The interface and steps of the Head Cast HIT.

the distribution of information in the larger dataset. We propose an alternative approach that builds up workers' mental models by asking them to repeatedly sample for new items, identify discriminative keywords, and search the dataset for similar items, taking advantage of people's capacity of information foraging [134].

A complementary set of approaches to crowd clustering research has focused on addressing the scaling problem through computation, applying approaches such as partial clustering [182], learning similarity metrics through triad-wise comparisons [162], or using matrix completion to reduce the number of labels needed from workers [183]. While these approaches have shown to be powerful on simple information such as images or travel tips, synthesizing more complex information can be difficult without providing novice crowdworkers with richer context or opportunities to deeply process the data.

3.2 System Design

The Alloy system clusters a collection of clips, or short text descriptions (Figure 3.3), using a machine learning backbone that gathers various judgments from human workers. In our terminology, each human task is a "Cast" for human judgements which are then "Gathered" together with the machine learning backbone. Alloy enables Casts (here, crowdworker tasks) of different types and in different orders to be fused together by calling a Gather after each one. In each Cast stages, arbitrary number of workers can be hired for better robustness or lower cost. In this chapter, we present three types of Casts with different purposes as well as one type of Gather. At a high level, the "Head Cast" is aimed at finding common categories in the head of the distribution, while the "Tail Cast" is aimed at classifying categories in the tail of the distribution for which machine clustering has low confidence. The "Merge Cast" aims to clean up existing categories by combining highly similar categories. We also describe a Gather Backbone that fuses the judgements from multiple crowdworkers, and connects multiple casts to form complete workflows. For ease of exposition we introduce each component in the context of a typical workflow: the Head Cast, the Gather, the Merge Cast, and the Tail Cast.

3.2.1 The Head Cast

The Head Cast aims to identify salient keywords to uncover the most common categories in the head of the distribution. Doing so involves challenges in providing workers sufficient context to know what a good category is, and also in how to structure their work process in order to train a machine learning algorithm to take over the classification of categories based on human-identified seeds and keywords. Previous studies show that presenting multiple items from a collection can help provide context to human workers [50], increasing the likelihood of obtaining better clusters. However, it can be difficult to determine how much context is sufficient and how to produce a good sample that captures the distribution of information of the whole dataset. Therefore, we introduce a new crowd-pattern we call "*sample and search*" for providing global context through active sampling and searching with keywords. We ask crowdworkers to identify coherent categories by presenting with four random items, but allowing them to replace each item by random sampling from the entire dataset until they are confident that the items will be in different categories in the final output. This requirement gives them the motivation to build up better global understanding of the dataset through repeated sampling. After obtaining the four seed items, we ask crowdworkers to identify keywords in each clips to search for related items in the dataset. This process takes advantage of people's capacity of finding new information [134]. To create a familiar experience, we allow the workers to freely change their query terms and update the search results in real time. This way they can refine their searches based on the results, the same way as when conducting online information foraging tasks [88]. As shown in Figure 3.2, the Head Cast HIT interface consists of three steps:

1. **Finding seeds**: Four random seed clips are presented to each crowdworker. Over each clip, there is a button that allows them to replace the clip with another random clip from the dataset. They are then asked to replace any clips that are too similar to the other seed clips. The workers repeatedly replace the seed clips until the four clips at hand belong to four different answer categories.
2. **Highlighting keywords**: The crowdworker is then instructed to highlight one to three unique keywords from each of the four seed clips that best identify their topics.
3. **Search and label**: For each seed clip, we automatically search for similar clips from the entire corpus based on the highlighted keywords and TF-IDF cosine similarity. The crowdworker is asked to label the top nine search results as *similar* to or *different* from their seed clips.

In Step 1, the crowdworkers need some understanding of the global context before they can confidently judge that the seeds belong to different categories in the final output. Previous work usually address this problem by presenting multiple items to each crowdworker, in hopes of sampling both similar and dissimilar items to give some sense of the global context. In reality it could be difficult to judge how many items is sufficient for different datasets, and overly small size could lead to bad samples that are unrepresentative of the global distribution. We took a different approach by presenting fewer items at first, but allowing workers to replace the seeds with random clips from the dataset. This provide them both the mechanism and motivation to explore the dataset until they have enough context to find good seed clips.

The intuition behind Step 2 is that people are already familiar with picking out good keywords for searching documents related to a concept via their online information seeking experiences. In addition, requiring them to highlight unique keywords in the seeds first, further ensures that

Tomato seedlings will need either strong, direct sunlight or 14-18 hours under grow lights. Place the young plants only a couple of inches from florescent grow lights. Plant your tomatoes outside in the sunniest part of your vegetable plot.

In its astrobiology roadmap, NASA has defined the principal habitability criteria as "extended regions of liquid water, conditions favourable for the assembly of complex organic molecules, and energy sources to sustain metabolism

Figure 3.3: Example clips from two datasets with crowd keywords.

they are familiar with the concepts in the seed clips, before they search for similar items. In Step 3, the crowdworkers can still change and refine their highlights from Step 2, and the system will refresh the search results in realtime. This gives the crowdworkers both the motivation and mechanism to extract better keywords that lead to better search results to label. In Figure 3.3, we show two example clips from the datasets collected using the two questions: *How do I get my tomato plants to produce more tomatoes?* and *What does a planet need to support life?* The highlighted words in each clips are the keywords selected by one of the crowdworkers, showing that workers are finding useful words for classification.

To learn a similarity function between clips, we use the crowd labels and keywords to train a classifier that predict how likely two clips to be labeled as similar. Although the judgments from workers via the HIT interface about which clips go together provide valuable training information, we need to leverage these judgments to bootstrap similarity judgments for the clips that they did not label and to resolve potentially conflicting or partial category judgments. To do so we trained an SVM classifier in real-time to identify the set of keywords that are most indicative of categories and predict whether two clips in the dataset belonged to the same cluster. The training events are all possible pairwise combinations of clips in the clusters obtained with the HIT interface, which may include both positive (similar) and negative (different). The feature dimensions are all the keywords highlighted by the crowdworkers, and the value of each dimension is the product of the number of times that keyword occurred in the two clips. In general, the keywords labeled by the crowdworkers contain little irrelevant information compared to all words in the clips, but there could still be some highlighted words that are not indicative of a category. For example, one crowdworker worked on the dataset for "*How do I unclog my bathtub drain?*" labeled "*use*", "*a*", and "*plunger*" as three keywords. Even though *plunger* is a very indicative feature for clustering this dataset, the first two highlighted words seem too general to be useful. Using a linear kernel to estimate the weights for the different dimensions (i.e., keywords) seems well suited for our purpose [29, 177]. Further, if the same keyword is used by different crowdworkers but lead to very different labels, the linear SVM model will give lower weight to the corresponding dimention and thus lower the effects of keywords that are less indicative of the categories. We use LIBSVM which implements a variant of Platt scaling to estimate probability [111, 136]. The overall intuition is that the SVM classifier is doing a form of feature selection, weighting those words in clips that could maximally distinguish clips amongst clusters.

In a preliminary experiment, we tested using all words in the clips as features to train the SVM model. The intuition is machine algorithms might do a better job at identifying keywords that can outperform keywords identified by crowdworkers. However, the results show that using all words as features did not yield better results, and having much higher feature dimensions increases the training time significantly.

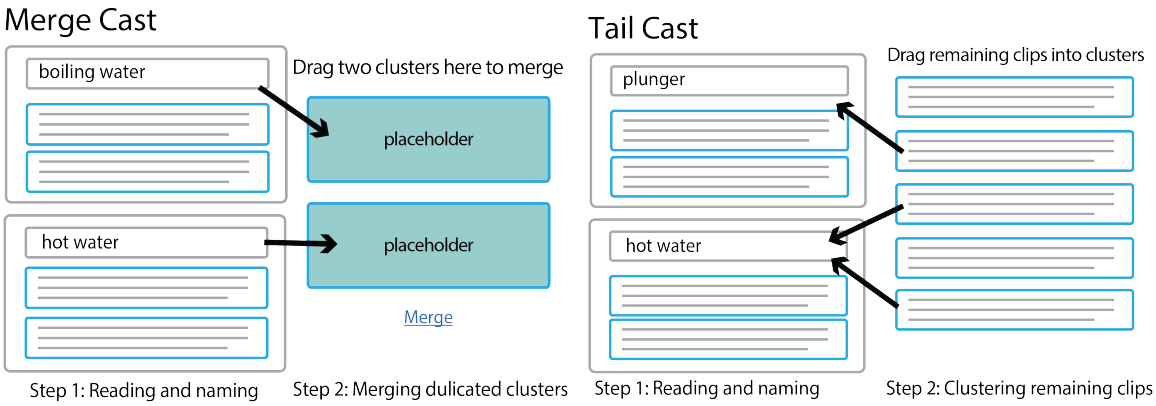

Figure 3.4: The HITs for Merge Cast: Naming and merging existing clusters and Tail Cast: Clustering remaining clips.

Finally, with the probability output of the SVM model as a similarity function between clips and a stopping threshold of 0.5 probability, we use a hierarchical clustering algorithm that serves as the Gather Backbone to capture head clusters.

3.2.2 Gather Backbone: Hierarchical Clustering

Using a multiple-stage approach with different types of microtasks can make it difficult to fuse together the different crowd judgements to form a coherent result. A key element to our approach in *casting* for category judgments in different ways is that we have a unifying mechanism to *gather* them back together. For example, throughout our process we cast for human category judgments in very different ways, including having people identify seed clusters (the Head Cast), merge duplicated categories (the Merge Cast), and classify the tail of the distribution (the Tail Cast). Instead of creating ad-hoc links between these judgments we propose using a unifying gathering mechanism composed of a machine learning backbone which translates the different *casted* judgments into similarity strengths used as the basis of clustering. We believe this *Cast and Gather* pattern may be useful as a way to conceptualize the relationship between machine algorithms and crowd judgments for a variety of tasks.

To build a complete clustering workflow with multiple casts, we use a hierarchical clustering algorithm as the backbone that connects different casts. More specifically, the backbone algorithm fuses the judgements from different crowdworkers working on the same cast into clusters, which, in turn, become the shared context transferred to the next cast of the workflow.

With a clip similarity function from the prior cast and a stopping threshold, the hierarchical clustering method initially treats each clip as a cluster by itself, and iteratively merges the two most similar clusters until a threshold is reached. The result is a partially clustered dataset with clusters and singletons. When the backbone is used after the last cast in the workflow, each singleton is then merged into the most similar cluster. The similarity between two clusters is defined as:

$$ClusterSim(\omega_1, \omega_2) = \frac{1}{|\omega_1||\omega_2|} \sum_{t_j \in \omega_1} \sum_{t_k \in \omega_2} ClipSim(t_j, t_k) \tag{3.1}$$

where ω_1 and ω_2 are the two clusters, t_j and t_k are each of the clips in ω_1 and ω_2, respectively,

and the $ClipSim()$ function is the given similarity function between clips.

3.2.3 The Merge Cast

While the Head Cast is designed to find the large clusters in the head of the distribution, since each crowdworker works independently, some of those clusters may actually be different subsets of the same larger category or the same categories based on different keywords (e.g., *sunlight* vs *natural lighting*). The Merge Cast is designed to consolidate existing clusters by merging duplicated categories. The input to this cast is a set of clusters that may or may not cover the entire dataset, and the output is fewer or equal number of clusters each with a list of ranked short descriptions. The challenge with detecting duplicate categories is that people need to understand what is in each category first. We start by presenting a set of existing clusters, and asking crowdworkers to name each of them. This acts as a defensive design[95] that ensures the crowdworkers understand the current context (scope and abstraction level), and also to obtain short descriptions for each of the clusters. Crowdworkers are then asked to merge identical categories by dragging them into the placeholders on the right (Figure 3.4).

If there are too many head clusters to fit into a microtask, the Merge Cast can be run recursively by first running on disjoint sets of existing clusters to consolidate them independently. Then, run another sets of Merge Cast on the output of each initial Merge Casts, and recurse until the output reduces to a set of clusters that could be presented in a global Merge Cast to ensure consistency. The assumption here is that the set of clusters in the final output of Alloy should be manageable by a single person to be useful. We also wanted to point out that the number of clusters is likely to scale much slower than the size of the dataset for many real-world data.

With the labels from the crowdworkers, we will again use the Gather Backbone to combine the judgements. The goal is to merge existing clusters if more than half of the crowdworkers also merged them in their solutions. Since in the Merge Cast workers can not break up existing clusters or reassign clips, we can formulate the clip similarity function as:

$$ClipSim(t_1, t_2) = \frac{1}{N}|\{\omega : t_1, t_2 \in \omega \; and \; \omega \in \Omega\}| \tag{3.2}$$

where t_1, t_2 are the two clips, N is the total number of crowdworkers, Ω is the set of all clusters created by all crowdworkers, and ω is any cluster that contains both clips. This function is robust against a few workers doing a poor job. For example, if one crowdworker assigned every clip in the dataset to a single, general cluster (e.g., *answers*), the effect to the similarity function would be equivalent to having one less crowdworker and applying Laplacian smoothing. It is a common concern for crowd-based clustering methods that novice workers may create overly abstract categories (e.g., *solutions* or *tips*), that covers all items in the datasets. With our approach, it would require more than half of the workers to merge all items into a single cluster to generate a single cluster in the output.

From the output of the Gather Backbone, we rank the short descriptions associated with each cluster. Since clips are labeled by multiple crowdworkers, each cluster is associated with multiple descriptions via its clips. We use the F1 metric to rank these names to find the most representative description for each cluster, where the precision of a name label is defined as the number of clips in the cluster that it associates with divided by the size of the cluster, and recall as divided by the total number of clips associated with it.

Dataset	sources	workers	clips	bad	clusters
Q1:*How do I unclog my bathtub drain?*	7	16	75	25%	8
Q2:*How do I get my tomato plants to produce more tomatoes?*	18	13	100	10%	8
Q3:*What does a planet need to support life?*	19	19	88	31%	7
Q4:*What are the best day trips possible from Barcelona, Spain?*	12	12	90	18%	16
Q5:*How to reduce your carbon footprint?*	20	11	160	14%	11
Q6:*How do I unclog my bathtub drain?*	17	23	159	14%	11
Wiki: Talk page sections for the Wikipedia *Hummus* article	N/A	N/A	126	0%	13
CSCW: Abstract sections of CSCW 2015 accepted papers	N/A	N/A	135	0%	45

Table 3.1: Datasets used for evaluation

3.2.4 The Tail Cast

The Tail Cast is designed to clean up the remaining singleton clips by classifying them into existing clusters or creating new clusters. The intuition is that even though machine learning techniques can produce high performance output, sometimes it is achieved at the expense of sacrificing the border cases. Human-guided "clean up" is often necessary for data produced by a machine learning model. The input of this cast is a set of existing clusters (with or without short descriptions) and a set of remaining clips. The output is a set of clusters with short descriptions.

We use an interface similar to the Merge Cast (Figure 3.4), and asked crowdworkers to review or name each of the existing clusters first, so that they build up better global understanding of the dataset before they organize the remaining clips. If Merge Cast was performed previously, their names are presented to lower cognitive load. The crowdworkers are then instructed to cluster the unorganized clips shown on the right by assigning them into existing clusters, creating new clusters, or removing uninformative clips. If there are too many remaining clips to fit into a single microtask, they are partitioned into groups of 20 items. Even though we may be dividing the remaining clips into partitions, all workers in the Tail Cast starts with learning the same global context that is the set of existing clusters from the Head Cast.

Finally, we use the Backbone Gather again to combine the multiple solutions from the crowdworkers. The goal is analogous to the goal of the Merge Cast: if two clips are assigned to the same category by more than half of the crowdworkers, they should be in the same cluster in the combined solution. For the similarity function, we simply replace the variable N in Equation 2 by the degree of redundancy.

3.3 Evaluation

3.3.1 Evaluation Metric and Datasets

Unlike evaluating a classification task, which would typically be based on the precision and recall of pre-defined classes, evaluating clusters is not as straightforward due to the potentially different number of classes in the gold-standard and the system output. For example, high precision can be achieved by simply having more clusters in the output and the mapping between them. To address this, we use the normalized mutual information metric (NMI), which is a symmetric measurement sensitive to both the number of clusters, and the precision of each cluster. Specifically, it compares all possible cluster mappings to calculate the mutual information, and normalizes by the mean entropy so that the numbers are comparable between different datasets:

$$NMI(\Omega, C) = \frac{I(\Omega, C)}{0.5 * [H(\Omega) + H(C)]} \tag{3.3}$$

where Ω is the output clusters and C is the gold-standard clusters. The mutual information I is defined as:

$$I(\Omega, C) = \sum_k \sum_j P(\omega_k \cap c_j) log \frac{P(\omega_k \cap c_j)}{P(\omega_k)P(c_j)} \tag{3.4}$$

where ω_k and c_j denotes each of the clusters in Ω and C, respectively. The probability $P(\omega)$ of item set ω is defined as $|\omega|/N$, where N is the number of total items. Finally, the mutual information is normalized by the mean entropy of Ω and C, so that the scores are comparable across datasets. To give some intuition, given w that maps to a gold-standard cluster c, we can calculate the precision by $P(w \cap c)/P(w)$ and recall by $P(w \cap c)/P(c)$, and the metric considers both with $P(w \cap c)/P(w)P(c)$. However, in reality it may be difficult to obtain such mappings, and the metric simply sums up scores of all possible mappings weighted by probability $P(w \cap c)$.

We use NMI for it is widely found in the literature for clustering evaluation. A more recent study found that it might favor datasets with more clusters, and proposed a variant that adjusts for randomness (AMI, [170]). We acknowledge this is a potential limitation, but found that the number of clusters Alloy produced were quite close to the gold-standard (average 10.3 vs 10.2), suggesting the concerns may be minimized. To be on the safe side, we also measured Alloy's performance using AMI on two datasets and found similar results.

In order to evaluate Alloy, we compared it to other machine learning and crowdsourcing clustering approaches in three different contexts: information seeking, Wikipedia discussions, and research papers. These contexts all involve rich, complex data that pose challenges for automated or existing crowd approaches. Below we describe each dataset and how we either generated or collected gold-standards.

3.3.2 Information Seeking Datasets

We picked five questions asked on popular Q&A forums (e.g., Quora, reddit, and Yahoo! Answers) that covered a diverse range of information needs. We then posted these questions to Amazon Mechanical Turk (AMT), and asked each crowdworker to find 5 webpages that best answered the questions in Table 3.1. The top sources were sent to workers to highlight clips that would help answer the question via an interface similar to that described in [97]. The first four datasets (Q1 to Q4) collected consist of 75 to 100 clips, extracted from 7 to 19 webpages using 12 to 19 crowdworkers. In addition, we also collected two datasets with more than 150 clips (Q5 and Q6) by gathering more clips from the sources.

To generate gold standards, two graduate students clustered each dataset independently. Raters were blind to Alloy's clusters, and no discussion on clustering strategies nor predefined categories were made prior to the process. Raters initially read every item in the dataset to build global understanding before they started organizing. Conflicts between raters were resolved though discussion. The first author participated in labeling two (out of the seven) datasets, but was always paired with another annotator outside of the research group. To measure inter-annotator agreement, we used the symmetric NMI metric as described in the previous section.

The agreements between raters are shown in Table 3.2. The datasets for "*How do I unclog my bathtub drain?*", "*How do I get my tomato plants to produce more tomatoes?*" and "*What are the best day trips possible from Barcelona?*" had high agreement between the two annotators of 0.7 to 0.75 NMI. For the "*What does a planet need to support life?*" dataset, the agreement was significantly lower (0.48). We kept this dataset to show the limitations of the proposed method, and we will discuss further in later sections. For the two larger datasets Q5 and Q6, the agreement scores were around 0.6.

3.3.3 Research Papers

Since some of the questions in the above dataset were about common daily life problems, an open question is whether crowd judgements were based on workers' prior knowledge or the context we provided them. To evaluate the system using more complex data where workers would likely have little prior knowledge we turned to research papers from the 2015 CSCW conference. For this dataset we used the official conference sessions as the gold standard for evaluation. The intuition is that conference organizers would place similar papers together in the same session. We acknowledge that the objectives of organizing conference sessions are not entirely the same as Alloy; most notably, conference session planning requires schedule conflict resolution and fixed size sessions. However, session co-occurrence represents valuable judgments from experts in the community about which papers belong to a common topic, and even though each cluster is smaller in size (e.g., 3-4 papers per session) we can look at whether papers put together by experts are also put together by Alloy and the other baselines [43].

3.3.4 Wikipedia Editor Discussion Threads

Wikipedia relies on its editors to coordinate effectively, but making sense of the archives of editor discussions can be challenging as the archives for a single article can consist of hundreds or thousands of pages of text. We use as a dataset the discussion archives of the *Hummus* article, a popular yet controversial article, and use the discussion threads as the set of documents. The talk page consists of 126 discussion threads about various issues of the main articles that spans over the past 10 years (Table 3.1). Two annotators read the main article and the full talk threads before they started the labeling process. The NMI score between the two annotators was *.604*, which is comparable to the two other large datasets Q5 and Q6.

Wikipedia data can be more difficult to organize than previously mentioned datasets, because it can be organized in very different ways, such as topics, relations to the main article sections, and mention of Wikipedia guidelines [5]. The annotators also had a hard time coming up with a gold standard through discussion, and found both their categorization solutions to be valid. Therefore, instead of creating a single gold standard, we report the NMI scores between Alloy's output and each of the annotators.

3.3.5 External Validation, Robustness, and Generalizability

In the following sections, we will describe three experiments and their results followed by an application-oriented evaluation. For the three experiments, two workflows that uses the Gather to connect the different Casts are tested. The first experiment is an external evaluation that compares Alloy with other approaches. We use the full workflow that consists of the Head Cast, the Merge Cast, and the Tail Cast to cluster the six information seeking datasets (Q1-

DS	InterAnnot.	Workflow1	Workflow2	TFIDF	Keywords	LSA	LDA	#clusters	
								Alloy	exp
Q1	.734	**.759*** σ=.033	.550* σ=.093	.510	.647	.512	.478	7	8
Q2	.693	**.687*** σ=.016	.467* σ=.046	.534	.562	.537	.506	8	8
Q3	.477	**.468**	.425	.390	.440	.467	.442	7	7
Q4	.750	**.727**	.633	.673	.676	.704	.603	14	16
Q5	.630	.576	-	.568	.508	**.582**	.551	16	11
Q6	.588	**.588**	-	.462	.492	.497	.456	10	11
AVG	.645	**.634**	-	.523	.554	.550	.503	10.3	10.2
CSCW	-	**.748**	-	.584	.652	.691	.725	23	45

Table 3.2: Evaluation Results. * indicates mean of 11 runs using different workers.[1]

Q6), and compare with previous crowd-based methods and four machine algorithm baselines. The second experiment is an internal evaluation that tests the robustness of Alloy by using different number of workers in the Head Cast and the Tail Cast. Finally, in our last experiment, we test Alloy's performance on two different types of datasets: Wikipedia editor discussions and research papers. Finally, to investigate the usefulness of the structures produced by Alloy, we used a prototype system called Knowledge Accelerator [66] to synthesize Alloy clusters for the information seeking datasets into report-styled articles and compare the articles against top Google search results.

3.3.6 Experiment 1: External Validation

We first look at how Alloy compares with machine algorithms, other crowd algorithms, and inter-expert agreements. In the Head Cast, crowdworkers highlight keyword and cluster similar clips via searching, and in the Tail Cast another set of crowdworkers organizes all remaining clips.

We compare this Workflow 1 to three baselines that are commonly used in the clustering literature: latent Dirichlet Allocation (LDA) [19], latent semantic analysis (LSA) [47], and TF-IDF [89, 114]. We also compare against a hybrid baseline that uses human-identified keyword vectors from the Head Cast. This aims to test the value of the approach beyond the human identification of keywords by trying to cluster using only the keywords. In addition to comparing against automatic methods, we also compare Alloy to a popular crowd based method. The evaluation conditions are summarized below:

- *Workflow1*. The workflow with ten crowdworkers each for the Head Cast and the Tail Cast for Q1-Q4. An additional five workers for the Merge Cast for Q5-Q6. Each HIT costs 1 USD.
- *TF-IDF*. Weighted cosine similarity as the similarity function for the Gather. No human-computation was employed.
- *Crowd keywords*. Cosine similarity based on worker-highlighted keywords from the Head Cast as the similarity function for the Gather.
- *LSA*. The LSA model is used as the similarity function for the Gather. No human-computation was employed.
- *LDA*. The LDA topic model is used as the similarity function for the Gather. No human-computation was employed.
- *Cascade*. A version of Cascade with only one recursion using the default parameters as

described in the paper.

Results

Alloy introduces a novel approach for providing context in the microtask setting with the sampling mechanism in the Head Cast. We captured crowdworkers' behavior during the tasks and found that nearly all (97.5%) workers used the sampling mechanism to gain context beyond the initial four items. On average, each worker sampled 15.1 items , and more specifically, 11.3% sampled more than 25 items, 23.8% sampled 15~24 items and 62.5% sampled 5~14 items.

Comparing with Machine Algorithms

On average, the proposed method performed significantly better and more consistent than all machine baselines (Table 3.2). In the worst case, Alloy clusters measured 0.058 NMI lower than the inter-annotator agreement, while the baseline systems measured more than 0.1 NMI lower in most cases. In a few cases some baselines also performed well (e.g., LSA performed slightly better on Q5), but none of them produced good results consistently across all datasets. Compared to the gold-standard clusters, Alloy produced clusters about as close to the gold-standard clusters as the two human annotators were to each other, despite the judges' advantages of having a global view of the datasets and multiple rounds of reading, labeling, and discussion. In addition, worker-identified keywords consistently outperformed TF-IDF, showing that the crowdworkers are extracting keywords in the Head Cast that are salient for identifying clusters each dataset. On the two larger datasets (Q5 and Q6), Alloy achieved similar performance as the four smaller datasets; better and more consistent comparing to the baseline systems, and near experts agreement comparing to the gold-standard.

Note that for every machine algorithm baseline we explored multiple parameters for each of the four questions, (hyper-parameters, number of topics, stopping threshold), and report the highest scores. The results of the baseline algorithms are likely over-fitting to the data, but we wanted to compare Alloy to these algorithms under their best possible settings [44].

Comparing with Previous Crowd Methods

We compare Alloy with Cascade using datasets Q1-Q4, a popular crowd-based method for discovering taxonomies in unstructured data based on overlapping crowd clusters [42]. We implemented a simplified version of Cascade using the parameters described in the paper, but with only one recursion. We acknowledge that fine tuning and multiple recursion might improve Cascade's performance, but the numbers from our evaluation are consistent with the results reported in the Cascade paper based on the same metric and similar datasets.

On average, 84% of categories generated with Alloy were shared with clusters in the gold standard, versus 50% for Cascade. Cascade produced soft clusters where child clusters did not necessarily have all the items included in their parents, which breaks the assumptions of using NMI. To produce a direct comparison, we use the gold standard to greedily extract best matching, overlapping clusters that cover all items, and evaluated them using the average F1. In essence, this simulates an omniscient "oracle" that gives Cascade the best possible set of cluster matches, and so is perhaps overly generous but we wanted to err on the conservative side. The average F1 scores for each questions using Alloy are .72, .54, .48, .52, and using Cascade are .50, .48, .42, .39, showing a consistent advantage across questions. Furthermore, Alloy

Expert	Alloy	Cascade Single Pass
Hot Water	Hot Water / Clearing a drain with hot water	problem/symptom (21) drain clean (55) slow drain (36)
Plunger	Plunge / Vigorous Plunging	plumbing (52) comments on why solutions are not working (8)
	Plunger / Dealing with Overflow	how do i unclog a drain (46) bathroom (39)
Plumbing Snake	Snake the Drain / Plumbing Snake	clean drain (42) chemical solution (16)
Remove Cover	Remove the Drain Cover / Check drain cover	drano (11) mechanisms that unclog drain (28)
Chemicals	Drain Cleaner / Use drain cleaner for hair clogs	steps in unclogging the drain (45) manual solution (32) internet forum suggestion (33)
Bent Hanger Wire	Remove Hair Clusters / Using a bent wire to clear a drain	how do i unclog my bath tub (49) drain water (31) help unclog the drain (47) helpful tip (45)
Call a Plumber		what should i do with a plunger cup to open a drain (9) plunge (15) help (37)
Shop Vacuum		what to do with a drain clog (43) helpful response (35) technical advice (26)
		drain (50) reasons why drains become clogged (15) reasons why a drain becomes clogged (13)
		uncategorized (2)

Figure 3.5: Categories comparison for Q1

achieved this better performance at a lower cost (average $20 for Alloy vs $71 for Cascade), suggesting that machine learning can provide valuable scaling properties. We show categories created by experts and elicited from the two systems in Figure 3.5 to give a better sense of the datasets and the output.

3.3.7 Experiment 2: Robustness

In this section, we examine the robustness of Alloy by varying the number of crowdworkers employed in the Head and the Tail Cast on datasets Q1-Q4. We start with having only 1 worker in the Head Cast, and evaluate performance as we hire more workers until we have 20. To test the two phase assumption, in a second condition, we switch to the Tail Cast after hiring 10 workers in the Head Cast, and continue to hire 1 to 10 more workers. This way, we can characterize the cost/benefit trade-offs in hiring different amount of human judgments. Further, by omitting the Tail Cast completely in the first condition, we can verify the two phase assumption by comparing the performance of a two-phase process (Head Cast and Tail Cast) with a one-phase control (Head Cast only) while equaling the number of workers:

- *Workflow1*. The workflow with ten crowdworkers each for the Head Cast and the Tail Cast. Each HIT costs 1 USD.

[1]We also evaluated Q1 and Q2 using the AMI metric that accounts for randomness. The inter-annotator agreements are .674 and .643, respectively, and Alloy performed .674 and .609, respectively. See the Evaluation Metric Section for detail.

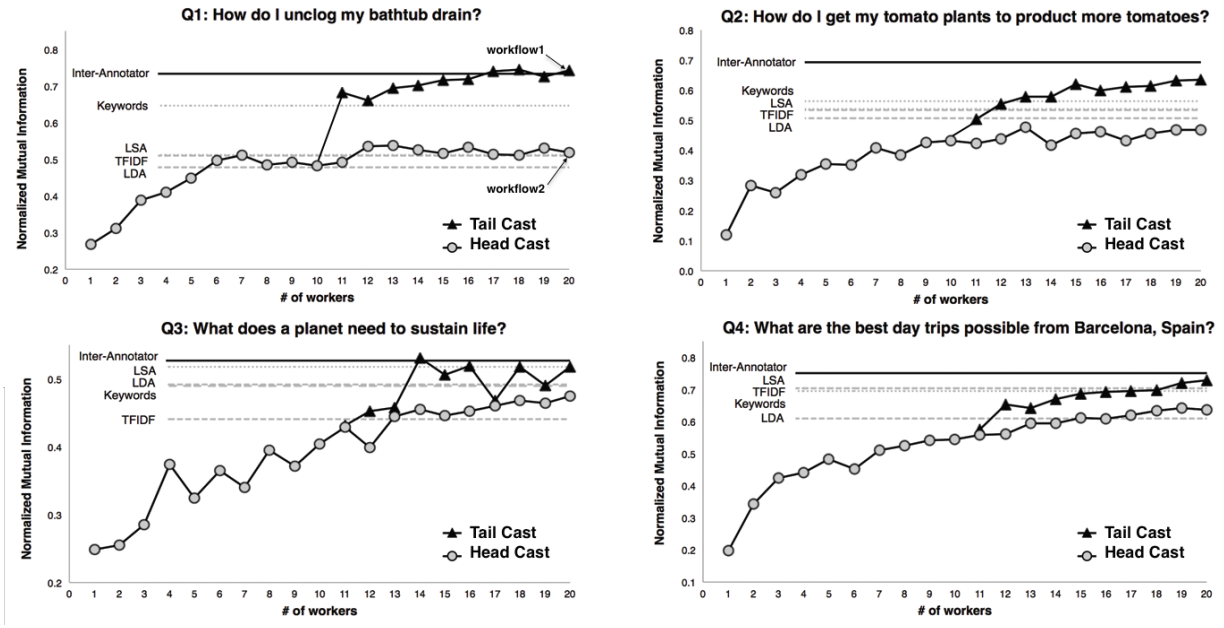

Figure 3.6: Performance comparison of using different number of crowdworkers in the Head Cast and the Tail Cast.

- *Workflow2*. The workflow with twenty crowdworkers and the Head Cast only. Each HIT costs 1 USD.

In addition, to test how robust Alloy is to the variance of crowdworkers on Amazon Mechanical Turk, we also hired eleven sets of ten different crowdworkers (a total of 440) for each Head and Tail Casts for Q1 and Q2.

Results

In Figure 3.6, we show the performance of employing different number of workers in the Head and the Tail Cast. Initially, increasing the number of workers in the Head Cast shows significant performance improvements. However, after gathering training data from around 10 workers, the performance gain from hiring additional crowdworkers decreases notably. Instead, performance improved significantly even with only a few additional crowdworkers in the Tail Cast to refine the clusters. Overall, having 10 crowdworkers in each of the Head and Tail Cast consistently outperformed having all 20 crowdworkers in the Head Cast across all four questions (Table 3.2), suggesting there is significant value in the Tail Cast.

For Q1 and Q2, we also ran Alloy eleven times using different crowdworkers, and compared the results against the gold-standard labels and also with each other. Comparing to the gold-standards, which have inter-annotator agreements of $.734$ and $.693$ for Q1 and Q2 respectively, Alloy produced an average NMI of .759 (SD=.016) and .687 (SD=.016), respectively. Further, the average pair-wise NMI score of the 11 runs are .819 (SD=.040), and .783 (SD=.056), respectively, suggesting Alloy produces similar results using different crowdworkers on the same datasets.

3.3.8 Experiment 3: Other Datasets

In this experiment, we use the same distributed workflow to test Alloy using the Wiki and CSCW datasets as described in the Dataset Section, in order to test how Alloy generalizes to other types of data. These datasets contain long academic documents or editorial discourses that are infeasible to present multiple items to the crowdworker in one HIT. Instead, we show a small portion of each item in the datasets to the crowdworkers. For each item in the Wiki dataset, we display the thread-starter post and the first two replies. For the CSCW dataset, we present the abstract section of each paper, and compare results with the official conference sessions. Machine baselines were however given access to all of the text of the paper and the full discussion threads in order to provide a strong test of Alloy's approach.

Results

For the CSCW dataset, Alloy outperformed all machine baseline systems with *.748* NMI score using conference sessions as the gold standard Table 3.2. The Keyword baseline outperformed the TF-IDF baseline (*.652* vs *.584*), showing that the crowdworkers are extracting valuable keywords in the Head Cast, despite that research papers may be difficult or impossible for crowdworkers to understand. On the other hand, Alloy produced 24 categories out of 135 abstracts, more than all other datasets. One possible assumption is that it may be more difficult for novice workers to induce abstract categories when organizing expert dataset, leading to higher number of more lower level categories in the outcome.

For the Wiki dataset, the NMI score between annotators was *.604*, which is comparable to the two other large datasets Q5 and Q6. Comparing to the two sets of expert labels independently, Alloy's output measured *.528* and *.507*. Compared to all previous results, Alloy seemed to perform less favorably on this dataset. As mentioned in the Dataset Section, the raters found the this dataset the most difficult to organize, as there are many different valid structures that the two annotators were unable to reach an agreement also hints that the space of valid solutions may be larger on this dataset. In addition, we only showed the first three comments of each discussion to the crowdworkers, whereas the annotators and the machine baselines have access to the full discussion. We acknowledge length of items is a limitation, and will discuss in detail in the Discussion Section.

3.4 Application: Knowledge Accelerator

This work was previously published in ACM SIGCHI 2016 [66] and has been adapted for this document.

To evaluate the usefulness of structures generated by Alloy in a more realistic scenarios, we first used Alloy to clusters a larger set of information seeking datasets (Table 3.3) collected using the same procedure as described in section 3.3.2. We then developed a prototype system called the "Knowledge Accelerator" (KA) to synthesize the output of Alloy into articles. Each of the cluster produced by Alloy corresponds to a different section in an article. An example of the output of the system for the target question "How do I get my tomato plants to produce more tomatoes?" can be found in Figure 3.7.

In addition, the KA system probes how to accomplish a complex information synthesis task entirely through relatively small contributions. We limited our maximum task payment to $1 US,

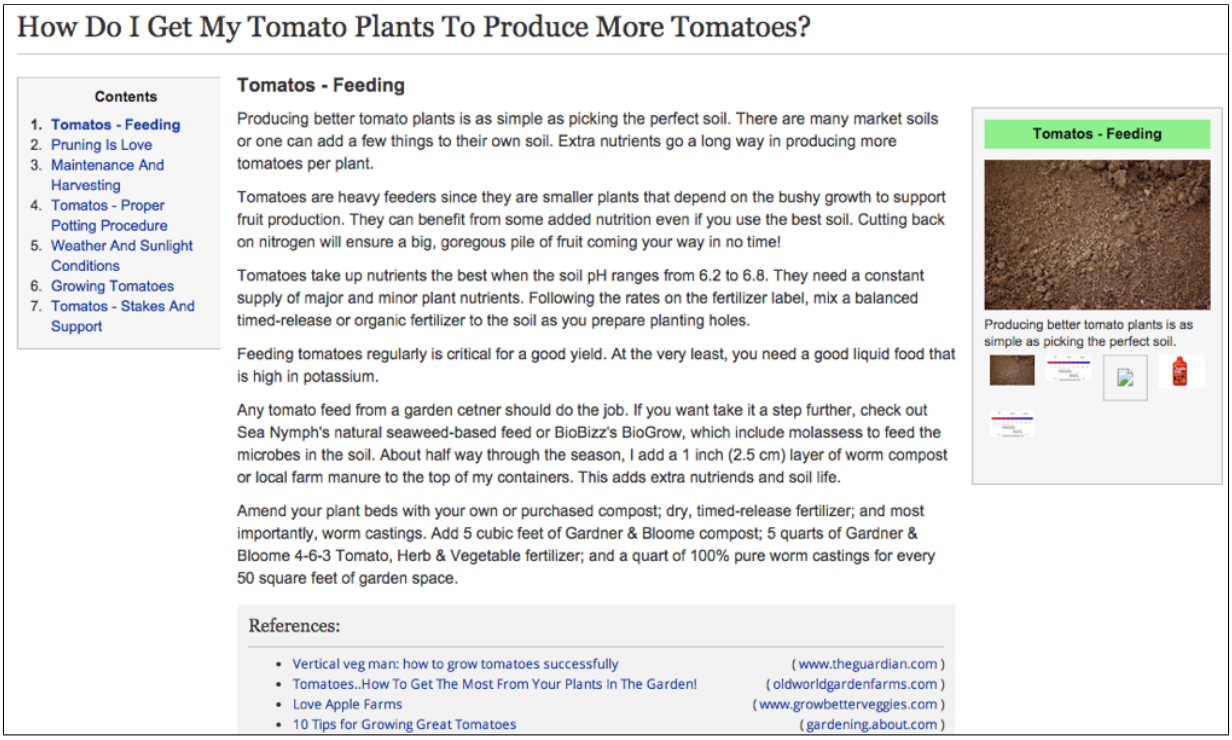

Figure 3.7: Example report synthesized by the Knowledge Accelerator system. The table of content on the left listed cluster names generated from Alloy, each corresponded to a different section in the report.

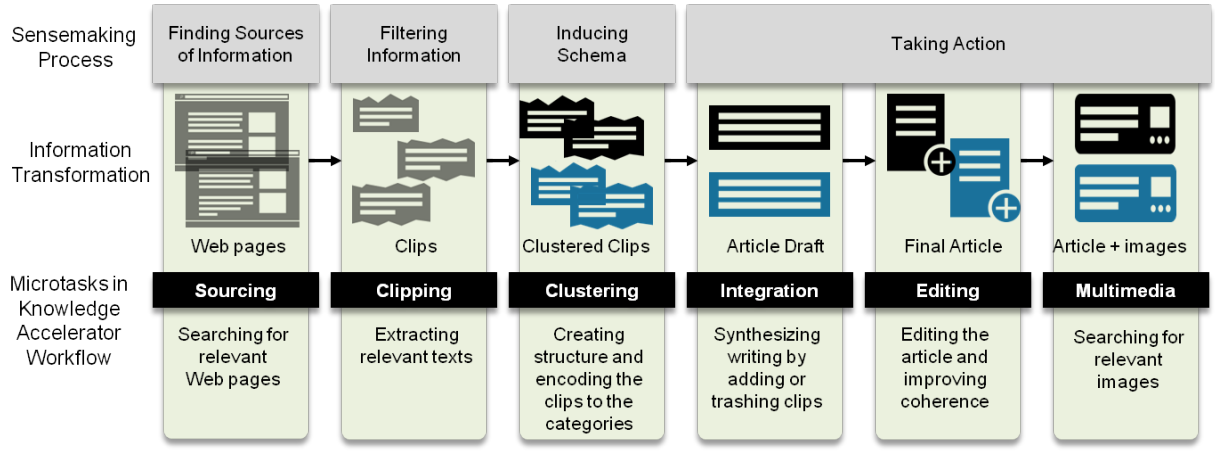

Figure 3.8: The process of the Knowledge Accelerator (KA). Alloy is used for the Clustering Stage of the pipeline.

Question	N	Score
Q1: *How do I unclog my bathtub drain?*	116	0.292 *
Q2: *How do I get my tomato plants to produce more tomatoes?*	177	0.420 *
Q3: *What are the best attractions in LA if I have two little kids?*	158	-0.044
Q4: *What are the best day trips possible from Barcelona, Spain?*	98	-0.109
Q5: *My Worcester CDi Boiler pressure is low. How can I fix it?*	139	0.878 *
Q6: *2003 Dodge Durango has an OBD-II error code of P440. How do I fix it?*	138	0.662 *
Q7: *2005 Chevy Silverado has an OBD-II error code of C0327. How do I fix it?*	135	0.412 *
Q8: *How do I deal with the arthritis in my knee as a 28 year old?*	139	0.391 *
Q9: *My Playstation 3 has a solid yellow light, how do I fix it?*	119	0.380 *
Q10: *What are the key arguments for and against Global Warming?*	138	0.386 *
Q11: *How do I use the VIM text editor?*	138	0.180

***** = significant at $p < 0.01$ after Bonferroni correction

Table 3.3: Average difference between the KA output and top websites for the eleven questions (positive indicates higher ratings for KA, negative indicates higher ratings for the competing website). Each rating was an aggregate of 6 questions on a 7-point Likert scale.

aimed at incentivizing a Target task time of approximately 5-10 minutes. Critically, the KA system accomplishes this process without a core overseer or moderator. Figure 3.8 shows the overview of the KA System with Alloy being the Clustering Stage. For more details on the KA system refer to [66].

We evaluated the usefulness and coherence of the articles by comparing them against webpages an individual might use if they were to complete the same tasks without KA and Alloy — Top Google search results that consists of expert-written articles published by trusted sources such as CDC.gov or the New York Times, as well as popular online forums such as TripAdvisor and Yahoo Answers.

3.4.1 Experimental Settings

Eleven topics were selected for evaluation by browsing question and answer forums, Reddit.com, and referencing online browsing habits [27]. For questions Q3 and Q8 we added additional constraints (i.e., having kids and age) to test the performance of the system for more personalized questions. To compare the two conditions, participants were recruited through the Amazon Mechanical Turk US-only pool and paid $1.50 for rating two webpages. Each participant was randomly assigned an output article from KA and a top search result webpage for the same topic (Figure 3.9), and rate both webpages based on six criteria using 7-point Likert scale questions and provided free-form explanations: *comprehensiveness*, *confidence*, *helpfulness*, *trustworthiness*, *understandability*, and *writing*. We averaged ratings on these dimensions into a single score representing the overall perceived quality of the page.

3.4.2 Results

The costs of running a question through the KA system is shown in Table 3.4. Across the 11 topics we tested, a full run with around 100 short text clips costed an average of $108.50, of which around $15 is spent on searching and extracting the text clips from webpages, $20.00 is spent by the Alloy system, and the rest on synthesizing each of the Alloy clusters into a section in the final article and making sure the different sections are coherent.

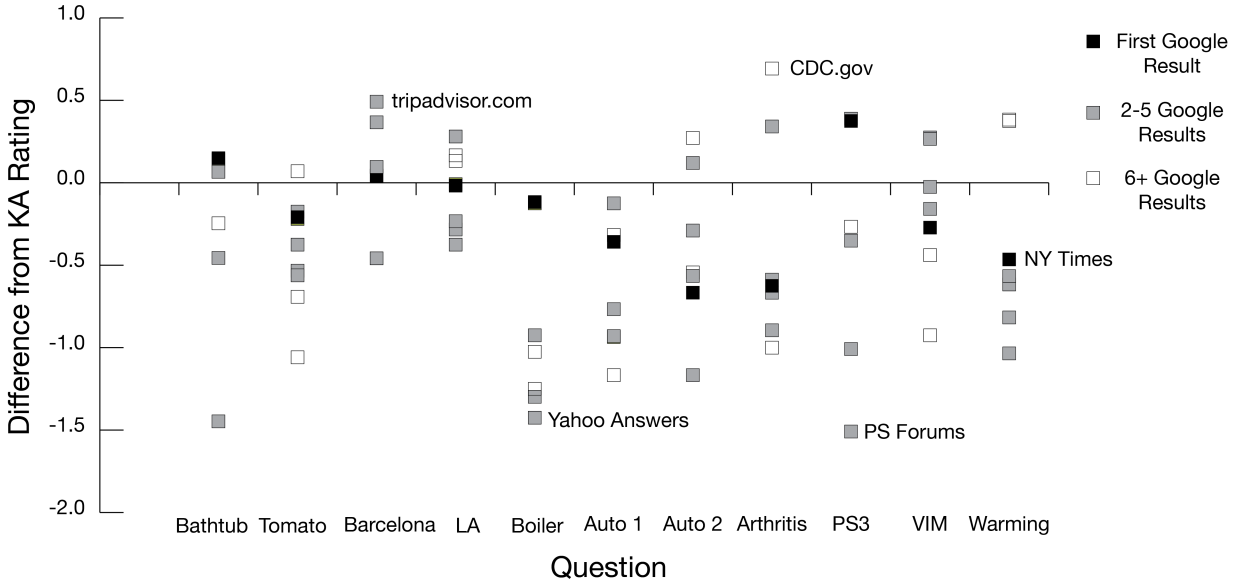

Figure 3.9: Results across questions and websites. Points represent the average aggregate score difference between the KA answer and an existing site

Phase	**Task Pay**	**Avg. # of Tasks**	**Avg. Cost**
Sourcing	$0.25	15	$3.75
Clipping	$0.50	21.6	$10.80
Alloy Head Cast	$1.00	10	$10.00
Alloy Merge + Tail Cast	$1.00	10	$10.00
Integrate	$0.50	37.2	$18.60
Edit 1	$0.75	28.8	$21.60
Edit 2	$1.00	28.8	$28.80
Images	$0.50	9	$4.50
Total		160.4	$108.05

Table 3.4: Average number of worker tasks and average cost per phase, and overall, to run a question.

categories induced during clipping (without Alloy):
Boil Water, use hot water, Plunger, try a snake, How to Remove drain stopper, bleach, Use Drano Max Gel, baking soda, drain, tips to unclog, problem, tools, research, internet research, ..., etc.

---

categories induced by Alloy:
Hot Water, Plunge, Plunger, Snake the Drain, Remove the Drain Cover, Drain Cleaner, Remove Hair Clusters.

---

gold-standard categories:
Hot Water, Plunger, Plumbing Snake, Remove Cover, Chemicals, Bent Wire Hanger, Call a Plumber, Shop Vacuum.

Figure 3.10: Categories induced from different stages for Q1: *How do I unclog my bathtub drain?*

Aggregating across all questions, KA output was rated significantly higher than the top 5 Google results (KA: $\bar{x} = 2.904$ vs Alt. Sites: $\bar{x} = 2.545$, $t(1493) = 13.062$, $p < 0.001$). An analysis of individual questions corrected for multiple comparisons is shown in Table 3.3.

The strongly positive results found were surprising because some of the websites in the comparison set were written by experts and had well-established reputations. Only on the two travel questions, Barcelona ($\bar{x} = -0.109$) and LA ($\bar{x} = -0.044$), and the VIM question ($\bar{x} = 0.180$) did the KA output not significantly outperform the comparison pages. A closer examination of these pages suggests that for the two travel questions, because of the strong internet commodity market surrounding travel, a considerable amount of effort has been spent on curating good travel resources. Even with the slightly more specific LA query, there were still two specialized sites dedicated to attraction for kids in LA (Mommypoppins.com and ScaryMommy.com). The VIM question represented a mismatch between our output and the question style. A number of the sources for the question were tutorials, however in the clipping phase, these ordered tutorials were broken up into unordered clips, creating an information model breakdown. This points out an interesting limitation in the KA approach, and suggests that adding support for more structured answers (e.g., including sequential steps) could be valuable future work.

The strong performance of the system is perhaps surprising given that its output was generated by many non-expert crowd workers, none of whom saw the big picture of the whole, and Alloy is a core component that provided useful and coherent structures for producing the final report. Initially we had workers provide labels to categorize each clip, which we planned to use to develop a structure for the article. However, the lack of context of the bigger picture made these labels poorly suited for inducing a good structure. For example, in Figure 3.10 the top box shows the category structure induced by crowdworkers during clipping and without using Alloy during clipping, categories induced using Alloy, and gold standard categories developed by two independent annotators with access to all clips and sources, respectively. Categories induced without using Alloy matched poorly with the gold standard categories, and include categories with very different abstraction levels (e.g., *Use Drano Max Gel* vs *tips*). On the other hand, Alloy produced categories that were more coherent and matched more with gold-standard categories.

While we do not believe that this should be interpreted as a replacement for expert creation and curation of content, instead, the power of the system may actually be attributable to the value created by those experts by generating content which the crowd workers could synthesize and structure into a coherent digest. This explanation suggests that the approach would be

most valuable where experts generate a lot of valuable information that is unstructured and redundant, such as the automative questions in which advice from car enthusiasts was spread across many unstructured discussion forums. In contrast, KA's output did not outperform top web sources for topics such as travel, where there are heavy incentives for experts to generate well structured content. We believe its performance is likely due to its aggregation of multiple expert viewpoints rather than particularly excellent writing or structure per se, never the less, the KA system showcased that the structures produced by Alloy can be synthesized into coherent articles that were useful for exploratory searchers.

3.5 Discussion

In this chapter, we took a step towards tackling the problem of clustering high-dimensional, short text collections by combining techniques from natural language processing and crowdsourcing. By using a two-phase process connected by a machine learning backbone, our proposed method compensates for the shortcomings of crowdsourcing (e.g., lack of context, noise) and machine learning (e.g., sparse data, lack of semantic understanding). As part of the system we introduced an approach aimed at providing greater context to workers by transforming their task from clustering fixed subsets of data to actively sampling and querying the entire dataset.

We presented three evaluations that suggest Alloy performed better and more consistently than automatic algorithms and a previous crowd method in accuracy with 28% of the cost (Exp.1), is robust to poor work with only 20 workers (Exp.2), and is general enough to support different types of input (Exp.3). Qualitatively, we noticed Alloy often produced better names for categories than machine algorithms would be capable of, including names not in the text (e.g., a cluster including items about *smart thermostats* and *solar panels* was named "*Home Improvements*" which was not in the actual text).

One potential concern might be whether Alloy's tasks take too long to be considered microtasks. While Alloy deploys HITs that take more than a few seconds to finish, we think they are still comparable to other complex microtask systems such as Soylent [14] and CrowdForge [96]. Specifically, based on a total of 281 HITs, the median run-time for the Head Cast HITs is 7.5 minutes (M=8,3, SD=4.1), for Merge Cast 8.3 minutes (M=16.2, SD=15.6), and for Tail Cast 11.4 minutes (M=13.2, SD=6.1). Despite having less workers doing longer tasks, Alloy performed consistently across different sets of workers on the same datasets.

During development, some assumptions, both explicitly and implicitly, were made about the input of the system: 1) there are more clips than categories. 2) the categories follow a long-tailed distribution. 3) clips belong to primarily one cluster. 4) there is a small set of gold-standard clusters. 5) workers can understand the content enough to cluster it. Note that we do not assume the crowdworkers can understand the semantics of the content, but just enough to identify ideas that are salient and common in the dataset. Thus they may be able to cluster complex topics such as machine learning without understanding those topics if enough relational context is embedded in the clips. For example, an abstract of a research paper may say "this paper uses POMDP machine learning approaches to cluster text", they might put it in a "clustering" cluster without knowing what a POMDP is.

One obvious limitation to our approach is clustering long documents. This is a common limitation for crowd-based systems that rely on workers reviewing multiple items for context (either from

random selection or active sampling). It becomes infeasible to fit multiple items in a single HIT if the length of each item is long. Another related limitation is organizing documents that describe multiple topics. Lab studies in a past work [97] showed that individuals are able to decompose long documents into short clips of single topics during information seeking tasks. One way to expand the proposed method to overcome the length limitation could be splitting documents into short snippets, either with the crowds or machine algorithms, and create topical clusters using Alloy.

Another limitation is organizing datasets that are inherently difficult to structure categorically. For example, concepts in Q3 (*planetary habitability*) have causal relationships without clear categorical boundaries (e.g., *distance to sun*, *temperature* and *liquid water*). As a result, all approaches had significant trouble, including low agreement between human annotators. On the other hand, some dataset can be organized categorically in multiple ways. In Q4 (*Barcelona*) we found that some categories fit a *place* schema (e.g., *Sitges*, *Girona*) while other categories fit a *type* schema (e.g., *museums*, *beaches*). One approach for addressing this could be trying to cluster workers to separate the different kinds of schemas; however, upon inspection we found that individual workers often gave mixtures of schemas. This interesting finding prompts further research to investigate what cognitive and design features may be causing this, and how to learn multiple schemas.

Looking forward, we identified a set of patterns that may be useful to system designers aiming to merge human and machine computation to solve problems that involve rich and complex sensemaking. The hierarchical clustering backbone we use to integrate judgments from a variety of crowdworker tasks allows us to *cast* for different types of crowd judgments and *gather* them into a coherent structure that iteratively gets better with more judgments. We also introduce useful new patterns for improving global context through self-selected *sampling* and keyword *searching*. One important consideration these patterns bring up is that while previous ML-based approaches to crowd clustering have focused on minimizing the number of judgments, we have found it is at least as important to support the rich context necessary for doing the task well and setting up conditions that are conducive for crowdworkers to induce meaningful structure from the data.

We hope the patterns described in this chapter can help researchers develop systems that make better use of human computation in different domains and for different purposes. For example, the *sample and search* pattern could potentially be adapted to support other tasks such as image clustering, where crowdworkers could use the sampling mechanism to get a sense of the variety of images in the dataset, highlight discriminative objects, and label images queried based on features extracted from the highlighted regions. Furthermore, the *cast and gather* pattern may provide a useful framework for combining crowds and computation that is both descriptive and generative. For example, Zensors [105], a crowd-based real-time video event detector, could be considered a form of the cast and gather pattern which uses a classification algorithm instead of a clustering algorithm as a backbone, and casts for human judgements whenever its accuracy falls below a threshold (e.g., if an environmental change lowers precision), with the classifier backbone retrained with the new human labels. While we used a clustering backbone in this work, future system designers might consider other machine learning backbones (e.g., classification or regression algorithms) for different tasks. Overall, we believe this approach takes a step towards solving complex cognitive tasks by enabling better global context for crowd workers and providing a flexible but structured framework for combining crowds and computation.

Chapter 4: SearchLens

Capturing and Composing Complex User Interests

This work was previously published in ACM IUI 2019 [34] and has been adapted for this document.

While previous chapters focused on providing general overviews to better orient searchers in the initial stages, this chapter explore a novel approach to better support the personalized and opportunistic aspects of the process. Using a restaurant review corpus, I focused on supporting users to learn from data and iteratively refine and evolve their nuanced interests. Consumer generated reviews are one of the most important influence in online decision making. To make sense of these rich repositories of diverse opinions, searchers need to sift through a large number of reviews to characterize each item based on aspects that they care about. We introduce a novel system, SearchLens, where searchers build up a collection of "Lenses" that reflect their different latent interests, and compose the Lenses to find relevant items across different contexts. Based on the Lenses, SearchLens generates personalized interfaces with visual explanations that promotes transparency and enables deeper exploration. While prior work found searchers may not wish to put in effort specifying their goals without immediate and sufficient benefits, results from a controlled lab study suggest that our approach incentivized participants to express their interests more richly than in a baseline condition, and a field study showed that participants found benefits in SearchLens while conducting their own tasks.

4.1 Introduction

People often rely on reading online reviews and forum posts to make predictions about how well different options might match their personal interests and needs. With the proliferation of online reviews, people now have instant access to millions of online reviews from people with varying perspectives and interests. It was estimated that in 2013 Amazon provided shoppers access to more than one million reviews for just their electronics section [120], and in 2016 Yelp provided around 250,000 reviews for over 6,000 restaurants for the city of Toronto alone [83]. Having access to this rich repository of diverse perspectives based on the past experiences of others has the potential to empower consumers to understand their choices thoroughly and make better decisions for themselves without being overly influenced by marketing and branding [46].

Unfortunately, it is often difficult for users to be able to quickly and efficiently match their personal interests to the large amount of information available for each potential option. One problem is that simple star ratings are often not sufficient, and recent research has shown reviews often play an important role in users online purchase decisions [58, 127]. For example, restaurants might receive negative reviews for its simple decor and lack of good ambiance, but some searchers might value more the authenticity of the food or whether vegan options were available on the menu. Subsequently, finding, reading, and evaluating relevant reviews is time-consuming and challenging. Users have to manually parse through the reviews for each restaurant and match them to their personal interests (e.g., kid friendly, authentic Indian cuisine). They then have to

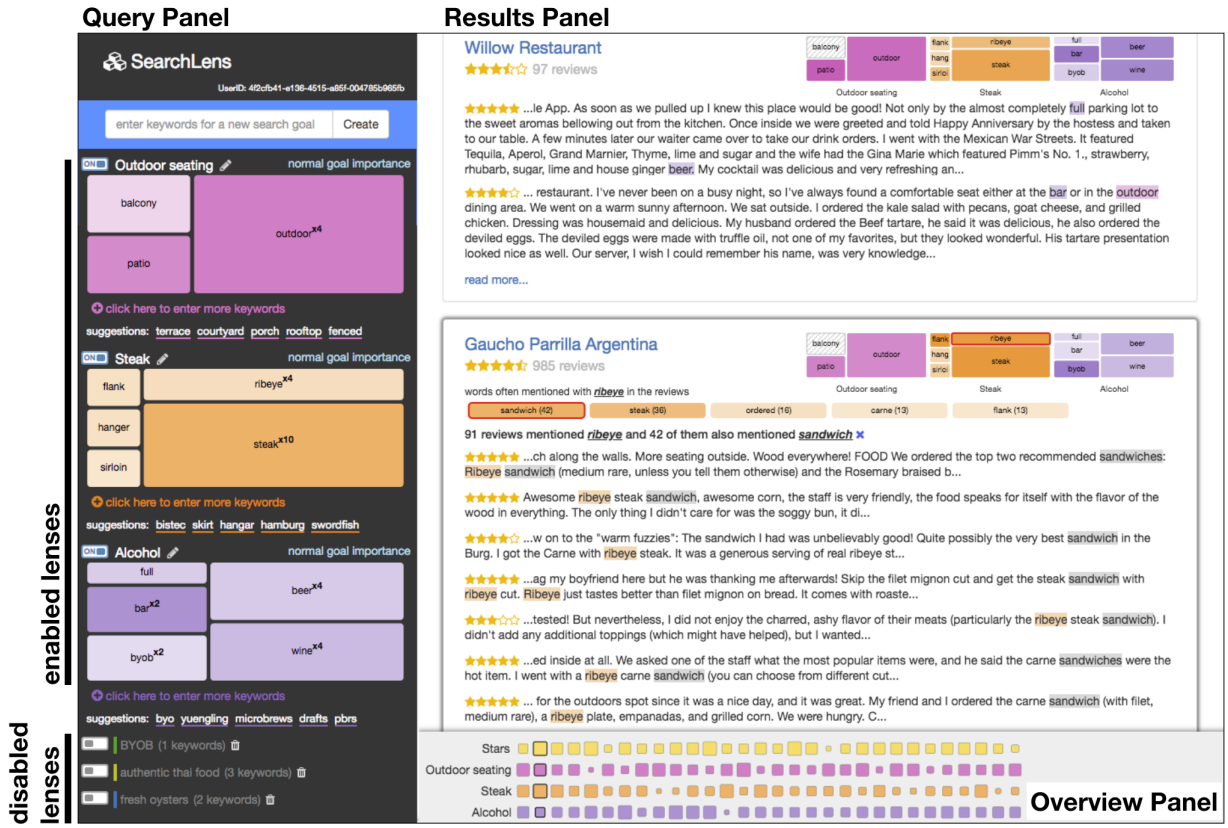

Figure 4.1: An overview of the SearchLens system. The Query Panel on the left allows users to specify search topics, or Lenses, by specifying multiple keywords. The keywords for a given Lens are show in colored cells sized by importance (weight). Lenses can be freely disabled or enabled for different scenarios. The Results Panel on the right shows a ranked list of search results that best match the enabled Lenses from the searcher. The same visualization for specifying queries are then used for explaining how each result matches with user's interests and mental model, and also serve as an interactive navigation for filtering mentions of specific keywords. The Overview Panel at the bottom shows a collapsed version of the cells that allows for quick comparison between results.

track which restaurant meets which criteria, and if they discover and add any additional criteria, they must back-fill that information and re-evaluate previously seen restaurants. Furthermore, once a user has finished searching, the work performed discovering and evaluating factors is lost, resulting in having to start from scratch even if a similar need arises in the future. For example, a traveler who has spent a lot of time choosing between ramen restaurants in Los Angeles must start from scratch evaluating ramen restaurants in Toronto, despite having discovered several important factors (e.g., thickness and chewiness of noodle, whether the broth is simmered for a long time with pork bones) that will be similarly utilized in their decision making.

Getting users to specify these nuanced interests and preferences has been a long standing challenge. Several decades of research have explored ways of getting users to externalize their interests [9, 87], for example by: using prompt and text field designs that promote longer query terms [10, 57], asking for relevance feedback on the results provided [132, 146, 147], or explicitly asking users to build up sets of query terms of different topics [75, 77]. There are two primary challenges brought up by this work. First, users have trouble specifying their interests, which includes challenges with identifying query terms that were neither too general nor too specific; providing more than a few terms (even when longer queries were more likely to lead to useful results); and learning terms from the content, rather than knowing them all beforehand [10, 147]. The other main issue found is that it is very difficult to get users to put in the work to externalize their interests, either as query terms or as explicit feedback, due to perceptions that the work will not be sufficiently paid off in the future or not understanding how their work will affect their results.

To tackle this issue of capturing, leveraging and exposing user interests, we introduce SearchLense, where users construct externalized representations of their interests as “Lenses”. Lenses are leveraged as an explanatory tool, providing users with a way to quickly parse, understand and make judgments based on the vast amount of review data instantaneously. Additionally, Lenses can be reused in different contexts and combined in different configurations. In the example above, imagine a system which could capture the factors that the traveler found important for ramen in Los Angeles and reuse them to quickly make a confident, personalized decision about ramen in Toronto. If traveling to Toronto with kids, a “kids” Lens might also be added with factors such as whether the restaurant typically has long lines and how many seats it has. These persistent Lenses could be useful in a variety of situations beyond reviews, ranging from academics keeping track of interesting research topics; travelers deciding which places to visit in an unfamiliar city; consumers deciding between products; lawyers doing case discovery; or voters tracking important issues. We explore this problem in the context of restaurant reviews, conducting a controlled lab study with 29 participants to examine if our visual interface for explanation and exploration is effective in providing immediate benefits to elicit rich interest expressions from the users. Additionally, we performed a three day field deployment study with 5 participants to explore the benefits of Lenses when users were conducting their own tasks. Results suggest that our prototype system SearchLens was able to learn richer representations of its users’ interests when compared to a baseline system by allowing users to fluidly capture, build, and refine Lenses to reflect their interests and needs, and that the user-generated interfaces can be reused over time and transfer across contexts.

4.2 Related Work

Past research has proposed a variety of approaches to collecting, modeling and leveraging users' interests and intents through both interface design and computation. Our work builds on this diverse of literature by allowing the system to learn the personal interests of the users through interaction to retrieve relevant data, and present data based on its understanding of the different users. This allows us to elicit structures that can be reused across different contexts and tasks and are more nuanced and personalized to each users when compared to traditional search structures such as search results clustering or pre-compiled facets.

4.2.1 Eliciting and Modeling Interests and Intents

A significant topic of research has been interfaces that can collect, explicitly or implicitly, the personal goals and interests of users as they search for information and modify their viewing of content correspondingly. While there is extensive literature on doing so in the context of personalized search and re-ranking of search results (e.g., [23, 24, 152, 156]), we focus here on work that enables more interactivity and transparency of users' interests to support more complex searching. One such thread lies in the collection of users' interests through keywords or interest vectors into an agent or user interest or intent model. This includes seminal work such as WebMate [38], which built up an agent composed of sets of TF-IDF [176] vectors to represent the user's different interests. Similar to WebMate, we aim to build collections of terms that represent the user's interests, but focus on explicit user selection of those sets, and making them explainable and composable. Interestingly, WebMate's "Trigger Pair Model" which looked at co-occurrence of words within a sliding window across a set of documents can be seen as a precursor to the word vector model that we use for keyword suggestions. More recent work in this vein includes user modeling of concepts, such as AdaptiveVIBE [2] and Intent Radar [132], which include two dimensional visualizations of documents and their relation to the user's inferred interests. Our work builds upon these but aims at increasing the richness of the structure, nuance, and specificity of the user's expression of interests. Specifically, our Lenses, composed of multiple keywords that can capture multiple levels of specificity, can be themselves composed into more complex expressions and reused across different contexts and tasks. We also focus on supporting users in the discovery process of building good terms that are discriminatory and explanatory.

4.2.2 Concept Discovery and Evolution

Research in interactive machine learning has also explored techniques to support data annotators or searchers in discovering and externalizing useful concepts when working in unfamiliar domains. For example, Alloy used a *sample-and-search* technique to categorize textual datasets with novice crowdworkers where they first explore the space of information through sampling items in the dataset to discover useful categories, then externalize each category using a set of query terms and search for other relevant items [32]. Past work has further suggested that the working concepts of an annotator may change over time as new items were examined [103]. Different techniques that can better support this concept evolution process were proposed, such as structured labeling [103], crowd collaboration [33], and interactive visualization [39]. These point to the importance of providing mechanisms that allow users to not only discover and define concepts based on data, but also to easily evolve their concept representations during the process

of exploring an unfamiliar domain. In a study more closely related to our work, CueFlik allowed image searchers to define conceptual filters (e.g., listing only *action shots* when searching for *baseball* images) by labeling items in a search result list as positive or negative training examples [56]. Previously defined filters are persisted and can be applied to future searches (e.g., applying the same *action shots* filter when searching for *football* images), but evolving existing conceptual filters would require recreating filters from scratch or re-labeling items in existing filters. Our work builds this past work to allow exploratory searchers in unfamiliar domains to discover concepts of interests from data and externalize these concepts in the form of "Lenses" that can be continually refined. Finally, the Lenses are persisted across different search sessions similar to [56], and can be modified and composed for different scenarios and goals.

4.3 System Design

The key motivating concept behind SearchLens was providing users with a way to externalize their complex interest profiles in a way that could be useful for ranking, explanation, and transference to different contexts. We aimed to make the interface simple and transparent but also powerful enough to express higher level, abstract concepts and differing levels of specificity. To do this, we introduce the idea of "Lenses": reusable collections of weighted keywords that contain "honest signals" of a user's interests that can be composed in different configurations to match a user's current needs. The Lenses that are enabled in a particular configuration drive various visualization and explanation elements to help the user understand how the information space meets their needs, and also whether they need to fix or reformulate their Lenses.

A key challenge here is incentivizing users to create rich Lenses by providing sufficient and immediate benefits. For this, SearchLens provides visual explanation of items in the search results based on users' Lenses, which also serves as an interface for deeper exploration. When a new Lens is created or enabled, its visual representation appears on the interface for each item, allowing users to understand how well each item matches with the Lens, and how frequently each keyword is mentioned in its reviews. To further explore each item, users can click on keywords in each Lens to see relevant reviews.

A typical use case is as follows. A user just moved to Pittsburgh and wants to go out to eat ramen. She starts by pulling up a restaurant she knows she likes from Toronto and goes through some of the reviews, noticing that the reviews of her favorite tonkatsu ramen mention interesting signals such as "bone" and "umami" and adds them to her ramen Lens along with other useful words such as "tonkatsu", "ramen", "bowl", etc. After checking to see that her Lens is bringing up other restaurants that serve ramen she likes in Toronto and adding a few of their terms to her Lens, she switches to Pittsburgh and looks for how her Lens is being used. She also activates her drinks Lens, which she's built up over time to incorporate her particular interests in unfiltered sakes as well as hoppy beers. Using the Lenses, she quickly see which ramen restaurants in the results list serve unfiltered sakes and/or hoppy beers. To further explore her different options, she can click on each keyword in her Lenses to filter relevant reviews. For example, "tonkatsu" might be often mentioned with "spicy" in one restaurant, and "creamy" in another, allowing her to further differentiate her options based on aspects that she cares about.

The following subsections describe the designs of the SearchLens system. We will first present our concept of "Lenses," and how users can use SearchLens to fluidly express and refine their different nuanced interests, and freely compose their Lenses for different contexts. We will also

describe how search Lenses can provide immediate benefits once specified, providing users visual explanation of each item in the search results, and also an interface for deeper exploration.

To test our prototype system in a realistic and manageable setting, we focused on the domain of restaurant reviews where personalization and searching with multiple goals is especially important. We used a subset of the dataset from the Yelp challenge [83] that included local business in 11 metropolitan areas.[1] Restaurants and reviews were selected by string matching on the *city* field of each restaurant available as metadata in the Yelp challenge dataset, resulting a subset of 48,485 restaurants and 2,577,298 reviews. This allows us to explore how user-specified Lenses can be composed and reused for different scenarios, as well as for the same scenario across different cities. In addition, we also use the same data to train a Word2Vec model [123] for generating Lens-specific query term suggestions.

4.3.1 Capturing User Interests with Lenses

Our goal was to develop a way to elicit users' interests which is both highly expressive and immediately beneficial. To explore the natural discovery and collection of users' interests we conducted a preliminary study in which we asked people to read reviews of their favorite restaurants on Yelp and see if they could identify terms that were good indications of their interests. We discovered that people found it intuitive to identify many different terms that matched their interests. Many of these terms were not simply general descriptors (e.g., "good", "tasty") but instead terms they considered indicative of matching their personal interests (e.g., an authentic ramen restaurant would include terms talking about the thickness of the noodles; a popular restaurant might be less favored if it also had very long lines). Terms also fell into different classes of factors users were interested in (e.g., service vs. food quality vs. parking). Users seemed to focus on finding reviews that mentioned these terms and use them in their decision making.

Based on these initial findings we developed a system for users to easily collect terms from reviews into "Lenses" and to use those terms to identify and summarize reviews that mentioned those terms. Similar to [75], we enable users to search with multiple Lenses at the same time. However, our Lenses differ from traditional search queries or faceted metadata in several important ways.

First, our system encourages the iterative development of Lenses as the user explores. A common activity in online exploratory search involves discovering new and interesting aspects from data. SearchLens aims to make it easy for users to add new Lenses and improve existing ones throughout their searching process. Users can create a new Lens by specifying a set of keywords using the text field in the Query Panel on the left (Figure 4.1). As users browse the results on the right, they might find some keywords in their Lenses were too general to be useful (e.g., "tasty broth"), and find discover more indicative keywords either from prior knowledge or from the reviews (e.g., "rich and thick broth"). In this case, users can refine their Lenses by adding new keywords using three different interactions, each for a different scenario. First, users can click on the plus icon under each Lenses to enter new keywords in a Lens specific text field. Second, as users discover more indicative keywords or new topics of interests from the reviews, they can highlight the keywords and use a context menu to add them to an existing Lens. In addition, a list of keyword suggestions are also listed under each Lens based on current keywords

[1] Pittsburgh, Charlotte, Phoenix, Las Vegas, Toronto, MontrÃľal, Mesa, Mississauga, Cleveland, Scottsdale, and Edinburgh.

(Figure 4.2). Users can hover over each suggestion to see example mentions, and click on the keyword include it. This allows users to assess the usefulness of the suggestions, such as to avoid ambiguous terms. The Lens-specific suggestions were computed based a word semantic model described in the below subsection. To remove a keyword, users can click on its cell and select remove keyword in the context menu.

Once constructed, Lenses can then be used to visually inspect and adjust their "projections" onto the data. Lenses are represented visually as boxes subdivided into cells, one for each term the user added. Initially, all keywords in the same Lens have equal importance (as reflected by being the same size), but users can click on each cell to select different importance in a context menu (x1, x2, x4, x10, exclude) to better reflect their personal preferences. The size of the cells will adjust accordingly to reflect the importance of each keyword (excluded keywords are represented using fixed size cells with a unique pattern fill). The shade of each cell shows the overall frequency of each keyword in the top 30 search results (Figure 4.1, Query Panel). This allows the user to get a sense of how items in the corpus reflect their mental representation of each topic. For example, a large cell with very light shade represents a concept that the users deemed as an important feature of the topic, but was rarely found in the results. Surfacing this information ensures user are aware of how useful each of their keywords are, and can refine their Lenses to include more indicative keywords.

As Lenses and terms are collected a user can over time build up a repository that reflects her personal interests. Each Lens can be disabled and re-enabled and are persisted across different visits to the SearchLens interface, with disabled Lenses are listed at the bottom of the Query Panel (Figure 4.1). Various combinations of Lenses can be activated depending on the goal and context. For example, for a date night a user might enable their personalized Lenses for "cozy and intimate", and "vegan", or for a weekday lunch activate their Lenses for "fast casual", "vegan", and "easy parking". Although our main thrust in this chapter is exploring the viability of this approach, further work will likely be needed to understand as Lenses accumulate how to scale them. For example, in the current prototype all disabled Lenses are shown, but future systems could further contextualize Lenses by inferring the task context (e.g., what type of item someone is searching for).

Keyword Suggestions

While creating a new Lens, listing all keywords from prior knowledge can be mentally taxing and have poor recall. To further reduce the required effort for building expressive Lenses, SearchLens generates Lens-specific keyword suggestions. As an example, when a user created an "Outdoor Seating" Lens with only three keywords ("outdoor", "patio", and "garden"), SearchLens automatically suggested relevant keywords including "balcony", "courtyard", and "terrace" (Figure 4.2). To do so, we trained a Word2Vec model [123] with 300 dimensions using the entire Yelp dataset of 2,577,298 reviews. The trained word model can project words onto a semantically meaningful vector space, which in turn allows for measuring semantic similarity between words. Alternatively, it can also be used to find a set of words that are semantically similar to a given term by searching in the vector space of nearby words. To generate Lens-specific keyword suggestions, we first project all its keywords in a Lens onto the vector space and calculate the average vector to obtain a list of similar terms around the average vector. To further increase the chance of presenting useful and discriminatory search terms, we only used terms that appeared more than 50 times in the corpus, were mentioned in reviews of more than three restaurants,

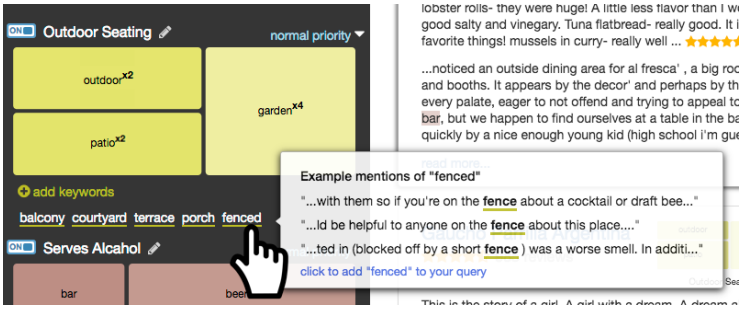

Figure 4.2: SearchLens provides keywords suggestions based on currently Lenses. Hovering shows a preview panel with mentions of the suggested keyword, allowing users to better understand the effect of adding the suggested keyword. In this case, SearchLens suggested balcony, terrace, fenced, and other keywords for the "Outdoor Seating" Lens. However, further inspection showed that fenced may not be a indicative keyword for the purpose of this Lens.

and were mentioned in less than 40% of all restaurants.

4.3.2 Interest-driven Explanation

Persistent, decoupled user interest models would be beneficial to users the long run by providing separate reusable and recomposable interests across multiple search sessions. However, without immediate and perceivable benefits, users typically are not willing to spent extra effort expressing their separate interests for future tasks. For this, SearchLens uses each user's Lenses to provide visual explanation of each item in the search results. This is based on our approach of allowing users to express their multiple topics of interest separately, which enables SearchLens to distinguish between keywords of different topics and opens the possibility of visualizing each result according to users' interests in easy-to-interpret ways. Explanation is especially important for supporting searching with multiple interests, as it can be difficult for the users to understand which interests and keywords were associated with each result. Consider traditional search interfaces that only offer a short snippet for each result as explanation. These short summaries provide little support for personalized interpretation beyond a few highlighted query terms and their context. Even if users listed keywords of many different topics at once, the linear result list also provides little information about each result beyond their overall relevance ranking.

One obvious approach to explaining items in the search results is to surface mentions and statistical information, such as mention frequencies, at the topic level. For example, [77] visualized the overall frequency of different search terms in different topics for each search result, and [75] visualized the mention locations of different topics within each document. Visualizing at the topic level allowed these systems to provide mechanisms for specifying many topics and keywords, while at the same time visualized deeper information about each result in a way that matches the mental model of the searchers. However, visualizing at the topic level can be prohibitive for keyword-level operations, such as query reformulation and assigning importance levels to different keywords based on their frequencies.

SearchLens supports rich explanation at the topic and keyword level through its user-specified Lenses. Explanation occurs by showing the each Lens visualization from the Query Panel (Figure 4.1) on each result and adjusting the term shading to correspond to the frequency of the

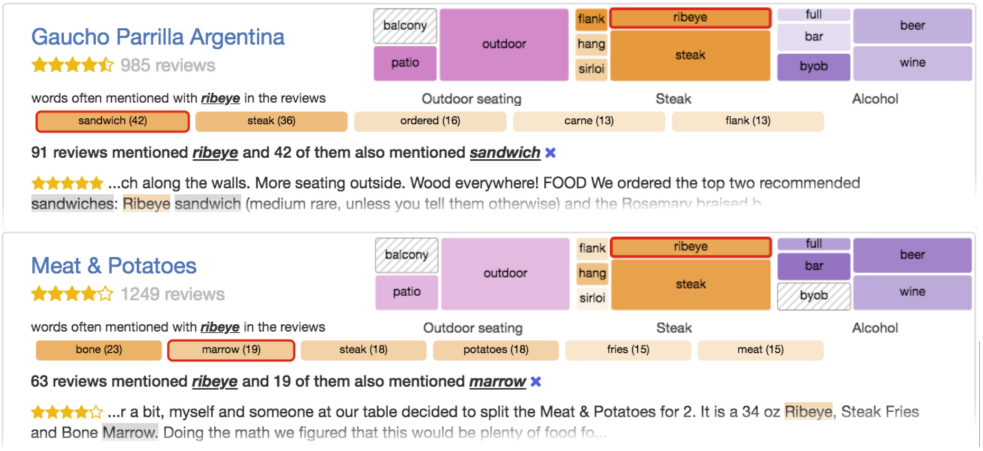

Figure 4.3: The visual explanation and exploration feature allows comparison of results at different levels of granularity using a familiar interface used for specifying queries - at the levels of Lenses, keywords, co-occurring terms, and mentions, allowing users to query with multiple Lenses at the same time, while still being able to comprehend how each result matches their different Lenses.

term within that search result (Figure 4.3). By using identical colors and layouts of each Lenses, and showing result-specific keyword frequencies, users can quickly interpret how each result matches with their different interests at both the topic and at the keyword level using a familiar visualization. As an example, Figure 4.3 shows how a user might examine two restaurants in a search result list using her Lenses for “Steak”, “Alcahol”, and “Outdoor Seating”. At the topic level, both restaurants matched well with her Steak Lens rendered in dark shades that incorporated her stronger preference for “ribeye” steak, and also also her other interests such as “flank” steaks. She can also see that the first restaurant matched her Outdoor Seating Lens better than the second one. Looking at the same Alcohol Lens at the keyword level, she can easily see that the two restaurants matched differently with her “Alcahol” Lens where the first one has many mentions of “byob” in the reviews and the second one with many mentions of “beer” and “bar” instead.

Finally, to provide a more compact, higher-level, topic-centric overview of all restaurants in the search results, SearchLens collapses the colored cells for each Lens into a single cell similar to [77]. The size of each cell to shows the overall frequencies of keywords in different Lenses for each result (Figure 4.1). This allows users to get a quick overview of restaurants in the search results, and compare different options at the topic level using the Overview Panel at the bottom.

4.3.3 Supporting Deeper Exploration of Items

In addition to acting as a visual explanation for each result, the cells in the visualization also act as a navigation tool for deep exploration at the keyword level. Users can explore mentions of different keywords by clicking on its corresponding cell and the summary will update in real-time to show a list of its mentions. In addition, the Lens also shows the top co-occurring words that were frequently mentioned near the selected keyword as overview and deeper navigation, a strategy found useful in exploratory scenarios by prior work [48, 49, 131]. As an example, Figure 4.3 shows the how the Lenses allow users to explore and compare options at different

levels of granularity. At the highest level, users can use the shading of different cells to see that the *Outdoor Seating* Lens has more mentions in the first restaurant (Figure 4.3). Searchers can use the shading of individual cells to compare options at the keyword level. For example, the term "BYOB" was frequently mentioned in reviews for the first restaurant, but did not show up in reviews for the second restaurant. Finally, clicking on the individual cells allows users to explore mentions of its corresponding keywords and words that were frequently mentioned together. For example, when exploring mention of the work "ribeye" for both restaurants, SearchLens shows that there were many mentions of "sandwich" near the word "ribeye" for the first restaurant, and many mentions of "bone marrow" near "ribeye" for the second restaurant (Figure 4.3).

4.3.4 Indexing and Ranking

Traditionally, faceted search systems typically combine factors from multiple facets for ranking using disjunctions (factors within facets, such as brands selected by the user on a shopping website) and conjuctions (factors between facets, such as brands and price ranges). In an early iteration of SearchLens, we tested using the Boolean OR operator between keywords within the same Lens, treating keywords within the same Lens as synonyms while ranking. However, users reported this approach lead them to restaurants that poorly reflected their Lenses, as some restaurants may have many mentions of few keywords in a Lens, but very few mentions of other keywords. Fundamentally, unlike faceted search systems, different keywords in the Lenses typically describe a criteria as a whole. For example, an authentic ramen Lens might contained keywords describing creamy bone broth and freshly made noodles. In this case, the different keywords combined represented what the user considered good ramen restaurants, instead of as alternate options in a facet (such as a set of preferred brands). In a later iteration, we switched to Okapi BM25 for ranking that used inverse document frequencies to weight keywords instead of eliciting importance rating from the users. However, users reported unable to construct Lenses that reflect their priorities and unable to construct expressive Lenses that lead to useful results. This lead to the current iteration where we used a modified version of the standard Okapi BM25 ranking function to combine keywords across Lenses [143], which by default considers both term frequency and document frequency to rank documents similar to TF-IDF ranking function, but also adjust for the length of each documents.

We modify the Okapi BM25 ranking function to account for the importance levels specified by users in the following ways. By default, Okapi BM25 uses the inverse document frequencies to weight each keywords, with the motivation that words appearing in many documents tend to be less important. Since in SearchLens users can specify keyword importance using the interactive visual explanation, we instead weight each keyword according to their user-specified importance level. By default, SearchLens assume each Lens is equally important, and normalizes the weights of keyword q in a Lenses ℓ in proportion to the user-specified importance level of all keywords $\acute{q}$ in search Lens ℓ:

$$weight(q) = \frac{importance(q)}{\sum_{\acute{q} \in \ell} importance(\acute{q})}$$

SearchLens then uses the normalized keyword weights in place of the inverse document frequency term in the Okapi BM25 ranking function, and the score of each document d in the corpus for a set of Lenses L is therefore:

$$score(d, L) = \sum_{\substack{\ell \in L \\ q \in \ell}} \frac{weight(q) * tf(d, q) * (k + 1)}{tf(d, q) + k * (1 - b + b * |d|/avgDL)}$$

where ℓ is the different user-specified Lenses, q is the different keywords in each Lens ℓ, $tf(d, q)$ is the term frequency of keyword q in document d, $|d|$ is length of the document d, and the constant $avgDL$ is the average document length in the corpus. We used the default parameters $k = 1.2, b = 0.75$ for Okapi BM25. Finally, we sum up the score of each Lens weighted by a coordination factor, which is the proportion of keywords in a Lens that has a non-zero document frequency. This modified version of the Okapi BM25 function can be easily translated to SQL queries for standard relational databases, or as a custom ranking function for the popular open sourced document retrieval engine Apache Lucene. This allows the SearchLens interface to be easily implemented using readily available tools that were already optimized for scaling and computational efficiency. Admittedly, more sophisticated ranking approaches may further improve the quality of results, but this simple method allowed us to explore the costs and benefits of providing reusable, re-composable, explanation-centric Lenses to users.

4.3.5 Implementation Notes

The backend of SearchLens was implemented in Python, using NLTK [17] and gensim [142] for indexing and word semantic model, respectively. In the indexing phase, text in each review is lowercased, tokenized, and stemmed using the Word Punkt Tokenizer [90] and Porter Stemmer [169]. Stop words are filtered out. An inverted index that records the document and the offsets of the mentions of each word stems is computed and stored in a PostgreSQL relational database. The Flask Python framework was used for our HTTP server. We implemented front-end of the SearchLens prototype as a web-based system using Javascript (ES6) and the ReactJS GUI framework, and the interactive visualizations are implemented using the D3.js library. User-specified Lenses were stored on client-side using browser cookies, so that they are persistent for the searchers between multiple visits.

4.4 Evaluation

We evaluated SearchLens in two studies. First, we conducted a usability study in a controlled lab environment. Using predefined tasks, we tested the usefulness and usability of the system, as well as whether the visual explanation and exploration features provide enough benefit to encourage participants to express their rich and multifarious interests. Second, we conducted a field deployment study where participants use SearchLens for their own tasks. This allowed us to explore the benefits and limitations of our reusable and re-composable Lenses in real-life scenarios.

4.4.1 Usability Study

The main goal of the usability study was to verify in a controlled lab environment the usability of the interface and whether the visual explanation and exploration features can provide benefits to encourage users to express their nuanced and multifarious interests. We considered these the preconditions for conducting a field deployment study to test the real-life benefits of reusable and re-composable Lenses. Therefore, we focused on the following:

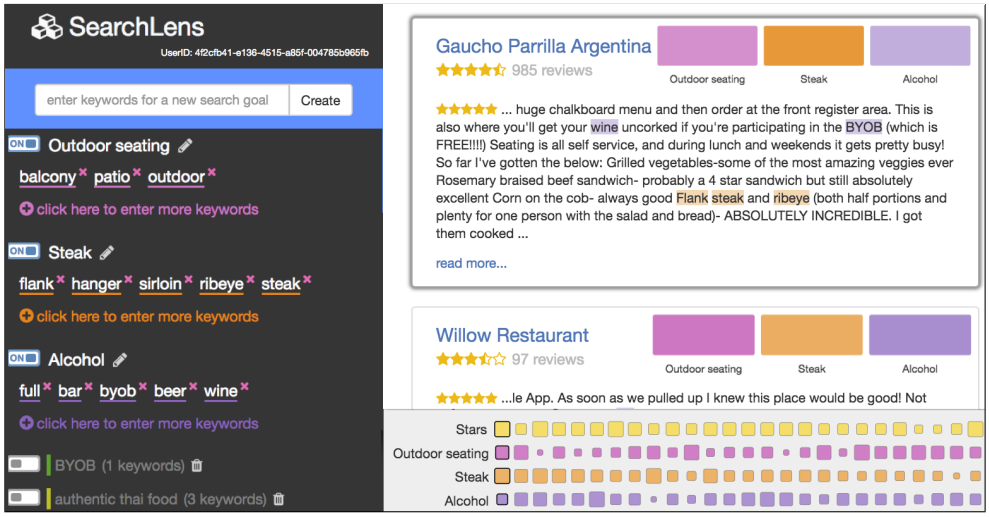

Figure 4.4: A Baseline system with topic-level visual explanation by collapsing the colored cells in each Lens and visualizing results only at the topic level.

- whether the interface encouraged participants to externalize multiple interests and structure them using Lenses
- whether participants found the visual explanation and exploration feature to be useful
- whether the added benefits of visual explanation and exploration encouraged participants to spend more effort to express, iterate, and refine their Lenses

To test the above, we compared SearchLens to a baseline interface as a between subject condition, where the detailed visual explanation and exploration features were removed by collapsing the colored cells in each Lens and visualizing results only at the topic level (Figure 4.4), resulting an interface similar to the TileBars and the HotMap systems [75, 77]. Unlike in the SearchLens condition, users can only explore each restaurants at the topic-level, but not at the individual keyword level. Since searchers can not assign importance levels for each keyword in the baseline interface, we used the standard Okapi BM25 ranking function that weights keywords based on inverted document frequencies [143]. We chose this baseline as a more conservative test of the interactive explanation features than, for example, a comparison to Yelp or other search query-driven site (which are the implicit comparisons for the field study below).

The three scenarios for the usability study are listed below. The first scenario was designed to have both clear criteria (nice decor and good atmosphere and serves beer or wine), and an exploratory aspect (find a specific type of Japanese restaurant based on your own preferences). Scenarios 2 and 3 were designed to explore whether users would be able to reuse their Lenses for different contexts and find value in doing so. Scenario 2 had overlapping criteria to Scenario 1 (serves beer, cocktails, or wine), and Scenario 3 involved performing an identical search to Scenario 1 but in a different city.

- **Scenario 1**: Stanley is in Pittsburgh, USA visiting some friends and he is in charge of finding a few good restaurants for the group. They are interested in Japanese restaurants. They're not familiar with Japanese food or the different types of Japanese restaurants, so it is up to you to find Japanese restaurants based on reading the reviews and your personal

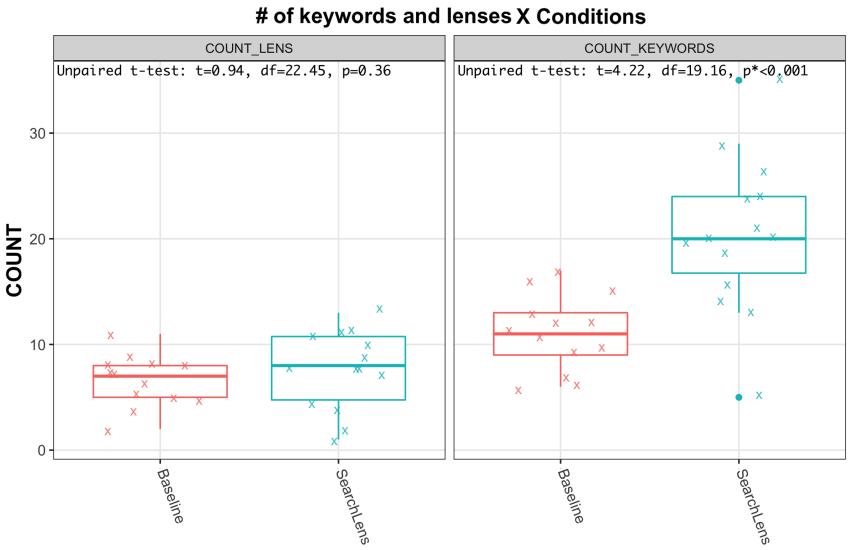

Figure 4.5: Number of Lenses and keywords saved by each participants at the end of the study. Participants in both conditions created comparable number of search Lenses, but participants in the SearchLens condition collected significantly more keywords in their Lenses.

preferences. The restaurants should have a nice decor and good atmosphere. Some of his friends like to have a few drinks with their meal, so if the place has a bar that serves beer or wine it would also be great. Since its pretty nice out, it would also be nice if the restaurants has outdoor seating or a patio, too.

- **Scenario 2**: John is looking for good seafood restaurants in Pittsburgh, USA, particularly places that serves fresh oysters and has a bar that serves beer, cocktails or wine. Decor or atmosphere are not important, but big plus if they offer outdoor seating, for example, a patio. Some of his friends are allergic to seafood, so the place must also have non-seafood options, preferably steak.
- **Scenario 3**: (Same as Scenario 1 but for finding restaurants in Montreal, Canada instead of in Pittsburgh, USA.)

A total 29 participants were recruited from a local participant pool, where 14 participants were randomly assigned the SearchLens interface with three predefined search tasks (N=14, Age=18-61, M=28.1, SD=12.7, 7 male, 6 female, and 1 other/not listed), and 15 participants assigned the baseline interface with the same search tasks (N=15, Age=18-54, M=28.1, SD=10.7, 7 male, 7 female, and 1 other/not listed). Each participant was given 60 minutes to complete the study and was compensated 10 USD. Before conducting the three tasks, participants watched a five minute introduction video that described the features in their given interfaces, which is followed a step-by-step training where participants created two pre-defined Lenses, report the name of the third restaurant in their search results, and report which keyword is missing from its reviews. Participants finished the training steps using an average of 5.9 minutes (N=29, SD=3.8). For the main task, participants were told to spend 10 to 15 minutes on each of the three tasks listed above in order. Finally, participants answered a short post-survey where we collected their subjective opinions about the systems using 7-point Likert scales and free-form responses.

Action	Lab Baseline	Lab SearchLens	Field SearchLens
add terms by typing	3.67 σ=2.82	5.50 σ=4.86	7.00 σ=5.39
add from suggestions	n/a	1.57 σ=1.99	3.20 σ=1.79
add from reviews	n/a	0.29 σ=0.73	0.40 σ=0.55
total add actions	3.67 σ=2.82	7.36 σ=6.10	10.60 σ=3.71
remove a keyword	4.67 σ=4.27	3.50 σ=2.79	4.20 σ=2.68
adjust weights	n/a	8.93 σ=7.54	12.80 σ=7.89
	N=15	N=14	N=5

Table 4.1: Mean statistics for number of Lens editing actions performed by participants. Participants used SearchLens in the lab study more frequently add keywords to refine Lenses compared to baseline ($t(27)=2.12$, $p<0.05$). Participants in the field study conducted their own tasks.

Results for the Usability Study

One of our key hypotheses was that the immediate visual explanation provided by Lenses would encourage participants to express their interests and continually collect and refine those interests throughout the search process. This hypothesis appears to have been validated by the data. On average, participants in the SearchLens condition saved 20.43 keywords across their Lenses (N=14, SD=7.33), significantly more than participants in the baseline condition who saved 11.15 keywords (N=15, SD=3.58; $t(27)=4.12$, $p<0.001$). Importantly, this difference is likely not attributable to different perceptions of the task across conditions, as in both the SearchLens and baseline conditions participants generally created one Lens for each task criteria and combined multiple Lenses for each task (e.g., decor, drinks) and there was no difference between the total number of Lenses created between conditions (SearchLens: 7.6, baseline: 6.5; $t(27)=0.92$, $p=0.36$). In other words, the term-based interactive visual affordances supported by SearchLens seemed to encourage people to collect more terms indicative of their interests.

This pattern appeared to hold true throughout the search process for the iterative refinement of Lenses as well (Table 4.1). On average, participants using SearchLens added keywords to existing Lenses 7.4 times (N=14, SD=6.1) while those in the baseline condition did so 3.7 times (N=15, SD=2.8), which was found to be a significant difference ($t(27)=2.12$, $p<0.05$). This suggests that the added benefits from the visual explanation and exploration feature encouraged participants to iteratively refine their Lenses and allowed them to discover useful keywords more often.

We also examined whether participants found the added visual exploration features to be useful, and how the added benefits affected their behavior. By examining the behavior logs, we found participants using SearchLens frequently use the visual exploration feature. On average, each participant clicked on 25.86 (SD=29.19) keywords to filter reviews that mention a specific keyword instead of sifting through reviews to find ones that mentioned it (Figure 4.6). In both conditions, participants can also click on the name of a restaurant to see a list of reviews ranked by all active Lenses. While there is suggestive evidence that the filtering of reviews led to less use of the generic review lists, the result was not significant based on the number of participants in the study (M=6.33, 3.07; SD=5.78, 5.92; $t(27)=1.50$; $p=0.15$).

These results suggest SearchLens allowed participants to maintain a broader search goal with multiple interests, while at the same time explore and compare different options at a finer-grain

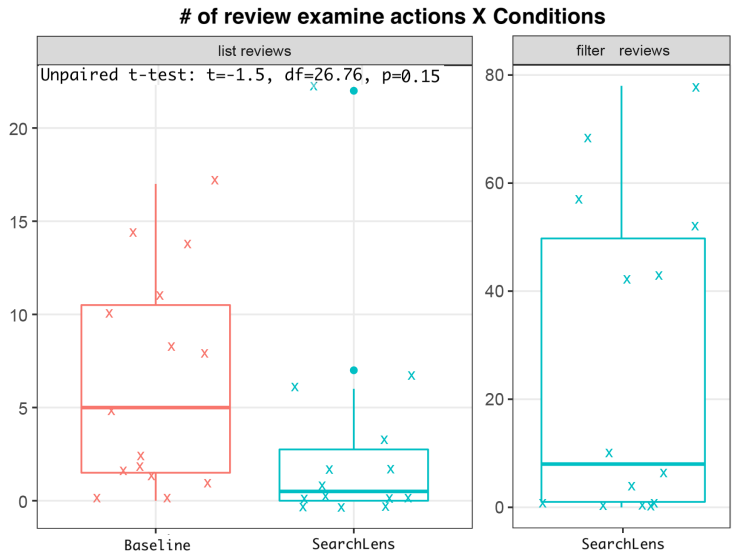

Figure 4.6: Participants in the SearchLens condition were less likely to read through unfiltered lists of reviews than the baseline condition, which was accompanied by increased use of the SearchLens-specific ability to filter reviews relevant to different keywords.

level interactively instead of sifting through the reviews of each restaurant.

4.4.2 Field Study

Our field deployment study aimed to test our idea of reusable and re-composable Lenses in real-world settings. Five participants were recruited from the first study based on their high self-reported interest in researching restaurants online and in participating in a follow up study (N=5, Age=18, 20, 22, 23, and 25, 4 male, and 1 others/not listed). The participants were given access to the SearchLens system via the internet, and were asked to use the system for at least 60 minutes in total over a three day period. Although they were free to choose from any of the 11 cities in the dataset for this study, all five participants conducted tasks for their current city. Afterwards, they return to the lab and were given 45 minutes to finish a survey with primarily free-form questions, and were interviewed for another 15 minutes. Each participant was compensated with 40 USD for finishing the study.

Participants created more Lens keywords when conducting their own tasks comparing to participants in the lab study (Figure 4.7). On average, participants in the field study created 13.40 (SD=3.65) Lenses, significantly more than participants in the lab study that created 7.64 Lenses (SD=6.54; $t(17)=2.46$, $p<0.05$). They also saved significantly more keywords than participants in the lab study (lab: 20.4, field: 30.0, $t(17)=2.50$, $p<0.05$). Admittedly, it can be difficult to measure how much time participants actually spent using SearchLens in the field, nevertheless, results suggest that participants were able to accumulate more interests Lenses over a three day period than participants who spent 60 minutes in the lab study.

All five participants conducted multiple tasks during the study. Many explored different types of restaurants that they liked in the city using multiple Lenses, using SearchLens to build *"an overview interface for restaurants in the city that I might like"* (P1, P3, P4, P5). Participants also

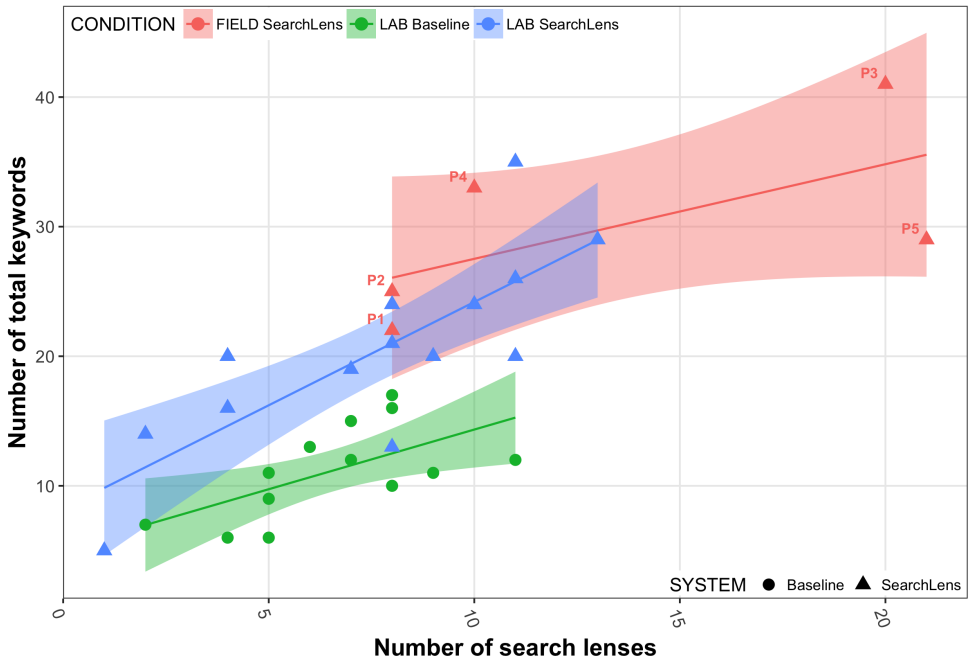

Figure 4.7: Number of Lenses and keywords specified by participants under different conditions. In the lab study with predefined search tasks, participants using SearchLens (blue) created a similar number of Lenses but used more keywords than the baseline condition (green). Participants in the field study (red) conducted their own tasks.

had more specific goals, including to check if there are vegan restaurants she has not discovered yet (P5), restaurants that serve bubble tea (P2), pizza places that offer Chicago deep dish-styled pizza (P3), and Mexican restaurants that has vegan options on the menu (P2).

Refining Lenses

While participants reported creating Lenses based primarily on prior knowledge, all five participants also reported refining their Lenses throughout the process. Several cited that the shaded cells of the visual explanation helped them quickly noticed some keywords were too uncommon, and that an important concept of interest was missing from the search results (P1, P2, P5). One also mentioned noticing and removing ambiguous keywords when using the mention filtering features (P4). Participants also learned about new keywords which they added to their Lenses, sometimes replacing existing keywords, from both the suggestions (P1, P2, P3, P5) and from the reviews (P1, P2). Interestingly, the behavioral logs (Table 4.1) suggest they frequently discovered them from the systems' suggestions, indicating the value of the word2vec approach which we initially were concerned about for being noisy. This also points to potential future work in auto-suggesting Lenses which we intentionally avoided here due to concerns about agency and explainability.

Breadth and Depth

Participants created both general, breadth-oriented Lenses and more specific, depth-oriented Lenses. P4 specifically mentioned that it was useful being able to search for different genre (i.e.,

American, Mexican, or Indian restaurants) and at the same time pay attention to very specific dishes (i.e., cheese steak sandwich made with chicken), while still being able to see how each result match with different things, citing that "*more specific things are hard to search for on Yelp.*" Alternatively, P3 presented an interesting use case for deeper exploration of a specific genre, by first creating an more general Indian Food Lens, and then creating multiple more specific Lenses describing specific dishes from different regions of India, generating an overview of different styles of Indian restaurants in the city. This suggests that some users may want to create higher level groups of Lenses

Reusing Lenses: Combinations and Task Resumption

Participants reported their strategies for how they reused their Lenses, which can be broken down into two non-exclusive categories. The first use case we observed was task resumption between multiple search sessions (P1, P3, P4). Participants described having the ability to switch to a different sets of Lenses yet still keep the original Lenses for the future being useful (P3). One participant (P1) searched with a single Lens most of the time, but still cited that being able to re-enable Lenses from past sessions and to continue work on previous tasks and refined restaurants being useful. For the second use case, participants mentioned reusing Lenses in combination with other Lenses (P2, P3, P5). When asked about which of their Lenses were used in combination with different other Lenses, participants reported Lenses that concerned style and environment (*Cute and Quirky* (P5), *Atmosphere and Vibe* (P2, P5), *Friendly Staff* (P3)), price (*Inexpensive* (P2, P3), *Large Portion* (P3)), and some food-related but not for a general genre (*Fresh* (P2), *Fast Casual* (P2), *Vegan Options* (P2, P5), *Strong Beer* (P3)).

4.4.3 Overall Usefulness and Other Usecases

Through the lab and the field studies, we found evidence that using user-generated Lenses to provide visual explanation for deeper exploration was beneficial and effective in incentivizing users to externalize and iteratively refine their interests using Lenses. This occurred throughout the search process almost twice as frequently when compared to participants in the baseline condition which did not include the visual explanation and exploration features (Figure 4.4). As a result, participants using SearchLens created richer Lenses with nearly double the number of keywords on average compared to participants in the baseline condition. Participants also frequently used the visual explanation feature to explore the individual items in their search results, filtering reviews using different keywords in their Lenses 25.9 times on average. To test SearchLens in real-world settings, participants in the field study conducted their own tasks, and provided insights into their strategies in building and refining Lenses, as well as their strategies of composing and reusing Lenses across context and across search sessions over a three day period.

From the field study interviews, three out of the five participants said that they actually found and saved interesting restaurants during the study, and intend to visit those restaurant in the near future (P1, P3, P4). P1 in particular went to one of the restaurants he discovered using SearchLens and was happy about the visit, and P3 used SearchLens to complete a previous task, saying "*I wanted to try deep dish pizza for some time since I moved to US. Finally found one near the city. Kudos!*" All participant expressed that they would be interested in using SearchLens in the future if available, many also cited other scenario that might benefit from SearchLens. P2 pointed to scenarios where he needed to "*find a place for many people that may*

want different things", and mentioned that SearchLens would be useful when her family visits her soon for his graduation. These results suggest that SearchLens was effective at helping users effectively find items that matched their specific interests.

4.5 Limitations and Future Work

One limitation of the current implementation of SearchLens is its lack of ability to filter restaurants using their metadata, such as geographic location. We intentionally did not expose this information to our participants so we can focus our studies on allowing them to build personalized Lenses. However, practical systems would likely combine both paradigms to maximize efficiency. Utilizing metadata can also augment user-defined Lenses, for example, taking into account whether the a review that matched a specific Lens was positive or negative and whether the review poster's interests matched with the user's personal interests. However, the interactions between the two paradigms would require further studies. On the other hand, utilizing existing techniques for query term generalization beyond stemming or lemmatization, such as synonyms, semantic word models, or query expansion, can potentially improve recall, but their effects on the visual explanations would also require further studies.

Another obvious limitation of SearchLens it that it required more user effort upfront in order to receive the benefits provided by the system, such as reuse, explanation, and exploration. On a 7-point Likert scale, most participants from our lab study responded favorably in the post-survey to this trade-off with 64% agreed or strongly agreed that SearchLens is an improvement to the traditional search interfaces, and another 21% somewhat agreed with the statement, however, the long-term effect remained to be seen. One way to extend SearchLens is to combine machine learning and information retrieval approaches to reduce the effort of building Lenses, such as building interest profiles automatically, or using collaborative filtering and query expansion for expanding or inferring Lenses automatically [3, 148, 178], or word-sense disambiguation techniques for resolving ambiguous keywords [180].

Alternatively, we could also explore ways to allow users to share their Lenses with each other through explicit or implicit collaborations. For example, one participant mentioned "*It would be nice if I can see what Lenses a local person would use if I'm traveling, because I always try to ask the locals about where I should eat.*" Allowing access to Lenses created by previous users or expert users could potentially enable expertise transfer and accumulation through continuing refinement of a set of Lenses. For example, locals and past travelers could iteratively curate a set of Lenses that leads to an interactive and explorable list of local specialties for future travelers.

Another promising direction is to more deeply explore the idea of user-generated interest profiles and how they could dynamically influence the different interfaces accessible to the user or interacting with users in more proactive ways. Since we asked the field study participants to use SearchLens for their own tasks, most participants searched for restaurants in the city they lived in. Some participants that conducted more targeted search tasks (P2, P3, P5) mentioned that they were already familiar with most of the options in the city that fits their goals, but would still occasionally search online to see if there were new restaurants that match their interests (P2, P5). As users continue to use SearchLens, the system will accumulate more understanding of what the users is interested in, and can potentially detect and notify the users of new information that might be of interests with high accuracy [179]. Alternatively, existing users may use their repository of Lenses to explore or curate the restaurants in an unfamiliar city. Participants in

the field study also pointed to the potential of Lenses being useful for other types of information and domain, including shopping (P2, P3), trip planning (P2, P5), buying a house (P2), and job hunting (P4).

In this chapter we introduced SearchLens, a novel approach that allows users to specify and maintain their profile of multifarious and idiosyncratic interests. This enabled them to reuse and re-compose their different interests across scenarios, as well as maintaining context across multiple search sessions. To encourage users to put in the up-front effort of curating Lenses, we explored ways of using Lenses to provide immediate benefits of visual explanation and deeper exploration of search results. Across a lab and field study we observed that participants expressed their interests with significantly more query terms, and found benefits in the SearchLens approach, including being able to transfer and reuse their Lenses across contexts, being able to interpret new information that reflects their own personal interests with transparency, and working at multiple levels of specificity and hierarchy. More fundamentally, being able to visualize and explore new information in ways that promote transparency can potentially empower users to be more aware of their online information diet. For example, as a way to manipulate their own social media feeds, and being more aware of how posts were selected or hidden. We believe SearchLens represents a first step towards a transparent and user-centered approach to addressing subjective and fragmented nature of information today.

Chapter 5: Intentionally Uncertain Highlighting for Foraging during Exploratory Search

`This work was previously published in ACM UIST 2016 [30] and has been adapted for this document.`

From a better understanding of the information space (chapter 3) to developing personal preferences and nuanced interests (chapter 4), users eventually use this understanding to collect and structure evidence to help them make better decisions. However, saving information can be cognitively costly in exploratory search scenarios because when exploring and learning the boundaries of what text may be relevant and useful later are themselves uncertain for the user. On mobile devices, this could also be physically challenging due to the small screen and font sizes combined with the inaccuracy of finger based touch screens makes it time consuming and stressful for people to select and save text for future use. In contrast to previous approaches which focused on speeding up the selection process by making the identification of hard boundaries faster, we introduce the idea of intentionally supporting uncertain input in the context of saving information during complex reading and information exploration. We embody this idea in a system that uses force touch and fuzzy bounding boxes along with posthoc expandable context to support identifying and saving information in an intentionally uncertain way on mobile devices. In a two part user study we find that this approach reduced selection time and was preferred by participants over the default system text selection method.

5.1 Introduction

Capturing information online for later use can be especially challenging during exploratory search tasks [25]. Studies of information foraging [134] and active reading [126] have identified the importance of collecting snippets of information while exploring and reading from multiple sources for comparison, cross-referencing, and structuring [1, 94, 97, 163]. As reading and learning in-

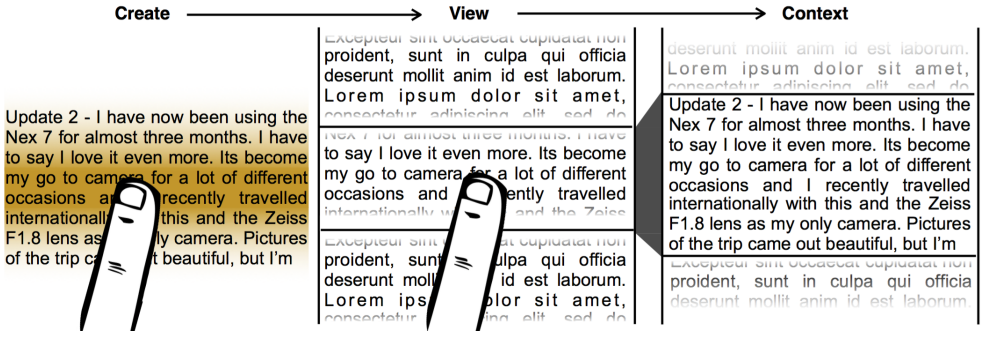

Figure 5.1: Highlighting with intentionally fuzzy boundary and a viewer that supports resolving uncertainty.

creasingly moves from handwritten notes and highlighted pages into web search and browsing, tools for supporting the curation and storage of online information have grown in popularity. For example, one well known tool for extracting snippets of information from web pages – Evernote Web Clipper – had over 4 million users on the Chrome desktop browser alone[1].

The need for reading and capturing information has expanded beyond the desk to domains where mobile devices are prevalent, such as in bed or at the kitchen table [1, 163]. Despite this need, identifying and saving snippets of textual information remains challenging on mobile devices. Small screens and font sizes combined with the inaccuracy of touch interfaces make selecting and saving text both time consuming and stressful. To understand the prevalence of text highlighting scenarios on mobile devices today, we conducted a survey with 153 participants (age 20-59, mean 32, 60% male, 76.5% from the U.S.) asking for their experiences with complex exploratory searches [116] on smartphones. Our results suggest that people frequently conduct complex searches either partly (70%) or completely (45%) on their phones. When asked about what makes these searches difficult, near half agreed that "*Selecting part of a webpage and save it*" is either moderately or extremely difficult, and 41% thought it would be valuable to have a better interface for it.

Approaches to improving capture interfaces have, to date, focused on improving the speed and accuracy of specifying the start and end boundaries of the selection area. Such approaches include using bezel or multi-push gestures [37, 68, 145], autocomplete [188], switching windows faster [35, 54], or leveraging the structure of the content to be copied [16, 84, 159]. These approaches are well suited for fast and simple copy-paste needs, such as copying an address from one application and pasting it in another.

However, in many learning and exploration tasks, people are uncertain about which and how much information to save. Early in the process, before a user has a good sense of the topic space, they might save information that later turns out to be irrelevant, or they may be uncertain about how much information on a particular page may be needed in the future [97]. For example, a researcher might extract a particular finding from a paper early in their exploration process, only to realize later that they also need the author's statistical model from the following paragraph. Conversely, over-selecting text that does not prove to be useful later can lead to additional effort in sifting out useful information from extraneous chaff (for example, in the limit if the entire page was selected and saved then the user would have to do all the filtering again). Furthermore, forcing a user to choose hard selection boundaries requires them to carefully predict their future information need, which can involve high cognitive effort [155]. Indeed, in a pilot survey, we found 11 out of 19 participants had trouble in the past identifying how much text to highlight. Additionally, 13 out of 19 mentioned that they had needed to return to a document to read additional text in order to understand the selections they created in the past. These findings suggest that these considerations are commonly encountered. Adding to the challenge, interactions for gathering information while reading need to be quick and low effort, otherwise people tend not to capture information in the first place [76, 118, 164].

In this chapter, we introduce and explore the concept of intentionally supporting uncertain input in the context of selecting and saving information during information exploration on mobile devices. We investigate the idea that in contrast to more defined selection tasks (such as copying an address or phone number), precise selection may not be the most appropriate interaction

[1] http://chrome.google.com/webstore/

paradigm for complex learning and reading tasks. In doing so we build on previous approaches that support fuzzy input, encourage lower granularity selection (e.g., lines of text vs. characters), or defer action until later [76, 97, 104, 149, 151, 164] In contrast to approaches which take uncertain input and maintain its uncertainty to be resolved later (e.g., [151]) in which the user intention is certain but the input is not, we suggest that there are cases in which the user intention is itself uncertain and resolving user input is inappropriate. This approach frees us to consider alternate ways to support selecting and saving information, especially on mobile devices where selecting and saving can be challenging for users.

Specifically, we explore two ways in which we can design for intentionally uncertain input. One is to support uncertainty in the selection interface through a fuzzy bounding box. This allows a user to feel less stressed about exactly where the boundaries of their selection lie and may reduce the need for careful prediction of their future information need. Another way is to support uncertainty in retrieval by saving context around the selection area and surfacing it later. We explore the tradeoffs of these two approaches and their interaction through a controlled experimental study. Furthermore, we introduce the idea of using pressure-sensitive touch as a new interaction approach for specifying selection boundaries. Pressure is particularly interesting as a modality because it has the potential to allow the fast selection of an area of interest while relaxing the cognitive and physical constraints of needing to specify exactly what should be saved. One potential drawback of such an approach is missing the information needed later due to inexact boundary selection, which motivated the idea of expandable context when reviewing snippets later. In the rest of the chapter, we will first describe the specific design of a system that embodies intentionally uncertain input in both selection (through pressure-sensitive touch selection) and retrieval of information. We then describe a two-part user study in which we investigate the performance of the two types of interfaces for a low-uncertainty task (targeted copy-paste) and a high-uncertainty task (exploratory search).

5.2 System Design

In this section, we introduce a new highlighting interaction that supports intentionally uncertain selection. Our aim with this technique is to reduce the stress and increase the efficiency of saving information for future use while exploring and learning new information. The explain this interaction, we break up the process of highlighting into three separate steps: initiating selection mode, indicating the start and end points, and saving the selected text. In the remainder of this section, we will give a high level overview of the proposed highlight interaction, and then discuss the different design options we explored for each step in the interaction.

5.2.1 Selection Interface

The proposed interaction is composed of three steps, which align with the above mentioned process (Figure 5.2):

1. Users initiate selection mode with a pressure (force) touch on the general area of interest.
2. Once selection mode is enabled, they then estimate the amount of context needed in the future by controlling their force while moving their fingers vertically to fine-tune the start and end points.
3. Finally, they swipe horizontally to confirm and save the information.

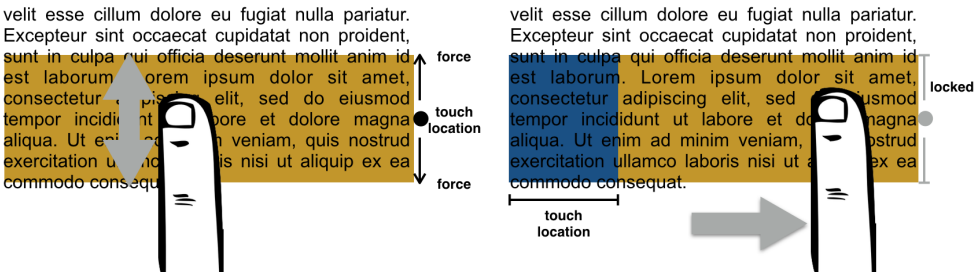

Figure 5.2: Selecting and saving text using force to set the selection area (left) then sliding right to lock it.

We explored three options for initiating text selection mode. One way is to use a simple touch gesture to start text selection, and swipe your finger across select text. This is analogous to using a highlighter pen to highlight text on papers. However, on a smartphone this may conflict with a number of gestures in reading mode, such as scrolling or canceling a tap on a hyperlink. The second option is to use the time dimension to initiate text selection to avoid conflicts with the reading mode gestures, such as tapping and holding at the same location for 500 milliseconds on iOS and Android devices. However, this will add an inherent cost to every highlight the user creates. 500 milliseconds might seem like a low cost if text selection is rarely needed. However, when people engage in complex sensemaking tasks, such as exploratory search, they often have the need to save pieces of information frequently in a short period of time [165]. The third option is to use the force dimension, which is beginning to appear in mass market products, to trigger text selection. This has the benefits of having no conflict with existing reading mode gestures and virtually no time delay. Consequently, we choose the third option for initiating our text selection phase.

In designing the interaction for the selection mode, we explored four options for indicating the start and the end points. The first option is to use two draggable handles for indicating the start and the end positions, and the second option is to use the initial touch location as the start point and the release location as the end point. Both approaches are used by many current touch systems, but both suffer from the inaccuracy of finger based touchscreens (minimal target area of 44 by 44 pixels) and the small font size (default of 17 points, or 34 pixels on iOS), making it difficult to physically pin-point the intended characters. Further, the finger view-blocking problem makes it difficult to for user to do fine-grained adjustments. To avoid having the users tap on exact words or characters, we instead chose the third option, which takes the touch coordinates as the center of the selection and uses the amount of force to determine the size of the selection. We explored two sub-options for adjusting selection range based on the amount of force. We first tested using force to adjust selection range at the character or word level. However, it was difficult to keep track of the start and the end points at the same time, especially near the beginning and the end of each line. The second option is to use force to adjust how many lines are selected. This way, the start and the end boundaries have the same vertical distance to the touch location, and was much easier to keep track of at the same time when adjusting the amount of force used. We tested mapping the same amount of force used to the same number of pixel height and number of lines highlighted according to the font size of the page. The second option made the system more consistent on pages with different font sizes, and also aligns better with our design goal of correlating force with the amount of context required by the user.

Finally, for saving the selection, we explored two options. The first is to have users quickly release their fingers from the touchscreen to save the selected text. Our pilot studies showed this option to be intuitive, and often what the users try first. However, in practice this approach proved challenging as it made capturing the right selection range ambiguous, since the force dimension is also used to control the range of the selection. Similarly, in our lab study some participants encountered similar issues with the built-in text selection, and often accidentally moved the handles when releasing their fingers from the screen. Instead, in our approach users move their finger horizontally across the screen and then release to save the selected text. By using a new dimension, we reduce the chance of accidentally changing the selected range when leaving the selection mode. To reduce the number of dimensions the users need to control, we lock the selection range (both the center location and the size) once the user begins to swipe their finger horizontally.

Previous work has shown conducting gestures in the force touched state can be laborious [107]. However, in our design, we only make use of the Y dimension movements during the selection mode, and we lock both the Y dimension and the force dimension when the user starts moving their fingers horizontally in the saving mode (Figure 5.2). User studies showed that participants were both able to efficiently create highlights using the proposed mechanisms, and prefer using the proposed interaction over the built-in text selection feature with draggable handles for highlighting information during a complex sensemaking task.

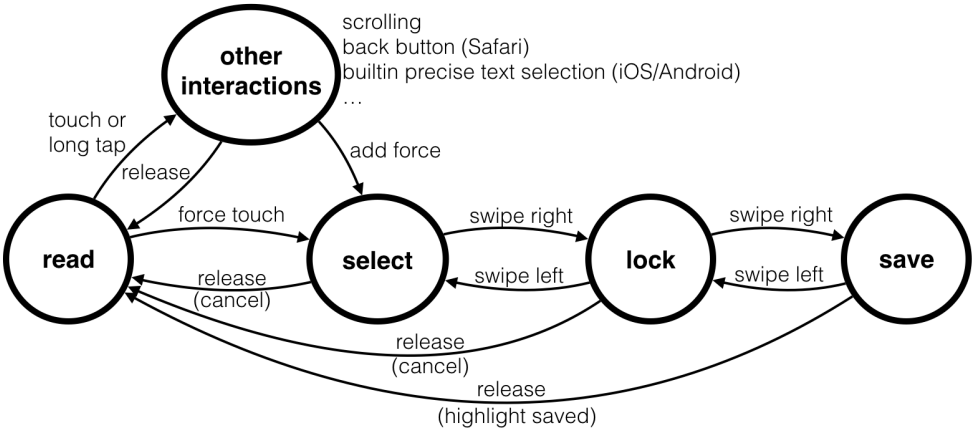

Figure 5.3: State transition diagram.

Figure 5.3 shows the states and transitions of the proposed interaction. Notice that the proposed interaction does not interfere with common reading interactions, such as vertical scrolling or horizontal swiping (backwards and forwards buttons in Safari). In addition, the proposed interaction can also co-exist with common precise text selection methods (both commercial and academic) that are initiated with long taps or edge taps [37]. We will discuss about supporting multiple selection methods in the discussion section.

5.2.2 Intentionally Uncertain Boundary

We explored designing for uncertainty in two complementary ways: through a fuzzy boundary during highlighting, and through an expandable context during review. To explore uncertain input as a design consideration for highlighting, we introduced a fuzzy boundary in the selection

	Study 1			Study 2 (with context)			Study 2 (no context)		
	Hard	Fuzzy	System	Hard	Fuzzy	System	Hard	Fuzzy	System
Mental	7.43	6.10	9.05	6.78	5.89	8.33	7.50	5.90	10.50
Physical	7.57	6.00	9.05	8.11	5.56	10.22	6.10	3.90	9.20
Temporal	7.90	7.10	9.52	8.22	6.56	11.00	6.50	6.60	9.80
Performance	14.10	15.00	13.43	12.44	14.00	12.11	12.20	15.30	14.30
Effort	8.86	6.76	11.00	7.78	5.00	9.33	6.20	6.10	10.80
Frustration	7.43	5.19	11.52	7.00	4.00	11.00	5.10	4.80	8.90
Overall (0-100)	39.20	31.10	50.14	37.89	27.00	49.89	31.40	27.30	49.20
			N=24			N=9			N=10

Table 5.1: Average NASA TLX scores for three highlighting modes from part 1 of the lab study: Targeted highlighting (left), reading and highlighting with an expandable viewer (middle), and without an expandable viewer (right). Higher numbers mean higher workload or higher performance.

mode (Figure 5.1). By intentionally hiding the hard boundaries from the user, we hope to free them from engaging in the difficult task of determining exactly how much context they will need when creating highlights, and postpone uncertainty resolution until the users review the saved information with a better idea of how much context they need. To achieve this, whenever the user creates a new highlight, the system will also save its surrounding text as context. To give users dynamic access to the context when reviewing, we made a simple highlight list interface that allows the users to use force touch gesture to expand the viewport and request for more context. The idea is that knowing they will have the chance to adjust the amount of context for each highlight during the review process, it will reduce both the cognitive stress and physical interaction load of creating highlights with exact boundaries.

5.3 User Study

We conducted a two-part lab study to evaluate three highlighting methods for saving information during exploratory search for later use: force touch with hard boundary, force touch with fuzzy boundary, and system selection. The first part of the study tested the overall interaction workload without the cognitive demands of exploratory search, while the second part added simulated exploratory search behaviors. In the first part, individuals were given articles and asked to highlight random portions selected by the system for 20 minutes. In addition to collecting data, this served to train participants to use the three modes efficiently. In the second part of the study, participants were given a complex sensemaking task involving reading multiple articles and creating their own highlights. Afterwards, they reviewed their highlights using either an interface which showed expanded context around their initial selection or that only included their original selection, and wrote a short summary integrating the content of the articles.

We implemented the proposed technique on an iPhone 6s Plus running iOS 9.3.3 through a custom native app that uses the standard WebKit browser for the reader interface. Force touch highlighting was implemented in Javascript by accessing pressure sensor data through WebKit APIs and injected into the WebKit reader using Cocoa APIs.

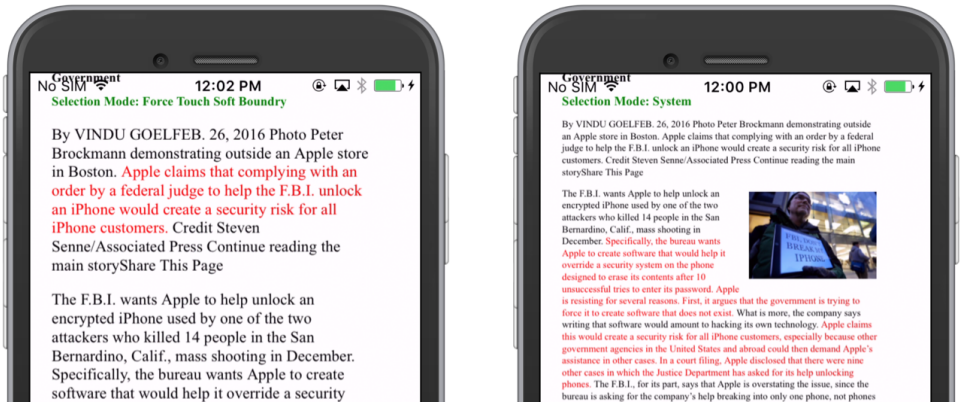

Figure 5.4: Examples from Part 1 of the lab study, where participants were asked to create highlights covering the red lines. Pages varied in conditions including font size and page layout.

5.3.1 Demographic

We recruited 24 participants from a local behavioral research participant pool. Participants ranged in age from 18 to 59. The majority of the participants were either undergraduate or graduate students, 11 female and 13 male. Participants were required to be fluent in English and use a smartphone as their main mobile phone. Based on self-reporting, 14 were Android users and 10 iPhone users.

5.3.2 Lab Study Part 1: Training and Interaction Cost

In the first part of the study, we evaluated the interaction cost of three highlighting methods: force touch with hard boundary, force touch with fuzzy boundary, and system selection. To remove the effects of prior knowledge and the cognitive demands associated with learning new information, the system marked random portions of article in red and asked participants to only highlight the red sentences without actually reading the article. Participants were required to highlight the sentences completely without highlighting surrounding sentences in order to proceed.

Before the study began, individuals filled out a pre-survey for demographic information and how they currently used text selection or highlighting on their smartphones. During the study, participants were given a minimum of 24 pages (8 for each mode) in random order. On each page there were four highlight targets (32 for each mode). We also randomized the font sizes (30px, 38px, 47px), page layout (with/without photos), and location and size of the targets (3-8 lines). If they finished highlighting the 24 pages under twenty minutes, more pages with random conditions were provided. Afterwards, participants filled out a NASA TLX survey for each highlight mode [69] to measure cognitive load.

Results

Table 5.2 shows the pre-survey responses from 24 participants about their smartphone text selection habits and opinions. The results show that many users find it frustrating and time consuming to use the text selection feature, and they are unable to do it efficiently or frequently. However, 22 out of the 24 participants agree that they would copy or highlight text more often if it was easier to do so. This suggests a strong need for saving information for future use on

Question	Mean[.95CI]
I find it frustrating to select text on my smartphone	5.71[5.19, 6.23]*
I find it time consuming to select text on my smartphone	5.50[4.84, 6.16]*
I can select text efficiently on my smartphone	3.17[2.50, 3.83]
I often copy and paste text on my smartphone	3.92[3.09, 4.74]*
I often highlight text on my smartphone	3.17[2.50, 3.83]
I would copy or highlight text more often if it was easier to do on my smartphone	5.79[5.26, 6.32]*

Table 5.2: Self-reported text selection and highlighting habits on a 7-point likert scale. A higher score indicates stronger agreement. N=24

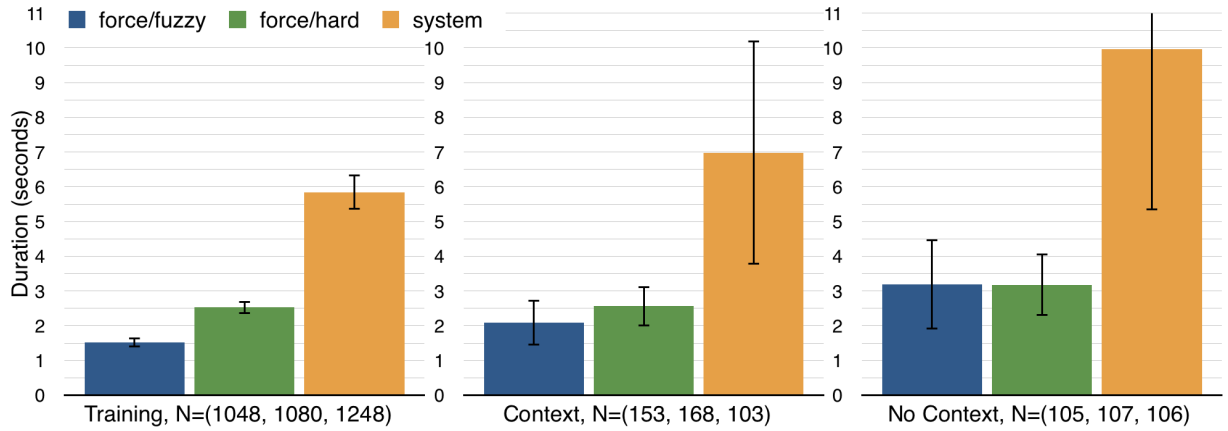

Figure 5.5: Average time spent creating highlights in each condition: training in part 1 (left), the context condition in part 2 of the study (middle), and the no context condition in part 3 of the study (right)

smartphone devices, and the lack of an efficient method to fulfill this need.

When asked about the reasons for copying and pasting on their smartphones, 63% of the participants reported copying text for later use with note taking apps or emailing themselves, and 58% reported copying text to share information with friends via social networks, emails, or text messages. For none textual copying and pasting, 54% of the participants reported sharing URLs with friends, and 50% reported saving URLs for themselves. Finally, 21% of the participants reported to not use the copy and paste feature.

Based on 1048 samples for force highlight with fuzzy boundary, 1080 samples for force highlight with hard boundary, and 1248 samples for system selection, the average time to create highlights using the three modes were 1.80 seconds, 2.77 seconds, and 6.06 seconds, respectively (Figure 5.5). We analyzed these results using an ANOVA model, where duration was found to be significant different between the three conditions ($F(2,20) = 18.0$, $p < 0.001$). Using a Fisher LSD means difference test, we found the soft boundary selection mode was the fastest, with the hard boundary mode being slightly slower, and system selection being the slowest ($p < 0.001$). Note that these times reflect the constraint that users were not able to proceed without accurate highlighting. No significant differences were found on NASA TLX measures. In the next study, we look at how this new highlighting technique perform when users are actively engaged with the content through a complex sensemaking task.

Question	Mean[.95CI]
Soft boundary makes it less stressful to create highlights than the other two modes	5.89[5.25, 6.54]*
Hard boundary makes it less stressful to highlight comparing soft boundary	3.00[2.25, 3.75]
The system selection mode is less stressful to create highlights	2.05[1.51, 2.60]*
Its fun to use the force touch with hard boundary mode to create highlights	3.58[2.77, 4.39]
Its fun to use the force touch with soft boundary mode to create highlights	5.32[4.61, 6.02]*

Table 5.3: Survey question about certain and uncertain boundary using a 7-point likert scale. A higher score indicates stronger agreement. N=19

5.3.3 Lab Study Part 2: Exploratory Information Seeking

In the second part of the study, individuals were asked to highlight important information while researching a new topic, and to write a short summary using the highlights they created. Half of the participants were given the reviewing interface in which they could expanding context surrounding their original highlights.

First, participants completed a pre-survey about their experiences and opinions about saving information for future use on mobile devices. Before they started on the main task, individuals were given six highlights we created from the Planet Habitability entry from Wikipedia and asked to write a short summary of the six highlights in five minutes. This was to ensure the participants in the context condition were aware that they could resolve uncertainty when reviewing their highlights when they wrote the summary. Finally for the main task, participants were given three pages to read, each containing two Amazon reviews for a different camera. Participants were told they had 15 minutes to read and highlight each source, and all three sources were of similar length, so they should spend roughly five minutes on each page. Each page required the participants to use a different method to create highlights; both the pages and modes were given in random order. After 15 minutes, participants were given 10 minutes to review their highlights, rank the three cameras, and explain their reasoning. The instructions were as follows:

"You have a friend who is looking to buy a new camera for taking pictures of his/her young kids at birthday parties. Using the highlights you saved, rank the three cameras, and write a short summary to explain to your friend how and why you ranked them this way."

After writing the summary, individuals answered a NASA TLX survey for each highlight mode according to when they were created their own highlights on the camera articles, as well as a questionnaire about the three highlighting modes and highlighting information in general.

Results

A total of 19 individuals participated in the part 2 of the study, where 10 of them were given the highlight review interface with that supports expanding viewport for more context, and 9 of them were given a highlight review interface with static viewports. All of the 19 individuals also participated in part 1 of the study, and had twenty minutes training of the three highlighting modes. Using a 7-point likert scale, user reported strong preference over having a uncertain input for highlighting during the complex sensemaking task of ranking digital cameras. On average, participants agrees (5.89/7.00) that the force mode with fuzzy boundary makes it less stressful comparing to force mode with hard boundary and the system selection mode, and find (5.32/7.00) using the force touch with fuzzy boundary mode to be fun (Table 5.3). When asked about which of the three highlighting mode they would use in the future if they need to read arti-

cles and learn new things on their phone, 15 of the 19 participants chose force touch with fuzzy boundary, 4 chose force touch with hard boundary, and 0 chose the system selection feature. In both conditions, only two participants chose the force highlighting mode with hard boundary, suggesting that even without the a way to resolve uncertainty when reviewing the saved information, some participants still prefer uncertain input while selecting and saving information. Below are representative quotes from participants about the soft boundary force touch interface:

“The soft boundary took a bit of getting used to, but once I got the hang of it it made things go a lot faster. It took away the pressure of getting the exact lines right, and let my intuition take over about how much needed to be highlighted ”

“The soft boundary is my favorite, because it is the least physically taxing, least mentally taxing and if I’m going to highlight in an article it just has to be generally around what I want not perfect, so this was my favorite.”

“I hated how exact you had to be with the hard boundary. It was just a huge pain. The soft boundary is so much better. I never realized how much I hated the generic copy and paste mechanisms. ”

We also asked the participants to fill out a NASA TLX survey for each of the three modes after writing the summary. To understand the workload effects, we utilized a generalized linear model to evaluate the differences between context vs no context, and the different modes of highlighting. In the results of the model, context was not found to be a significant factor ($F(1, 17) = 0.07$, $p = 0.789$), however the highlighting mode was ($F(2, 17) = 11.25$, $p < 0.01$). Additionally, was no interaction effect between the level of context provided and the highlighting condition ($F(2, 17) = 0.29$, $p = 0.749$). In order to understand the difference between the three highlighting modes, we ran a Fisher LSD means difference test, and found both the force touch soft ($t(17) = 4.66$, $p < 0.01$) and the force touch hard ($t(17) = 3.10$, $p < 0.01$) conditions to be significantly easier at $p < 0.01$ than the system selection feature. There was no difference ($t(17) = -1.56$, $p = 0.128$) between the two force touch highlighting conditions.

5.4 Discussion

We investigated the idea of interfaces that support intentionally uncertain input in the context of complex reading and sensemaking tasks, where precise input may be undesirable because it mismatches the high level of uncertainty in users’ understanding of the topic space, leading to stress and poorer selections. To do so we developed a system in which users could highlight information on a mobile phone using force touch, and manipulated whether the boundary was hard or soft. We also manipulated whether when they reviewed their highlights they could query for additional context around their highlights. Through a two-part user study we discovered that participants strongly preferred the force touch interaction technique with the soft, fuzzy boundary over the hard line boundary and over the default system selection with hard character boundary. We also found that both force touch highlighting approaches resulted in significantly faster selection speeds than the default system text selection.

While the force touch approach showed benefits in terms of speed, ease of use, and user preference, we also consider the drawbacks of such a solution. We initially hypothesized that people would not like the fuzzy force touch approach if they could not query for additional context later. However, only one participant mentioned this as a negative:

"I did like the Soft Boundary better because I found it a lot less frustrating, but in the end, I think it is not as practical as the Hard Boundary just because of the accuracy level. It didn't take as much work, but I was also worried if I was able to include everything I wanted to include in the highlight."

Instead, it seemed that most users appeared to adjust for the soft boundary, for example by oversampling the selected text, and were not bothered even when they were not able to adjust context posthoc. This suggests that the soft boundary interface may be useful even with existing review interfaces that do not support posthoc adjustment, but also lead to concerns that the proposed technique promotes mass highlighting, which past work has suggested to have negative effects on learning [133]. By examining the highlights participants created in the second part of the study, we found the proportion of highlighted words did not significantly differ across conditions with absolute numbers trending against the hypothesis: On average, participants highlighted 27% of the camera reviews under the force touch with hard boundary condition, 20% with the system selection, and 19% with force touch with fuzzy boundary, suggesting the fuzzy boundary did not encourage mass highlighting. We analyzed these the highlighted information proportions with a generalized linear model, and did not find the highlighting condition ($F(2,13) = 2.39$, $p = 0.111$), context condition ($F(1,13) = 0.51$, $p = 0.487$), nor their interaction ($F(2,13) = 0.38$, $p = 0.687$) to cause any significant variation in the amount of text highlighted.

Another non-optimal case for this approach is when the task is a strict copy-paste task in which hard boundaries are important, for example copying a telephone number or an address. Therefore, supporting both scenarios the same devices seems crucial for the proposed technique to be practical, and past work has also pointed to benefits of supporting both fine-grained and coarse-grained manipulations with fully-engaged interactions and casual interactions [137]. We believe there are at least two possible interaction paradigms for supporting multiple selection interactions at the same time that can be explored in future work: 1. *Independent sets*. The proposed method does not conflict with many existing academic and commercial techniques (e.g,. edges of a screen[37] or long taps) which do not currently make use of the pressure dimension. Thus each method could be implemented and the user could choose which to use given their current need. 2. *Sequential sets*. The two approaches could be combined by performing a fuzzy selection first and then allowing the users to switch to a precise selection mode to further adjust the boundaries (e.g., using force to select an area including an address, which brings up optional handles to trim off text around the address).

A final limitation we will discuss is the size of the selection area, which is currently fixed to a maximum limit. One participant mentioned this as a concern, stating:

"I really liked both of the force touch modes but at times I felt that the maximum force touch highlight box size was not large enough."

Informally, we noticed that we were able to select reasonably accurately when testing the system with larger size selections than explored in the study. However, it is possible that with a sufficiently large selection jittering from finger tremors or inaccuracy may become problematic. Exploring smoothing and transformation functions from the pressure input to match human cognitive expectations and physical capabilities is a fruitful area of future work. Furthermore, there is an interesting edge case when the selection consists of the entire screen, and whether users consider this a phase transition that should mean the entire page should be saved or simply the highlighted area. Appropriately addressing this concern is something that future work will be

need to answer.

Although we have focused here on the particular use case of highlighting information on mobile devices, it is possible that the idea of supporting intentionally uncertain input may have broader implications. The most obvious inference is for information exploration on desktops: although mice or pens as pointing devices make selecting much easier, the cognitive uncertainty of where the boundaries should be drawn remains. There may be other kinds of tasks in which uncertain input may be supported better as well. For example, many applications and operating systems require files and folders to be named as soon as they are created, which can lead to inconsistencies between the name generated early on and what the contents of the file and folder end up being later, with resulting problems in refinding and organizing that information. More generally, we believe that uncertainty in user input should in the future be treated as a design feature, not only a limitation.

Chapter 6: Fusion

Entity-Centric Foraging across Webpages in the Browser

As people search online to plan trips or shop for new products, they encounter many entities (e.g., attractions, restaurants, camera models) and collect evidence across information sources to make informed decisions (e.g., travel blogs, top ten lists). Current browsers treat entities and evidence on each webpage independently of other pages, making it difficult for users to keep track of what they are interested in and why. We introduce Fusion, a novel browser add-on that weaves pages together through common entities using an in situ interface integrated into users' existing browsing experience. When users open a webpage, Fusion "infuses" it with information extracted from other webpages and knowledge bases relevant to entities on the current page. When users save notes about an entity, their notes are "diffused" across other pages where the same entity was mentioned. In evaluation, we found our participants valued the entity-centric approach that helped them gather, reuse, and accumulate evidence across webpages for multiple entities. Our findings have implications for the design of future browser interfaces.

6.1 Introduction

Whether planning a trip to a new city, figuring out which camera to purchase, or researching the different treatments for a medical issue, learning and searching for information online has become the most common way that people make sense of the world today [115, 134]. People spend a significant amount of time exploring available options and gathering evidence about them that are scattered across multiple webpages in order to make informed decisions. Estimates suggest that up to 33% of the time spent online, or, as of 2009, more than 24 billion hours per year in the US alone, are spent doing this type of aggregation and synthesis [4, 92, 115, 144]. Consider for example the task of planning a trip: there may be hundreds of possible restaurants to dine at, attractions to see, and places to stay, each with corresponding evidence about its suitability for an individual's goals and preferences. Evidence about each of these options is often spread out across multiple search results, such as Yelp or TripAdvisor reviews, top ten lists, travel blogs, forum posts, and travel guides. These webpages typically contain sets of overlapping options along with subjective evidence and past experiences about them. In order to compare different options, users need to go through a large number of webpages and cross-reference between them to synthesize evidence about each potential option. However, this process of intense cross-referencing and note taking for sensemaking across webpages can be disruptive during reading and consuming information [15, 119, 128, 164], and is poorly supported by current browser and note taking interfaces. As the amount of user-generated content on the internet grows, supporting users in fully benefiting from this large repository of rich evidence is likely to become increasingly important [58, 127].

Due to the ubiquity of entity-related web queries in online sensemaking tasks [61, 112], one way that search systems have tried to support the above process has been by focusing on entity-centric approaches that present information about relevant entities in the search interfaces. For example, entity cards with rich attributes for entity-bearing search queries [21, 124], lists of

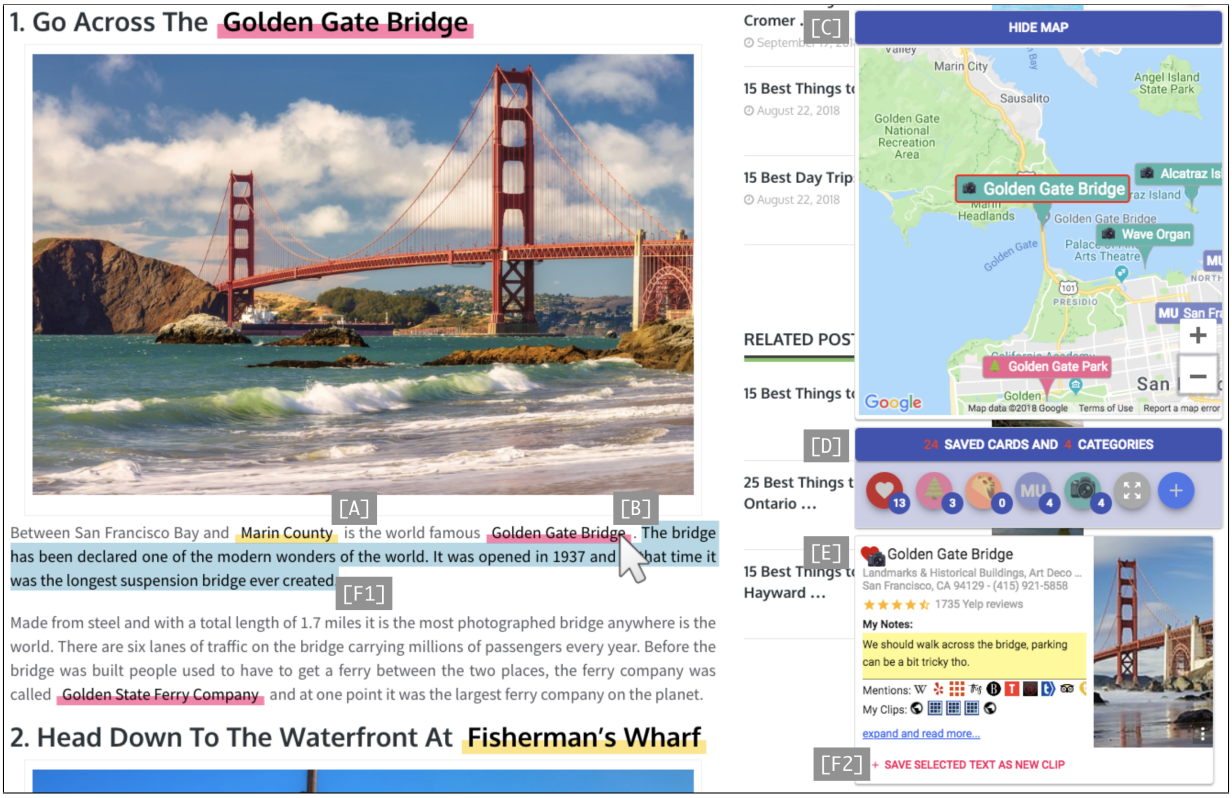

Figure 6.1: An overview of the Fusion browser add-on. Fusion identifies and highlights entity mentions on webpages [A,B] to indicate additional information is available. Highlights in red [B] indicates users had previously interact with the entity. Hovering an mention [B] brings out its corresponding entity card [E] as an overlay, with relevant information "infused" from other pages and knowledge bases as *mentions*. Users can also save notes or selected sentences [F1] to a card as *clips* [F2]. Saved clips are then automatically "diffused" to other webpages that mentioned the same entity. Users can also create categories [D] and drag the card under them. Finally, the Map view [C] shows its location in context of previously saved entities.

important entities to be used as subsequent queries (such as listing actors when searching about a movie) [18, 20, 98], or factual attributes about an entity or relationships between entities (such as the population of a city) [8, 40, 62]. While these approaches can efficiently provide factual and structured information about entities in search interfaces (such as figuring out the location of a restaurant), in many cases people still depend on examining and synthesizing the unstructured and descriptive information scattered across multiple webpages opened in their browsers. This is especially true when comparing and making decisions about many entities in a complex task (such as making an expensive purchase). In these cases, there is no single objective answer that can be surfaced in a search results page directly.

Although most prior work on entities have focused on providing objective information focused around a single entity during retrieval [21, 124], we posit that entities can also be useful for more complex exploratory search tasks by acting as a substrate connecting different information sources and the user's mental model. Leveraging them has the potential to enable deeper interactions with unstructured online information by focusing on meaningful concepts rather than webpages. Furthermore, recent advances in entity linking algorithms have been particularly promising in bringing the ability to better understand web content to the browser interfaces where users read and learn from individual webpages. For example, leveraging common entities mentioned across webpages to provide a sensemaking structure for users conducting complex exploratory searches and foraging across multiple webpages.

In this paper, we explore a new paradigm for interacting with unstructured and potentially subjective evidence about entities while reading and foraging from webpages in an exploratory search task. Since the user's personal evaluation of subjective information and how it meets their goal is critical in this situation, our design goal was to help the user to see scattered evidence about an entity in one place while also attaching personal notes and web clippings. These together served as a way to build up an external mental model and track search progress.

To investigate our entity-centric approach, we developed a prototype browser add-on called Fusion. Fusion allows users to keep track of the information scattered across multiple sources by "infusing" evidence about an entity from other webpages to the webpage the user is currently reading. It also "diffuses" users' thoughts about different entities across webpages where the same entities are mentioned for future reference and to accumulate more evidence. In our user study, we tested how participants utilized our entity-centric approaches while conducting complex exploratory search tasks, focusing on whether Fusion allowed them explore, gather, reuse, and accumulate evidence about entity options across multiple webpages. The primary domain on which we aimed to test Fusion was a travel planning task with 20 participants. In addition, we also tested a camera shopping task with 8 participants who had varying domain knowledge. Finally, we discussed implications for the design of future intelligent interfaces that can better understand the information being consumed by its users by taking advantage of advances in natural language processing to support online sensemaking in various scenarios.

6.2 Related Work

Past research has proposed a variety of approaches to better support complex search tasks at varying stages. Our work builds on this diverse literature by leveraging state-of-the-art entity-centric approaches in information retrieval and natural language processing to drive our infusion- and diffusion-based user interactions in an in situ interface. This allowed us to empower the

browser interface to better understand the information being consumed by its users, and to provide an entity-centric approach to sensemaking and information foraging across multiple webpages.

6.2.1 Saving and Organizing Information from the Web

When reading individual webpages, users conducting exploratory search tasks often need to take notes or save information from many webpages [115]. Past work showed that up to 86% of users save information from webpages when conducting search tasks [108]. Due to its ubiquity, browser add-ons supporting online foraging and organizing information have become popular and widespread in recent years, including Evernote with over 200 million users and Pocket with over 2 billion items saved. Researchers have also built tools to support online foraging from saving and organizing entire webpages [26] to specifying and saving parts of webpages and organizing them [31, 51, 160, 189]. Extracting and saving information across webpages in an exploratory search task can be challenging in that these webpages typically contain evidence for overlapping options, but the majority of the information is unstructured, requiring users to manually cross-reference between pages in order to gather evidence about the same option. At the same time, prior work has also pointed to how frequent context switching between different documents and taking notes is distracting, and sometimes prohibits users from investigating deeper or stop to take notes in order to avoid disrupting the flow of reading [15, 119, 128, 164]. One approach requiring content publishers to provide machine readable annotations such as using semantic web markups [12, 13], has failed to gain momentum due to a lack of available end-user tools that can consume these annotations [91]. Alternatively, researchers have also explored using in situ interfaces (e.g., a sidebar) to enable access to user notes while reading [164, 168]. However, past systems either persisted notes only on individual pages and do not support synthesizing across sources, or did not provide a scalable way for reusing and organizing large collections of saved notes. In particular, [168] reported that their participants relied on skimming or targeted search using keywords from memory for re-finding previously saved notes, which potentially led to a large portion of user notes rarely re-accessed nor deleted. Fundamentally, note taking softwares treat options mentioned on webpages and users' notes about them as independent, and the cost of cross-referencing between webpages and their notes to accumulate evidence for the same options can be prohibitively high.

Another thread of research related to our work focused on saving entity information listed on webpages in plain text via end-user programming or interaction techniques. For example, [16] and [159] allowed for efficient copying and pasting of entity attributes (e.g., addresses and phone numbers) by automatically identifying them in text, and [51, 79, 82] assisted users in extracting both entities and their attributes from webpages (e.g., a list of faculty with contact information.) While past work also points to users' needs to interact with entities and collect information about entities while browsing webpages, they mainly focused on helping users collect objective attributes about entities from individual pages and do not provide support for gathering descriptive and subjective evidence about entities across pages, which often play an important role in decision making during complex tasks such as trip planning.

In this paper, we built a prototype browser add-on, Fusion, that aims to address the aforementioned limitations of prior work — the high costs of reusing previously saved notes and evidence, cross-referencing, and context switching. Fusion utilizes open and commercial entity databases [7] and state-of-the-art entity linking algorithms [122] to automatically identify entities mentioned

across information sources in an exploratory search task. When a user encounters an entity on a webpage, Fusion presents an in situ entity card with rich attributes similar to ones used in modern commercial search interfaces [21, 124]. For example, showing the location and review ratings of different restaurants mentioned on a webpage. The in situ entity cards also serve as a foraging structure where users can attach notes and web clips to them. This also allows Fusion to automatically resurface previously saved notes and clips on other webpages that also mention the same entities so previously saved information can be efficiently reused.

6.2.2 Entities in Search Interfaces

Users conducting exploratory search tasks are unsure about their goals, and often need to rely on reading and foraging from multiple webpages to iteratively learn about the available options and gather useful evidence [115]. Prior work on search engine interfaces have focused on ways to help searchers better orient themselves by providing an overview of webpages in a search result [117, 130, 167] or managing multiple searches and information sources [67, 125]. More closely related to our work, significant research has also gone into entity-centric approaches due to the ubiquity of entities in online sensemaking tasks. Studies in 2009 and 2012 have found that entity-bearing queries and entity category queries accounted for up to 71% to 85% of web search traffic [61, 112]. This has led to significant academic and commercial efforts devoted to building large-scale entity databases (such as DBPedia [7], Yelp.com, and Google Places), and a decade of research on ways to enrich search interfaces with information about entities. Major threads of research include identifying entities mentioned in queries to present entity cards for quick referencing [21, 124], answering factual questions about entities directly [62], and showing lists of related entities to be used as subsequent queries [18, 20, 98].

While these approaches have made great strides in making retrieval more efficient based on better understandings of users' intent [124] and the content of web documents [55], significant user effort is still required after retrieving documents — when consuming and extracting information from individual webpages and organizing them. Yet the browser interface where users conduct these intense sensemaking and decision-making processes remain largely unchanged, and relatively less explored in research [91]. Current browsers treat entity mentions and their evidence described in each opened webpage independent of other webpages, making it difficult for users to cross-reference, forage, and keep track of what they are interested in and why across webpages. In this paper, we instead utilize techniques used in search systems to explore an alternative design space where the browser is aware of the same entities mentioned across webpages opened during an exploratory search task. For this, we propose to empower the browser interface with readily available open and commercial entity databases [7] and entity linking algorithms [122], and explore entity-centric approaches for supporting sensemaking across webpages.

6.3 System Design

We introduce Fusion, a novel browser add-on that uses an entity-centric approach to facilitate sensemaking across webpages in exploratory search tasks. Figure 6.1 shows an overview of how an exploratory searcher planning a trip might use Fusion. Unlike previous approaches for supporting sensemaking in exploratory search tasks, such as re-ranking or enriching the search result lists or using external note management interfaces [124, 161], we focused on providing *in*

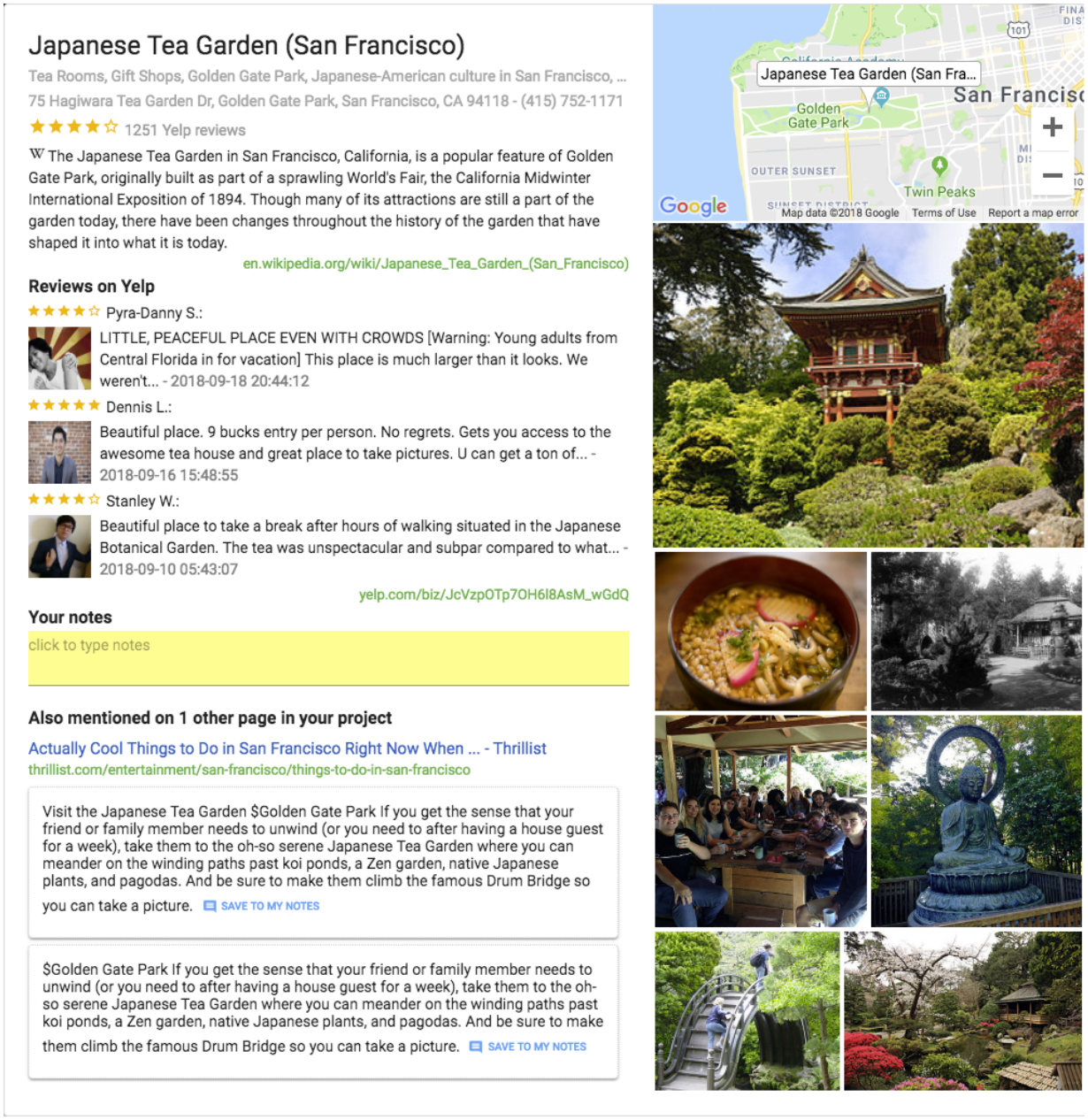

Figure 6.2: Expanded view for an entity card showing information *infused* from external knowledge sources (Yelp and Wikipedia), user's notes, and evidence of the same entity from other webpages in the exploratory search task. See Figure 6.1 for the non-expanded view of an entity card.

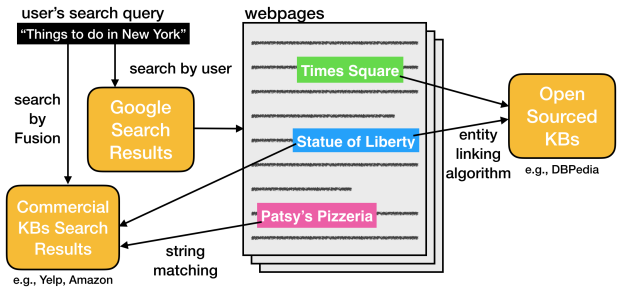

Figure 6.3: Fusion links entity mentions from webpages to both open and commercial knowledge bases.

situ support for sensemaking while reading and capturing information. Fusion provides users a lightweight overlay interface embedded and synced across webpages in different browser tabs, allowing users to make quick and lightweight cross-referencing without switching between tabs, windows, or applications. The two core components of Fusion augment content in two ways, which we introduce as "*infusion*" and "*diffusion*." First, when users open a webpage from their search results, the system "infuses" the webpage with relevant snippets about mentioned entities from other webpages in their search results and external knowledge sources to help users cross-reference and evaluate newly encountered options. Second, when users save notes or extract content from a webpage, the system "diffuses" them to mentions of the same entities in other webpages of the same task, allowing them to easily access previously saved information without having to switch to and search through a separate interface [168]. To drive these operations and connect the different webpages, we use the DBpedia Spotlight algorithm [122] to automatically identify common entities mentioned in the different webpages. In our implementation, we use Yelp and DBpedia as our entity repositories and focus on travel planning tasks, but other knowledge bases can also be used or added to support other types of projects. For example, using the Microsoft Academic Graph[1] and the Gene Ontology [6] as knowledge bases to support literature review projects in biology. In the next subsections, we will first describe in detail how Fusion identifies entities in web content, and then describe both the infusion- and diffusion-based features.

6.3.1 Linking to Open and Commercial Knowledge Bases

Users can add searches and individual webpages to their Fusion projects. When a search is added to a Fusion project, Fusion parses the HTML of the search results page to obtain the list of webpages. In the background, Fusion analyzes the content of webpages to identify entities mentioned using the following methods (Figure 6.3). First, it uses the Spotlight library [122] to identify entities mentioned in different surface forms (e.g., San Francisco Museum of Modern Art and SFMoMA) to DBpedia which contain rich attributes extracted from Wikipedia. Unlike DBpedia, Yelp is a commercial services in which neither the entity database nor a pre-trained entity linking model were publically available. In order to identify Yelp entities in webpages, we use keywords and a location extracted from the original query term users performed on Google (e.g., best sushi bars in new york) to query the Yelp Search API[2] for a list of 450 Yelp entities. This allows Fusion to retrieve from closed databases for entities that match with users query

[1]https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/
[2]https://www.yelp.com/fusion

intent and information needs. Simple string matching is used to identify mentions of any Yelp entities on each webpage.

To avoid showing duplicate entities from DBpedia and Yelp and to improve the coverage of identifying Yelp entities, Fusion use a location-based heuristic to merge entities from the two sources: two entities are merged if 1) they are from different knowledge bases, 2) have overlapping surface forms listed in the knowledge bases, and 3) have geographic coordinates that are less than two kilometers from each other. Using simple string matching to identify Yelp entities on webpages can have limited coverage since Yelp only lists one surface form for each entity (e.g., name of restaurant). However, if a Yelp entity was merged with a DBpedia entity, it is automatically applied to mentions of different surface forms as identified by the entity linking algorithm [122].

A caveat of driving end-user interfaces with machine learning is potentially having the model make occasional mistakes that can degrade the user experience, and it is crucial to provide mechanisms for the users to recover from them [99, 106, 109]. For example, in a trip planning task, a webpage might miss a popular destination not recognized by the Spotlight algorithm, not listed on DBpedia, or not covered in the results returned from the Yelp API. In addition, since Fusion depends on users' original search terms to obtain relevant entities via the Yelp Search API, webpages opened by typing URLs into the address bar does not support recognizing Yelp entities automatically in the current implementation. These issues may disrupt users' exploration processes, forcing users to resort to external tools for capturing information. To provide a way to recover from these situations, Fusion allows users to manually mark phrases on the webpage as entities and link them to entities in DBpedia and/or Yelp (Figure 6.4, we will describe in detail in the next subsection.) Finally, Fusion extracts the paragraphs around entity mentions from each webpage as supporting evidence.

6.3.2 Infusion: Gathering Evidence from other Webpages

A common activity in complex exploratory search involves collecting information from multiple sources and make informed decisions. Fusion supports this need by "infusing" entities mentioned on a page with context pulled from other webpages mentioning the same entity or entries in external knowledge sources (in our current implementation, DBpedia and Yelp). When users open a webpage, entity mentions that were recognized by Fusion are highlighted with a half-height yellow highlight (Figure 6.1, A) to indicate they have information from other sources. By hovering over an entity, a user can see an "entity card" (Figure 6.1, E) which displays those sources and relevant information (e.g., number of stars on Yelp, paragraphs from other web sites in which the entity was mentioned) which a user can use to gain context about the entity beyond the current webpage [21]. To read the mentions, the user can click on the icon of each external source to see an extracted snippet. Alternatively, the user can also expand the Card to see a larger view (Figure 6.2), showing all mentions, multiple images, and a map of the location using metadata from Yelp and/or DBpedia.

As a running example, a user planning a trip to a new city might open an article from a travel blog and see all the destination and restaurant mentions highlighted in yellow by Fusion. As the user reads the article, he or she finds a highlighted restaurant and the author recommended it for reasons that also fit our user's personal interests. However, instead of relying on this single piece of evidence, the user hovers over the restaurant name to query for its entity card for additional

information. The entity card contains Yelp review scores, Wikipedia description, and a list of relevant snippets from other webpages from the user's previous searches. After reviewing these information, the user drags the entity card under the restaurant category created previously.

While the state of entity recognition is continuously improving, there are situations when an entity isn't recognized, for example due to a lack of coverage in the recognition system or errors on the page itself. To recover from cases where an entity of interest was not recognized by Fusion, the user can still create a custom entity card by first selecting the entity name on the webpage, and click on the "Create Card" button in Fusion (Figure 6.4). In the background, Fusion queries the two knowledge sources for candidates, merges the two results lists using the location-based heuristics described in the previous subsection, and finally presents the list of candidates from which the user can pick. Alternatively, if the entity was not found in the knowledge bases, the user can still create a custom entity card (Figure 6.4). In early pilot testing, we found a common user need for this in creating an "ad hoc" entity where there might not be a specific, concrete location or entity (e.g., creating a card to collect tips about packing for Machu Pichu or general descriptions of beaches in New England).

There are several ways in which entity cards might be surfaced to provide context to users. In the first version of Fusion, we detected which entities were mentioned in the browser viewport and displayed a list of entity cards. Our intention in exploring this design was to provide a visual trigger for users to learn about the context of entities and potentially act on that context by annotating or saving relevant entities. However, participants in our preliminary user studies raised the issue that many of the entities surfaced were not relevant to their tasks, and having to sift through the list of entity cards and locate relevant entities was time consuming. Many of these irrelevant entities were either overly general locations (e.g., U.S.A), ambiguous or partial matches (e.g., "park"), or general knowledge entities from Wikipedia (e.g., Cuisine of the Southern United States). While it is possible that approaches taking into account the user's prior knowledge about the task and their query intent might improve the relevance of returned results (e.g., [129]), the density of entities within the browser viewport constitutes a more fundamental issue that leads to cluttering the browser interface and overwhelming users with too much information at once [174]. Based on these observations, we changed the interface from actively pushing all the entity information to the users to underlying recognized entities and allowing users to query information for entities that they determined to be relevant.

6.3.3 Diffusion: Propagating Notes to other Webpages

After using "infused" context to judge the relevance and suitability of options (i.e., entities), users often need to keep track of and organize the options they found valuable. At the same time, users may evaluate newly encountered options against ones they have already saved. Typically, this happens by copy-pasting or typing entity names and notes into a separate interface, for example a separate document or email or note taking software (e.g., Evernote). Researchers have tried to lower the switching cost involved in this interaction [128, 164], for example, by adding a sidebar to the browser for taking free-form notes [168]. However, in the cases when the user encounters additional evidence about an option they already have information about, they need to re-find it in the external system before being able to continue, which can lead to significant adoption issues [168].

Fusion addresses this challenge this by "diffusing" notes that users associate with an entity to

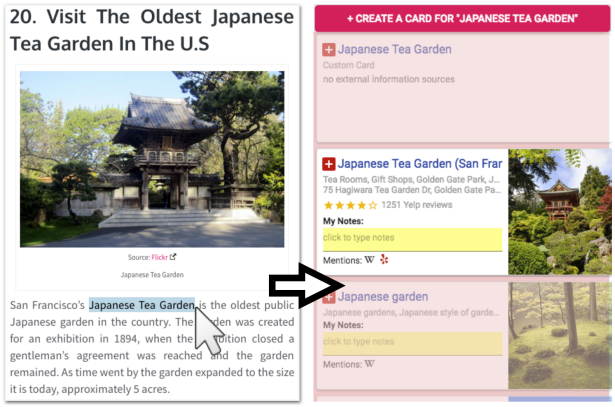

Figure 6.4: To create a missing entity, users can select a phrase (here, Japanese Tea Garden) on the page and see a list of candidates to choose from. In this case, the top 3 candidates were 1) a Custom Card not linked to external knowledge bases, 2) an entity card linked to a specific Japanese garden on both Yelp and DBpedia, and 3) and entity card linked to the general entity for Japanese gardens in DBpedia.

all other webpages in the project that also mentioned the same entity, reducing the need for user-driven re-finding. Continuing with our running example of trip planning from the previous subsection, imagine that after the user reviewes the information in the restaurant entity card, he or she decides to take notes and save the restaurant for future reference. To do so, the user can add various levels of annotation to the card, including just "hearting" it to save it in the Saved Cards view as uncategorized (Figure 6.1, top-left corner of E), typing notes about reasons for saving it (Figure 6.1, yellow region in E), or selecting sentences (Figure 6.1, around B) from the webpage to add to the entity card as a clip (Figure 6.1, E). When the user moves on to other webpages in the project, mentions of the same restaurant will be highlighted in half-height light red (Figure 6.1, B), indicating that the user previously interacted with this entity, and upon hovering will see its entity card with annotations and clips they have previously added.

Using this entity-centric approach, users can save notes of information collected across webpages under entity cards without having to switch back and forth between the browser and note-taking software, and easily re-find and reuse previously saved information when encountering the same entities on other webpages. To recover from cases where an entity of interest was not recognized by Fusion automatically, users can manually create entity cards using interactions as described in the previous subsection. If a user created an entity card that was linked to DBpedia and/or Yelp entities, all the information that was associated with them will also appear on the user-created entity card. This ensures that users can still save and retrieve information to accumulate what they have learned, even when an entity mention was not automatically recognized by Fusion.

6.3.4 Project Overview and Organizing Entities

As users in exploratory search tasks gradually progress from discovering entities and gathering evidence to focus more on synthesizing and making decisions, they may also need to organize and compare the collected entities. For example, in a travel planning task, users may want to group their entities into categories of restaurants, attractions, and hotels for comparison, and

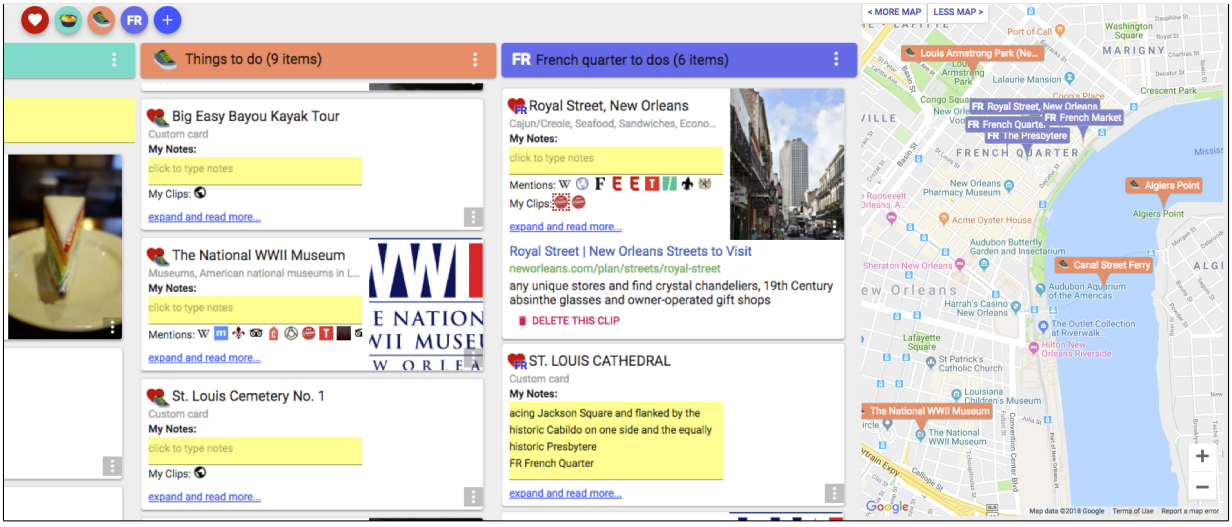

Figure 6.5: An project overview page created by one participant after searching for 50 minutes, containing entities saved under different categories, text clips from multiple webpages, and typed notes. Custom cards were created, including one for a kayaking tour that was not available on Yelp and DBpedia. The entity cards were scaled for clarity in this figure.

also to figure out the location and distances between the different entities to plan their trips.

In Fusion, in addition to simply "hearting" an entity card, users can also create categories with custom names, colors, and icons in the Saved Cards view (Figure 6.1, D). To categorize an entity card, simply drag and drop it between categories. This allows users to start structuring any time during their exploratory search process when the need arises. Saved geographic entities (entities with coordinates metadata from Yelp and/or DBpedia) will also show up in the Map View (Figure 6.1, C) with their icons and color coded pins. In addition, when users hover over an unsaved geographic entity on the current webpage, its location is also shown on the Map view. This allows users to better situate a newly encountered option with previously discovered entities to make informed decisions. For example, a user could quickly figure out that a hotel recommendation on the current webpage is not relevant by noticing in the Map view that it is too far away from most attractions that they have saved previously from other webpages. At later stages of the exploration process, users might shift their focus from reading and gathering information to synthesizing and organizing information. For this, they can open the Project Overview page by clicking on the expand button in the Saved Cards view to see all their entity cards listed in multiple columns of each category along with an integrated map view (Figure 6.5).

6.3.5 Implementation

Fusion was built as a add-on for the Google Chrome browser implemented in Javascript using the ReactJS library. The application uses Google's Firestore real-time database to store mappings between webpages and entities and user-generated notes and clips. We utilize the Yelp Search API to fetch entities from Yelp, and we use the open-sourced DBpedia Spotlight [122] software in a custom backend that identifies DBpedia entities [7] in webpages and syncs them with the end-user interface through Google's Firestore service.

6.4 Evaluation

We conducted a lab study to evaluate the benefits of Fusion for users conducting complex exploratory search tasks. The main goal of our study was to explore the benefits and challenges of an entity-centric approach for reading, cross-referencing, and collecting information across multiple webpages in the browser. More specifically, we explored whether participants find our entity-centric approach to be beneficial for sensemaking across webpages during a complex exploratory search task.

6.4.1 Study Design

We recruited 20 participants (age=19-43, $\bar{x}$=25.40, σ=7.67, 55% female) from a local participant pool. The study began with a pre-survey to collect demographic information and self-reported expertise in the domain of the assigned tasks (described below). The main part of the study was to conduct an exploratory search for 50 minutes using Fusion, followed by a 30 minute post-survey about the experience. The study was conducted using 12.5 inch Chromebooks running Chrome version 69. Each participant was compensated 15 USD.

The primary domain on which we aimed to test Fusion was a travel planning task with the following description:

> You and your friends are going on a trip to New Orleans. Help the group figure out which places you should go and where to eat during the trip.

This travel planning task was designed to test Fusion's ability to support collecting and managing evidence from multiple sources for multiple options. Travel planning has a number of characteristics that make it a good task to test new sensemaking and exploratory search approaches. For example, information is often scattered across many sources; there is a strong degree of contextualization and personalization needed (e.g., traveling somewhere with kids is very different than without); and evidence such as reviews can be noisy and subjective [36, 187]. This domain also worked well with the backing knowledge bases in our implementation – Yelp provided entities for local restaurants and tourist attractions with images and reviews, while DBpedia provided general entities extracted from Wikipedia. We tested Fusion using the travel planning task with 12 randomly selected participants.

In addition to our primary domain, we also used a camera shopping task with 8 participants using the following description:

> A friend is asking for your help to figure out what DSLR and mirrorless cameras are and which model she should buy as a beginner.

This camera shopping task aimed to test the boundary conditions of Fusion. Unlike the travel planning task, Fusion's current implementation does not have a knowledge base specialized for camera products, and most webpages in this task do not contain any geographic entities. However, unlike the travel task which assumes only general knowledge, camera shopping for DSLR or mirrorless cameras involves significant learning and necessary expertise with concepts and terms that might be unfamiliar. Participants who were novices in the domain would need to first learn the technical terms and domain knowledge in cameras and photography before advancing to the stage where they can evaluate different options (i.e., camera models) to collect evidence. In theory, Fusion could also help with general knowledge-building using information

Statement (7-point Likert-scale responses)	Travel		Camera (experts)		Camera (novices)	
Presurvey: *I know what DSLR cameras are*	-		6.75	σ=0.50	3.50	σ=1.73
It is an improvement to my current practice	5.67	σ=1.11	5.75	σ=0.50	2.75	σ=2.22
The cards were useful for reading new pages & learning new things	5.42	σ=1.11	6.00	σ=0.82	3.25	σ=2.63
Information from Yelp was useful	5.92	σ=0.86	-		-	
Information from Wikipedia was useful	5.67	σ=0.85	5.50	σ=1.29	4.00	σ=2.45
Information from other webpages was useful	5.58	σ=1.11	4.75	σ=0.96	4.50	σ=2.08
Cards allowed me to quickly refer to thing I saved from other pages	6.00	σ=0.71	5.25	σ=1.70	4.00	σ=2.45
Cards allowed me to save notes without switching between programs	6.33	σ=0.85	6.00	σ=0.82	4.25	σ=2.06
Behavior						
#of unique entity cards displayed	25.80	σ=11.75	43.50	σ=8.73	56.00	σ=13.08
#of entity cards saved in the workspace	9.40	σ=6.40	15.25	σ=5.26	5.50	σ=2.87
#of webpages each entity card accessed from	4.45	σ=1.44	4.41	σ=0.91	4.81	σ=0.75
#of entity cards with notes or clips attached	10.00	σ=6.78	11.25	σ=3.96	10.00	σ=9.80
#of notes or clips created by each participant	16.30	σ=10.72	36.00	σ=20.31	33.75	σ=15.93
#of times entity cards were displayed	116.20	σ=50.80	196.75	σ=71.80	272.00	σ=76.84
#of webpages where notes or clips saved from	5.60	σ=3.41	5.50	σ=4.56	3.75	σ=1.79
		N=12		N=4		N=4

Table 6.1: Mean statistics for Likert-scale responses in the post-survey and behavior during the study. A score of 1 indicates strong disagreement with the statement and 7 indicates strong agreement. Participants under the DSLR Camera shopping task were split into two groups based on their self-reported domain expertise in the pre-survey. Given the differences in prior knowledge and that the novices likely spent significant time on learning domain knowledge and interacted less with Fusion's features, we focused most analysis on participants assigned our primary domain of travel and the expert participants in the camera task.

from DBpedia. For example, helping novice participants to understand and take notes about unfamiliar technical terms (e.g., full-frame CMOS sensors) using entity cards from DBpedia. We chose these characteristics to probe the application of an entity-centric approach in a situation where there might not be good coverage of entities and attributes, and where some users might not have the required domain knowledge to evaluate the different options in webpages.

In both tasks, we used the following description as a motivator to encourage the participants to put in more research effort, derived from previous studies of sensemaking:

> Imagine after this task you will share your project overview page with your friend(s), along with a short summarization in an email. To convince your friend(s) of your choices, provide enough reasons and information about your choices.

After conducting the main task for 50 minutes, participants spent another 30 minutes answering a post-survey using both Likert-scale ratings and free-form responses focused on their experiences during the main task and to compare Fusion with their current practices.

6.4.2 Results

Prior Domain Expertise

In terms of domain expertise, none of the 12 participants in our primary domain task reported having ever traveled to New Orleans, and conduct travel planning once to a few times per year or once in a few years (N=11, 1). However, we discovered that the camera task had very different characteristics for half of the participants who either strongly agreed or agreed that they *understand what DSLR cameras are* versus 4 camera novices who did not agreed with the statement, using a 7-point Likert-scale where 1 indicates strong disagreement to 7 for strong agreement (Table 6.1; $\bar{x}$=6.75, 3.50, σ=0.5, 1.73). Based on reviewing their notes and saved cards, domain experts in general used Fusion to collect information about various camera models, while the novices largely spent their time reading to understand the various terminologies and differences between the two camera types listed in the task descriptions. Closer examination showed that the novices on average saved only around a third as many entity cards when compared to the experts (N=4, 4; $\bar{x}$=5.5, 15.25; σ=5.3, 2.9). Given the differences in prior knowledge and that the novices likely spent significant time on learning domain knowledge (i.e., DSLR and mirrorless cameras) and interacted less with Fusion's features, **we reported the two groups separately in Table 6.1, and focused our analysis on participants assigned our primary domain of travel and the expert participants in the camera task**. In the Limitations section, we will discuss the effects of prior domain knowledge and potential ways to address them. In the results reported below, we report descriptive statistics around interactions where appropriate.

Sensemaking across Webpages

Our main goal in designing Fusion is to leverage common entities mentioned across webpages in a complex exploratory search task to facilitate sensemaking. Fusion supports this using two core mechanisms: *Infusion* that allows users to access evidence about an entity scattered across other information sources, and *Diffusion* that allows users to save information about an entity to be propagated and resurfaced across other webpages that also mentioned the same entity. Through the two mechanisms, Fusion aims to allow users to evaluate, collect, reuse, and accumulate evidence about options across webpages for sensemaking.

Participants encountered overlapping entity options across multiple webpages in their tasks. On average, each participant in the travel planning and the camera shopping task examined with 25.80 and 43.50 unique entities in Fusion (Table 6.1; σ=11.75, 8.73; N= 12, 4), and the entity cards for each were called out via hovering over their mentions on 4.45 and 4.41 different webpages, respectively (σ=1.44, 0.91). This suggests that participants were using a set of entity cards of interests to evaluate their options when browsing different webpages.

Participants also collected information across multiple webpages. On average, each participant in the two tasks saved clips from 5.60 and 5.50 different webpages (σ=3.41, 4.56), respectively. In the post-survey, participants reported that the *diffusion*-based features were useful for *quickly referring to things I've saved from other pages* ($\bar{x}$=6.00, 5.25; σ=0.71, 1.70. Using a 7-point Likert-scale where 1 indicates strong disagreement to 7 for strong agreement). Participants cited they could keep fewer tabs opened while accumulating knowledge across webpages towards making decisions:

> "I could compound things that I had already said and supplement my knowledge to

help me arrive at a decision."

"They [the entity cards] were very useful when I needed to pull up information on something I tagged [saved], without the need to use multiple [browser] tabs..."

While re-accessing previously saved notes was a potential issue in a previous system that also allowed users to save notes using an in situ sidebar persistent across webpages [168], our findings suggest the entity-centric approach of Fusion can potentially address this by allowing participants' collected evidence from one webpage to be automatically resurfaced by the system at appropriate moments. Our findings further suggest that besides using the entity cards to re-access previously saved notes, participants were also accumulating evidence about the same options from multiple webpages to support decision making.

On the other hand, participants also found value in the evidence *infused* from other webpages, agreeing that *Information from other webpages was useful* ($\bar{x}$=5.58, 4.75, σ=1.11, 0.96), and used it to verify uncertain or potentially biased information on the current webpage:

"They [the information on entity cards] allowed me to see if the comments on this page was true."

"[They were useful because] I can know the exact condition other than the advertisements provided by their own company [referring to contents on camera product page]."

Participants also cited that the entity cards allowed them to better evaluate options encountered on the current webpage without creating and switching to additional searches:

"I liked seeing similar material for an entity [referring to clips from different webpages] and being able to continue research without having to do an actual search."

These responses showed that information *infused* in the entity cards helped participants build confidence by using them to evaluate newly encountered options and validate information presented on the current webpage. The above response in particular, suggested that lightweight cross-referencing can potentially address an issue pointed out in [119], where their participants intend to investigate other articles referenced by the current one, but avoided doing so in order to maintain their flow of reading of the current article.

Keeping Track of Options and Evidence

Participants saved multiple entity cards to keep track of the different options that they were interested in. These were mostly attractions and restaurants or camera models, depending on the tasks they were assigned to. On average, each participant in the travel and the camera tasks saved 9.40 and 15.25 entity cards at the end of the study (Table 6.1; σ=6.40, 5.26; N=12, 4) out of the 25.80 and 43.50 entities they each examined (σ=11.75, 8.73), respectively.

Participants also used Fusion to save notes and clips from webpages to entity cards as supporting evidence or reminders of why the entity cards were saved. On average, each participant in the two tasks saved 16.30 and 36.00 notes or clips (σ=10.72, 20.31) to 10.00 and 11.25 different entity cards (B4; σ=6.78, 3.96), respectively. Fusion used an in situ interface motivated by prior work that points to the high cost of context switching for note taking can break the linearity of documents, and be disruptive for reading and consuming information [128, 164]. Our participants agreed in the post-survey that the in situ design allowed them to *save notes without the*

need to switch to other programs ($\bar{x}$=6.33, 6.00; σ=0.85, 0.82) and described how it required less effort:

> "I liked that I could save entities with notes attached and look back at them all put together. I used to do this with a word doc and links and it wasn't nearly as easy."

Interestingly, participants also described how when an previously saved option became not useful during the task, they can easily remove all notes and web clips attached to its entity card with lowered efforts. This also offered an explanation on the lower number of saved entity cards when compared to the total number of cards that have notes or clips attached (Table 6.1):

> "They [the entity cards] were very useful... I could save anything, and if I didn't need it later it was simple to erase them."

These results show that entities served as options in the two tasks we tested, and that entities mentioned in text represented a useful structure for foraging across webpages. By identifying them in the browsers, users were able to use the entity cards to keep track of interesting options and organize their notes and evidence collected from webpages about them.

Engaging with Entities during Foraging and Browsing

Traditional entity-centric approaches required users to interact with entities only during the retrieval stage. Such as querying for entity cards presented in a search results pages. Enabled by Fusion, our participants were also actively engaged with entities on individual webpages throughout the browsing and foraging stages. On average, participants in the travel and camera tasks each hovered over entity mentions 116.20 and 196.75 times, respectively, (Table 6.1; N=12, 4; σ=50.80, 71.80) which correspond to 25.8 and 43.50 unique entities (σ=11.75, 8.73). In the post-survey, they agreed that *the cards were useful for reading new pages and learning new things* (σ=5.42, 6.00; σ=1.11, 0.82), and responded favorably when asked if the information from Yelp and DBPedia were useful using Likert-scales (Table 6.1) and free-form responses:

> "The entity cards were useful when I found a restaurant and saw it was also mentioned on Yelp I could go read reviews..."

These results confirmed our initial assumption that information about entities are useful beyond the retrieval stage, and that information *infused* from DBpedia and Yelp provided benefits to our participants in the two tasks we tested, helping them better characterize the different options they encountered while reading individual webpages.

6.4.3 Limitations

In general, participants in the travel planning task and expert participants in the camera shopping task considered Fusion to be an improvement compared to their current practices (Table 6.1; N=12, 4; $\bar{x}$=5.67, 5.75; σ=1.11, 0.50), while novice participants in the camera task saw less value in the system (N=4; $\bar{x}$=2.75; σ=2.22). While we originally expected Fusion to be beneficial for general knowledge building by allowing users to quickly look up the definition of unfamiliar terminologies and concepts using the entity descriptions from DBpedia, novices did not find the short description extracted from Wikipedia sufficient for this purpose (N=4; $\bar{x}$=4.00; σ=2.45). This suggests that Fusion is more beneficial for the evidence gathering and deciding stages of sensemaking compared to early learning, although it is possible that with knowledge bases more appropriate for general learning, an entity-centric approach might still be beneficial.

Participants also pointed to issues they encountered during the study, which inform ways to improve future versions of Fusion so it can be adopted by a wide range of domains and scenarios. One common theme was the high effort of capturing missing structured entity attributes from the webpages. Participants in the travel planning task further pointed to the lack of support to save structured attributes, such as a missing address and the corresponding inability to pin an entity on the Map view.

> "Easy to make a record, but not easy to record all the information I need, such as addresses and photos."
>
> "[Improve Fusion so I can] Mark the address on Google map if a place doesn't have an existing map location."

Similarly, participants in the camera shopping task cited the high cost of capturing camera specifications:

> "The cards did not provide any additional information about the cameras. However, they were useful in keeping track of the features of different cameras."

This potentially explains the higher number of saved notes for participants in the camera task. While DBpedia actually contains detailed specifications of many DSLR camera models,[3] entities in DBpedia typically have dozens to hundreds of attributes. Fusion only surfaced the most common attributes (i.e., location, short description, and categories) so it does not overwhelm users. Future work could involve better extracting structured information, for example looking at alignment between DBpedia information and information on the page, or structured information extraction from webpages through end-user interaction [16, 79].

Surprisingly, participants also mentioned discovering and navigating to useful information sources from examining information on the entity cards, which was not our original design intention:

> "[It was useful when] I was taken to mentions on sites I would not have thought of"
>
> "[entity cards] Give you info about other websites that might be useful later on."

Conversely, other participants in the camera shopping task described how they would prefer to use information sources that were familiar and trusted to them instead of webpages returned from search engines:

> "Mkbhd [Marques Brownlee, a popular YouTuber] and others are good tech reviewers and they can provide a better pro vs con list than reading [the mentions]."

This suggests a future direction for personalizing the entity cards to prioritize using sources already trusted by the user.

Finally, while participants did not make many comments about the simple category structure provided by Fusion for managing entity cards, they did point to the need for further synthesizing the collected information:

> "Fusion is helpful in the first stage of information collection, but when it comes to the final detailed plan of the trip, I still need more place for editing, adding specific time and so on."

[3] http://dbpedia.org/page/Canon_EOS_750D

However, ways to support further organizing saved entity cards into useful artifacts requires further investigation.

6.5 Discussion and Future Work

We chose a holistic evaluation despite it required us to implement a relatively complete set of features, because we believed we would learn more from testing the end-to-end system in more realistic sensemaking scenarios as opposed to trying to isolate individual interactions. An earlier version of Fusion did not include maps and the overview page and led to participants trying to maintain entities cards in Fusion and a separate Google Maps project. As a result, participants viewed the earlier version of the system as incomplete, and responded that they were spending significantly more effort than their current practice. Early testing also uncovered that showing all cards detected in the viewport immediately in the sidebar was too intrusive during complex sensemaking tasks. As a result, we switched to the current design of only showing an entity card when its underlined mentions were hovered on, which was much better received. While we used simple highlighting with different colors to indicate an entity mention has additional information and whether the same entity was interacted with before, in situ visualization techniques (such as [78]) can also be explored for annotating webpages with richer information.

Evaluation results suggest participants found Fusion to be an improvement over their current practice, and valued both infusion- and diffusion-based features. However, we also learned that adoption may be critically sensitive to the cost structure of extracting entities and attributes. Some limitations identified by our participants can potentially be addressed with approaches in previous work. For example, using end-user interactions to bootstrap entity-attribute extractors [16, 79].

We think the entity-centric approach can support exploratory tasks in a wide variety of domains involving identifying potential options and collecting evidence. However, the cost of adapting the current framework to support different domain is unclear, especially for domains where high quality knowledge bases are not readily available. In the post-survey, we also asked participants if they could think of other search tasks that may benefit from using Fusion, and participants pointed to a variety of different tasks, including essay writing, event planning, literature review, job searching, and deciding on a college major.

Many other future research directions present themselves. For example, automatically summarizing evidence gathered across different websites for an entity instead of simply listing them could help Fusion scale to much larger projects with many sources, while keeping gathered information easy to consume for the users. However, how to surface information sources in the summarization so users can better evaluate the trustworthiness of the pieces of information is still an open problem. While Fusion supported synthesizing entities and evidence into categories, providing support for creating different structures (e.g., tables or essays) still needs further investigation.

Our results have implications for the design of future intelligent browser interfaces that can better understand the information being consumed by its users, and building novel interactive systems for supporting online sensemaking. As phenomena such as fake news and shill reviews have demonstrated, there are significant drawbacks to the easy availability and generation of online content. Interactive systems that can provide additional context to users *in situ* may become

increasingly necessary to help navigate the information overload. Anecdotal evidence for this need can also be seen in the rise of aggregation-based sites such as Metacritic or Wirecutter, which act as virtual meta-analyses of evidence and opinions but fail to take into account the personal context of the user and their goals. We believe that this work presents a step forward in illustrating a design space for interactive systems which can take advantage of advances in machine learning and natural language processing to help end users actively gain context and personalize their online sensemaking activities.

Chapter 7: Proposed Work

Foraging and Structuring across Webpages in the Browser

The first goal of the proposed work is to further develop the Fusion browser extension prototype described in Chapter 6 for a public field deployment and test the system at scale and in real-life scenarios. Based on lab studies and interviews with the participants, I have identified two areas of user needs that are critical and currently not supported by the system. More specifically, many participants expressed the need to save and manage many information sources (i.e., entire webpages instead of short clips) for different purposes, citing how conducting complex search tasks can lead to an overwhelming amount of opened tabs that can be difficult to manage. Participants also pointed to the limitation of saving notes to entity cards generated by the system, and wanted to organize information more flexibly to reflect their evolving mental models throughout the process. Addressing these has the potential of allowing the system to better supporting the next steps in online sensemaking – structuring collected information while efficiently managing multiple information sources.

My primary focus will be on further developing the research prototype for a public release while experimenting with new approaches within the same system to support structuring and source (webpages) management. While a public deployment has the potential of allowing the system to reach a wider audience outside of the lab and collect usage information at scale, user tests and lab studies may also be required throughout the process of system design and development. Success in gaining popularity would also depend on many factors outside of the scope of my dissertation, such includes visual design, marketing and market understanding, and competing commercial products. While the aim is to develop a large user-base, the proposed approaches can also be evaluated with lab studies and smaller scale field deployments using participants recruited from a local participant pool (CBDR).

7.1 Preliminary Study: Challenges in Tabbed Browsing Behavior

Browser tabs have become an integral part of how people browse and navigate the web since they were introduced in the early 2000s, and have since became an ubiquitous feature in all major web browsers. However, since their introduction, the internet has gone through dramatic changes. Tabs are now simultaneously used to check emails, control media players, stash articles to read later, organize references, plan trips, research products, write articles. More fundamentally, online information seeking has evolved from navigating web directories to find useful websites (e.g. DMOZ [138]) to searching the entire web for dozens to hundreds of individual webpages to support complex sensemaking tasks [116, 134]. These changes reflect an increasing amount of dependence on the functionality and interfaces of modern web browsers in meeting these needs. Despite this expansion of functionality, browser tabs still remain instantiated as simple temporally-ordered lists with few contextual cues. There is increasing evidence that using tabs for this wide array of functions leads to breakdowns, overload, and missed opportunities [63, 64, 65, 140, 141].

Different Pressures for Closing Tabs versus Keeping Tabs Opened	
C1.Limited Attention	Keeping too many tabs can be overwhelming and makes it difficult to focus
C2.Screen Real-estate	Having too many tabs makes it hard to navigate and have situational awareness
C3.Computing power	Drains processors and memory, causing browser and other applications to slow
C4.To be Organized	Social and self pressure to avoid looking disorganized
O1.Remind and Resume	Keeping tabs around as a reminder to work on them or keep track of progress
O2.Revisit References	Keeping frequently used tabs for quick access; has a diminishing return
O3.Costly Re-finding	Avoid closing tabs in fear of not being able to re-find valuable information
O4.Aspiration/Sunk Cost	The hopes to process more information than capable; while aware of the situation
O5.Mental Model	Tabs and windows represent external memory and mental models for complex tasks
O6.Uncertain Relevance	Difficulties in judging the current and potential relevance of tabs in the future

Table 7.1: Browser tabs are overloaded with different functionalities, such as todo items, external memory, or references, leading to two sets of opposing pressures that drive tabbed browsing behavior.

To investigate design opportunities that can expand the current prototype to better support the process of searching, foraging, and structuring, I conducted an empirical study on the challenges people face when using modern browsers. Instead of examining the specific functions of tabs as they are currently used (e.g., as in Dubroy, 2010 [52]), I focus on developing a model of the way that tabs break down from their daily use. Ten graduate strudents or full-time researchers were recruited from the university and a research facility as participants for in-person interviews (age: M=23.0, SD=4.7, min=19, max=32, 40% male). Participants were first asked to walk through each tab they had opened on their work computers and explain the tasks, goals, or purposes of why each tab was opened in the first place and why it was kept opened, using questions including "*Was this tab intentionally kept around for later usage?*" If answered "*yes*", we followed up with "*Did you came back to it recently, why or why not?*" And if answered "*no*", we followed up with "*Why was this tab kept around if you did not plan to use it again?*", and "*Do you struggle to close your tabs?*" We also asked about how and how effective are they managing their tabs with questions include "*How frequently do you evaluate your tabs to see if you can close them? How difficult is it?*", and "*Does the number of tabs you have opened affect how you feel?*" The interviews were recorded, transcribed, and analyzed using a grounded theory approach with four rounds of discussions and coding [158].

Table 7.1 shows an overview of major drivers people encountered while using tabs for information work. Overall, these drivers could be classified as two opposing forces: pressures to close tabs, and pressures to keep tabs open. I found strong evidence that participants had a number of pressures to close their many tabs, ranging from limited human attention to limited computing resources to self presentation. At the same time, there are also a diverse set of drivers making it not so simple to close tabs even under these pressures. These included previously reported drivers such as reminding users of unfinished tasks [52], but also new factors relating to cost structure of tabbed browsing, such as the cost of re-accessing pages, the sunk costs of finding and organizing information, the benefits of supporting an (unrealistic) aspirational self, and the uncertainty of the expected value of information in the future, especially when searching in unfamiliar domains. These pressures to close vs. keep open tabs interact to create feelings of stress, being overwhelmed, and even shamefulness of appearing unorganized in our participants. These findings have implications for the design of new forms of web browsing that can better support the underlying drivers behind the use of tabs. In the next two subsection, I

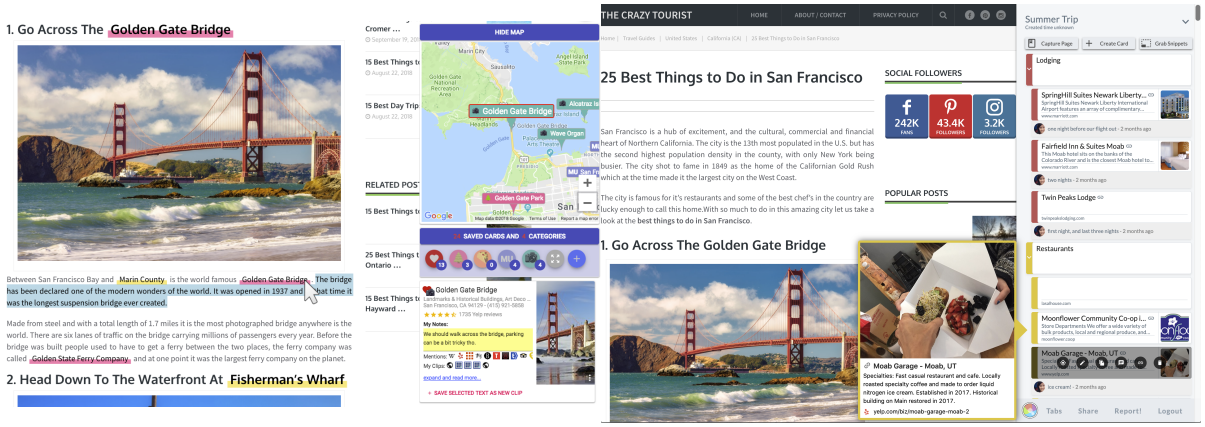

Figure 7.1: The current prototype allowed users to access entity cards generated by the system, and saved on into categories (left). In the new design, users can also create notes and category cards, and nest cards to create a hierarchy structure (right).

propose two approaches that aim to better support issues caused by different drivers that we characterized from both Chapter 6 and this preliminary study.

7.2 Foraging and Structuring Information

In Chapter 6 I presented a prototype system called Fusion Browser that focused on exploiting common entities mentioned across webpages in a complex search task as a substrate to support foraging. One the features of Fusion Browser is to allow users to save information about an entity to be propagated and resurfaced across other webpages that also mentioned the same entity, allowing users to efficiently cross-reference and build on their prior effort as they explore more webpages. While participants generally agreed that this entity-centric approach lowered the cost of saving information and accessing them during complex search tasks involving multiple webpages, they also pointed to its limitations. Specifically, the current prototype only allowed users to attach information to entity cards and organize cards under categories. While participants found this to be effective for foraging, they also expressed needs for creating structures more flexible than categories of pre-compiled entity cards that can better reflect their changing mental models. They cited their current practices of using word processors (such as Google Docs) to create outlines or tables containing both information copied from webpages and manual notes about their own thoughts and decisions. In addition, similar to participants from the preliminary study, participants from Chapter 6 also reported the need to save links of useful webpages.

Due to its ubiquity and generalizability, I plan to extend the current prototype to support building outlines during foraging. In the new design, instead of attaching collected information to entity cards created by the system, users can freely create manual note cards to externalize their thoughts and decisions. To create structures that can better reflect users' mental models, in the new design users can also nest cards under other cards to create hierarchies. This design is similar to how many participants reported using note taking software during complex search tasks, but the core challenge here is to lower the cost of structuring and externalization as they can be either disruptive for users' reading process or prohibitive for externalizing [15, 119, 128, 164]. My insights from developing Fusion Browser (Chapter 6) that focused on in-situ information

foraging represent a unique starting point for investigating ways to support in-situ information structuring as users explore and foraging from multiple webpages in a complex search task. In addition, my past work on the Alloy system (Chapter 3) for structuring web content with machine learning and crowd-microtasks can also provide insights on designs and interactions that can further lower the costs of structuring for the end users through machine learning.

7.3 Managing Information Sources in the Browser

From the preliminary interviews and the lab studies from Chapter 6, We also observed that exploratory searches can lead to large numbers of open tabs in a short period of time, and quickly becomes difficult for the users to manage. These tabs can also be overloaded with different purposes, such as queuing up to-do tasks, reminders for things to go back to, or potential future references. Existing approaches for saving browser tabs have different issues. For example, the built-in bookmarking feature saves tab in a separate folder structure. This not only introduce additional costs of maintaining a structure separated from users notes, but can also create conflicting mental model representations requiring users' cross-refernce between them. In addition, turning a browser tab into a bookmark in the browser also makes it much less accessible and out of sight for the users when compared to tabs. On the other hand, popular bookmarking browser extensions such as OneTab[1] allow users to save and re-open sets of tabs as sessions for resumption retain more task structures, but also takes away many features of browser tabs such as reminding. I plan to extend the current system to provide better support for managing information sources during online sensemaking tasks to address issues introduced by tab overload, such as:

- Allowing users to organize information sources using their existing notes outline vs creating a separate bookmark folder structure.
- Allowing users to close and save tabs and indicate their current functions. For example, closing tabs and marking them as reminders or to-dos, useful references, or articles a user aspired to read.
- The system can in turn use different strategies to resurface previously saved tabs. For example, showing reminder tabs in prominent places such as the default newtab page, resurfacing references when users is reading about a related webpage, or proactively encourage users to read previously saved articles when they were browsing social networks.

These designs were informed by the preliminary interviews.

7.4 Evaluation and Contributions

Since note taking and structuring are longer term activities when compared to foraging, I plan to publically release a browser extension for Chrome to reach a wider audience, collect usage information. This would allow me to gather quantitative data based on real-life tasks, for example:

- Time spent using the system
- Types of tasks conducted

[1] https://www.one-tab.com/

- Features utilized
- Number of cards and notes created per task
- Types of structures created

Throughout development, I will also conduct lab studies and usability tests to collect qualitative data and better understand how users may utilize the system, for example:

- General usability of the system
- Confidence in their process and decisions
- Overall preferences, satisfaction and motivation

The core contribution of the proposed work is an in-situ and context-aware approach for supporting sensemaking across webpages in the browser when conducting complex exploratory search tasks, which includes:

- An extension that enables the browser to better understand the content of webpages by identifying entities using existing natural language processing algorithms and external knowledge sources (e.g., Wikipedia and Yelp), lowering the cost of cross-referencing different entities across webpages, searches, and external knowledge sources.
- A context-aware workspace where users can easily gather and structure options and evidence across multiple webpages, where relevant information saved previously will be resurfaced by the system as users browse different pages.
- Qualitative analysis on issues that information workers face when trying to manage multiple browser tabs during complex tasks, and a set of new approaches for managing multiple online sources under uncertainty that aim to address such issues.

Equipping browsers and note-taking interfaces the ability to identify and connect options and evidence scattered across tabs and personal notes have the potentials of lowering the efforts required to make sense of and forage from many information sources. Empowering users to capture, associate, and structure fluidly their findings as they read and understand more information. Successes in doing so may also lead to useful insights on the design of future intelligent browser interfaces that can better understand the information being consumed by its users, and building novel interactive systems for supporting online sensemaking.

7.5 Timeline and Submission Plans

Timeline

- April - June 2019: Prototype Development
- June - August 2019: Deployment
- August - September 2019: Evaluation and Analysis
- October - December 2019: Write Dissertation and Defend

Submission Plans

- ACM UIST 2019: Chapter 6

- ACM SIGCHI 2019: Chapter 7

Bibliography

[1] Annette Adler, Anuj Gujar, Beverly L Harrison, Kenton O'Hara, and Abigail Sellen. A diary study of work-related reading: design implications for digital reading devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 241–248. ACM Press/Addison-Wesley Publishing Co., 1998. 5.1

[2] Jae-wook Ahn and Peter Brusilovsky. Adaptive visualization of search results: Bringing user models to visual analytics. *Information Visualization*, 8(3):167–179, 2009. 4.2.1

[3] Jae-wook Ahn, Peter Brusilovsky, Jonathan Grady, Daqing He, and Sue Yeon Syn. Open user profiles for adaptive news systems: help or harm? In *Proceedings of the 16th international conference on World Wide Web*, pages 11–20. ACM, 2007. 4.5

[4] J Anderson. Forrester Market Report: Consumer Behavior Online: A 2009 Deep Dive. http://www.forrester.com/go?docid=54327, 2009. Accessed: 2017-09-10. 6.1

[5] Paul André, Aniket Kittur, and Steven P Dow. Crowd synthesis: Extracting categories and clusters from complex data. In *Proc. CSCW 2014*, 2014. 2, 3.1, 3.1.1, 3.3.4

[6] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25, 2000. 6.3

[7] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007. 6.2.1, 6.2.2, 6.3.5

[8] Krisztian Balog, Pavel Serdyukov, and Arjen P De Vries. Overview of the trec 2010 entity track. Technical report, NORWEGIAN UNIV OF SCIENCE AND TECHNOLOGY TRONDHEIM, 2010. 6.1

[9] Nicholas J Belkin, Colleen Cool, Judy Jeng, Amymarie Keller, Diane Kelly, Ja-Young Kim, Hyuk-Jin Lee, Muh-Chyun (Morris) Tang, and Xiao-Jun Yuan. Rutgers' trec 2001 interactive track experience. In *TREC*, 2001. 4.1

[10] Nicholas J Belkin, Diane Kelly, G Kim, J-Y Kim, H-J Lee, Gheorghe Muresan, M-C Tang, X-J Yuan, and Colleen Cool. Query length in interactive information retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 205–212. ACM, 2003. 4.1

[11] Richard Ernest Bellman. *Dynamic Programming*. Dover Publications, Incorporated, 2003. ISBN 0486428095. 3.1, 3.1.1

[12] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. DIANE Publishing Company, 2001. 6.2.1

[13] Tim Berners-Lee and James Hendler. Publishing on the semantic web. *Nature*, 410(6832):

1023, 2001. 6.2.1

[14] Michael S Bernstein, Greg Little, Robert C Miller, Björn Hartmann, Mark S Ackerman, David R Karger, David Crowell, and Katrina Panovich. Soylent: a word processor with a crowd inside. In *Proc. UIST 2010*, pages 313–322. ACM, 2010. 3.5

[15] Andrea Bianchi, So-Ryang Ban, and Ian Oakley. Designing a physical aid to support active reading on tablets. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 699–708. ACM, 2015. 6.1, 6.2.1, 7.2

[16] Eric A Bier, Edward W Ishak, and Ed Chi. Entity quick click: rapid text copying based on automatic entity extraction. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems*, pages 562–567. ACM, 2006. 5.1, 6.2.1, 6.4.3, 6.5

[17] Steven Bird and Edward Loper. Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 31. Association for Computational Linguistics, 2004. 4.3.5

[18] Roi Blanco, Berkant Barla Cambazoglu, Peter Mika, and Nicolas Torzec. Entity recommendations in web search. In *International Semantic Web Conference*, pages 33–48. Springer, 2013. 6.1, 6.2.2

[19] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003. 3.1, 3.1.1, 3.3.6

[20] Ilaria Bordino, Yelena Mejova, and Mounia Lalmas. Penguins in sweaters, or serendipitous entity search on user-generated content. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 109–118. ACM, 2013. 6.1, 6.2.2

[21] Horatiu Bota, Ke Zhou, and Joemon M Jose. Playing your cards right: The effect of entity cards on search behaviour and workload. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, pages 131–140. ACM, 2016. 6.1, 6.2.1, 6.2.2, 6.3.2

[22] Jonathan Bragg, Daniel S Weld, et al. Crowdsourcing multi-label classification for taxonomy creation. In *First AAAI conference on human computation and crowdsourcing*, 2013. 2, 3.1, 3.1.1

[23] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM, 2005. 4.2.1

[24] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. ACM, 2007. 4.2.1

[25] Robert Capra, Gary Marchionini, Javier Velasco-Martin, and Katrina Muller. Tools-at-hand and learning in multi-session, collaborative search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 951–960, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. doi: 10.1145/1753326.1753468. URL http://doi.acm.org/10.1145/1753326.1753468. 5.1

[26] Stuart K Card, George G Robertson, and William York. The webbook and the web forager: an information workspace for the world-wide web. *in Proceedings of ACM SIGCHI'96*, 1996. 6.2.1

[27] Pew Research Center. Generational differences in online activities. Report, July 2015. http://www.pewinternet.org/2009/01/28/generational-differences-in-online-activities/. 3.4.1

[28] Allison June-Barlow Chaney and David M Blei. Visualizing topic models. In *ICWSM*, 2012. 3.1

[29] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM 2011 TIST*, 2(3):27, 2011. 3.2.1

[30] Joseph Chee Chang, Nathan Hahn, and Aniket Kittur. Supporting mobile sensemaking through intentionally uncertain highlighting. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 61–68, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4189-9. doi: 10.1145/2984511.2984538. URL http://doi.acm.org/10.1145/2984511.2984538. 5

[31] Joseph Chee Chang, Nathan Hahn, and Aniket Kittur. Supporting mobile sensemaking through intentionally uncertain highlighting. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 61–68. ACM, 2016. 2, 6.2.1

[32] Joseph Chee Chang, Aniket Kittur, and Nathan Hahn. Alloy: Clustering with crowds and computation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 3180–3191. ACM, 2016. 2, 3, 4.2.2

[33] Joseph Chee Chang, Saleema Amershi, and Ece Kamar. Revolt: Collaborative crowdsourcing for labeling machine learning datasets. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, New York, NY, USA, 2017. ACM. doi: 10.1145/3025453.3026044. URL http://doi.acm.org/10.1145/3025453.3026044. 4.2.2

[34] Joseph Chee Chang, Nathan Hahn, Adam Perer, and Aniket Kittur. Searchlens composing and capturing complex user interests for exploratory search. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, IUI '19, Marina del Rey, CA, USA, 2019. ACM. doi: 10.1145/3301275.3302321. URL http://doi.acm.org/10.1145/3301275.3302321. 4

[35] Olivier Chapuis and Nicolas Roussel. Copy-and-paste between overlapping windows. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 201–210. ACM, 2007. 5.1

[36] Chao Chen, Daqing Zhang, Bin Guo, Xiaojuan Ma, Gang Pan, and Zhaohui Wu. Tripplanner: Personalized trip planning leveraging heterogeneous crowdsourced digital footprints. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1259–1273, 2015. 6.4.1

[37] Chen Chen, Simon T Perrault, Shengdong Zhao, and Wei Tsang Ooi. Bezelcopy: an efficient cross-application copy-paste technique for touchscreen smartphones. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, pages 185–192. ACM, 2014. 5.1, 5.2.1, 5.4

[38] Liren Chen and Katia Sycara. Webmate: A personal agent for browsing and searching. In *Proceedings of the second international conference on Autonomous agents*, pages 132–139. ACM, 1998. 4.2.1

[39] Nan-Chen Chen, Jina Suh, Johan Verwey, Gonzalo Ramos, Steven Drucker, and Patrice Simard. Anchorviz: Facilitating classifier error discovery through interactive semantic data

exploration. In *23rd International Conference on Intelligent User Interfaces*, pages 269–280. ACM, 2018. 4.2.2

[40] Tao Cheng, Xifeng Yan, and Kevin Chen-Chuan Chang. Entityrank: searching entities directly and holistically. In *Proceedings of the 33rd international conference on Very large data bases*, pages 387–398. VLDB Endowment, 2007. 6.1

[41] Alexander Chernev, Ulf Böckenholt, and Joseph Goodman. Choice overload: A conceptual review and meta-analysis. *Journal of Consumer Psychology*, 25(2):333–358, 2015. 1

[42] Lydia B Chilton, Greg Little, Darren Edge, Daniel S Weld, and James A Landay. Cascade: Crowdsourcing taxonomy creation. In *Proc. CHI 2013*, pages 1999–2008. ACM, 2013. 1, 2, 3.1, 3.1.1, 3.3.6

[43] Lydia B Chilton, Juho Kim, Paul André, Felicia Cordeiro, James A Landay, Daniel S Weld, Steven P Dow, Robert C Miller, and Haoqi Zhang. Frenzy: Collaborative data organization for creating conference sessions. In *Proc. CHI 2014*, pages 1255–1264. ACM, 2014. 3.3.3

[44] Jason Chuang, Christopher D Manning, and Jeffrey Heer. Termite: Visualization techniques for assessing textual topic models. In *Proc. of the International Working Conference on Advanced Visual Interfaces*, pages 74–77. ACM, 2012. 3.1, 3.3.6

[45] Jason Chuang, Daniel Ramage, Christopher Manning, and Jeffrey Heer. Interpretation and trust: Designing model-driven visualizations for text analysis. In *Proc. CHI 2012*, pages 443–452. ACM, 2012. 2, 3.1

[46] Bart De Langhe, Philip M Fernbach, and Donald R Lichtenstein. Navigating by the stars: Investigating the actual and perceived validity of online user ratings. *Journal of Consumer Research*, 42(6):817–833, 2015. (document), 1, 4.1

[47] Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990. 3.1.1, 3.3.6

[48] Cecilia di Sciascio, Vedran Sabol, and Eduardo E Veas. Rank as you go: User-driven exploration of search results. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, pages 118–129. ACM, 2016. 4.3.3

[49] Cecilia di Sciascio, Peter Brusilovsky, and Eduardo Veas. A study on user-controllable social exploratory search. In *23rd International Conference on Intelligent User Interfaces*, pages 353–364. ACM, 2018. 4.3.3

[50] M. M. Schaffer D.L. Medin. Context theory of classification learning. *Psychological review*, 85(3):207, 1978. 3.2.1

[51] Mira Dontcheva, Steven M Drucker, Geraldine Wade, David Salesin, and Michael F Cohen. Collecting and organizing web content. In *Personal Information Management-Special Interest Group for Information Retrieval Workshop*, pages 44–47, 2006. 6.2.1

[52] Patrick Dubroy and Ravin Balakrishnan. A study of tabbed browsing among mozilla firefox users. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 673–682, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. doi: 10.1145/1753326.1753426. URL `http://doi.acm.org/10.1145/1753326.1753426`. 7.1

[53] Jerry Alan Fails and Dan R Olsen Jr. Interactive machine learning. In *Proc. IUI 2003*, pages 39–45. ACM, 2003. 3.1

[54] Guillaume Faure, Olivier Chapuis, and Nicolas Roussel. Power tools for copying and moving: useful stuff for your desktop. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1675–1678. ACM, 2009. 5.1

[55] Miriam Fernandez, Vanessa Lopez, Marta Sabou, Victoria Uren, David Vallet, Enrico Motta, and Pablo Castells. Semantic search meets the web. In *2008 IEEE international conference on semantic computing*, pages 253–260. IEEE, 2008. 6.2.2

[56] James Fogarty, Desney Tan, Ashish Kapoor, and Simon Winder. Cueflik: interactive concept learning in image search. In *Proceedings of the sigchi conference on human factors in computing systems*, pages 29–38. ACM, 2008. 4.2.2

[57] Kristofer Franzen and Jussi Karlgren. Verbosity and interface design. *SICS Research Report*, 2000. 4.1

[58] Qiwei Gan, Qing Cao, and Donald Jones. Helpfulness of online user reviews: More is less. *Eighteenth Americas Conference on Information Systems*, 2012. 1, 4.1, 6.1

[59] Anindya Ghose and Panagiotis G. Ipeirotis. Designing novel review ranking systems: Predicting the usefulness and impact of reviews. In *Proceedings of the Ninth International Conference on Electronic Commerce*, ICEC '07, pages 303–310, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-700-1. doi: $10.1145/1282100.1282158$. URL `http://doi.acm.org/10.1145/1282100.1282158`. 1

[60] Malcolm Gladwell. The ketchup conundrum. *New Yorker*, 6, 2004. 1

[61] Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 267–274. ACM, 2009. 6.1, 6.2.2

[62] Kelvin Guu, John Miller, and Percy Liang. Traversing knowledge graphs in vector space. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 318–327, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: $10.18653/v1/D15\text{-}1038$. URL `https://www.aclweb.org/anthology/D15-1038`. 6.1, 6.2.2

[63] Life Hacker. Master Your Browsers Tabs with These Tricks and Extensions. http://lifehacker.com/5883299/master-your-browsers-tabs-with-these-tricks-and-extensions, 2012. Accessed: 2017-09-10. 7.1

[64] Life Hacker. Why You Should Never Have More Than Nine Browser Tabs Open. http://lifehacker.com/5984149/why-you-should-never-have-more-than-nine-browser-tabs-open, 2013. Accessed: 2017-09-10. 7.1

[65] Life Hacker. It's Okay to Open More Than Nine Browser Tabs; Here's How to Easily Manage Them. http://lifehacker.com/5985462/its-okay-to-open-more-than-nine-browser-tabs-you-just-need-to-manage-them-properly, 2013. Accessed: 2017-09-10. 7.1

[66] Nathan Hahn, Joseph Chang, Ji Eun Kim, and Aniket Kittur. The knowledge accelerator: Big picture thinking in small pieces. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 2258–2270. ACM, 2016. 3, 3.3.5, 3.4, 3.4

[67] Nathan Hahn, Joseph Chee Chang, and Aniket Kittur. Bento browser: Complex mobile

search without tabs. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 251. ACM, 2018. 6.2.2

[68] Jaehyun Han and Geehyuk Lee. Push-push: A drag-like operation overlapped with a page transition operation on touch interfaces. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, pages 313–322. ACM, 2015. 5.1

[69] Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Advances in psychology*, 52:139–183, 1988. 5.3.2

[70] John A Hartigan. *Clustering algorithms*. John Wiley & Sons, Inc., 1975. 3.1.1

[71] Marti Hearst. Design recommendations for hierarchical faceted search interfaces. In *ACM SIGIR workshop on faceted search*, pages 1–5. Seattle, WA, 2006. 2

[72] Marti Hearst, Ame Elliott, Jennifer English, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Finding the flow in web site search. *Communications of the ACM*, 45(9):42–49, 2002. 2

[73] Marti A Hearst. Clustering versus faceted categories for information exploration. *Communications of the ACM*, 49(4):59–61, 2006. 2

[74] Marti A Hearst and Jan O Pedersen. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 76–84. ACM, 1996. 2

[75] Marti A Hearst and Jan O Pedersen. Visualizing information retrieval results: a demonstration of the tilebar interface. In *Conference Companion on Human Factors in Computing Systems*, pages 394–395. ACM, 1996. 4.1, 4.3.1, 4.3.2, 4.4.1

[76] Ken Hinckley, Xiaojun Bi, Michel Pahud, and Bill Buxton. Informal information gathering techniques for active reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1893–1896. ACM, 2012. 2, 5.1

[77] Orland Hoeber and Xue Dong Yang. A comparative user study of web search interfaces: Hotmap, concept highlighter, and google. In *Web Intelligence, 2006. WI 2006. IEEE/WIC/ACM International Conference on*, pages 866–874. IEEE, 2006. 4.1, 4.3.2, 4.3.2, 4.4.1

[78] Jane Hoffswell, Arvind Satyanarayan, and Jeffrey Heer. Augmenting code with in situ visualizations to aid program understanding. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 532. ACM, 2018. 6.5

[79] Andrew Hogue and David Karger. Thresher: automating the unwrapping of semantic content from the world wide web. In *Proceedings of the 14th international conference on World Wide Web*, pages 86–95. ACM, 2005. 6.2.1, 6.4.3, 6.5

[80] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: $10.1145/1014052.1014073$. URL `http://doi.acm.org/10.1145/1014052.1014073`. 1

[81] Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. Interactive topic modeling. *Machine learning*, 95(3):423–469, 2014. 3.1

[82] David Huynh, Stefano Mazzocchi, and David Karger. Piggy bank: Experience the semantic web inside your web browser. In *International Semantic Web Conference*, pages

413–430. Springer, 2005. 6.2.1

[83] Yelp Inc. The Yelp Dataset Challenge: Discover what insights lie hidden in our data. https://www.yelp.com/dataset/challenge, 2016. Accessed: 2017-09-10. 4.1, 4.3

[84] Zachary Ives, Craig Knoblock, Steve Minton, Marie Jacob, Partha Talukdar, Rattapoom Tuchinda, Jose Luis Ambite, Maria Muslea, and Cenk Gazen. Interactive data integration through smart copy & paste. *arXiv preprint arXiv:0909.1769*, 2009. 5.1

[85] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31:3, 1999. 3.1

[86] Anil K Jain and Richard C Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988. 3.1.1

[87] Bernard J Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information processing & management*, 36(2):207–227, 2000. 4.1

[88] Bernard J Jansen, Danielle L Booth, and Amanda Spink. Patterns of query reformulation during web searching. *Journal of the American society for information science and technology*, 60(7):1358–1371, 2009. 3.2.1

[89] Karen SpÃďrck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972. 3.1.1, 3.3.6

[90] Bryan Jurish and Kay-Michael Würzner. Word and sentence tokenization with hidden markov models. *JLCL*, 28(2):61–83, 2013. 4.3.5

[91] David R Karger. The semantic web and end users: What's wrong and how to fix it. *IEEE Internet Computing*, 18(6):64–70, 2014. 2, 6.2.1, 6.2.2

[92] Melanie Kellar, Carolyn Watters, and Michael Shepherd. A field study characterizing web-based information-seeking tasks. *Journal of the American Society for Information Science and Technology*, 58(7):999–1018, 2007. 6.1

[93] Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 423–430, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. ISBN 1-932432-73-6. URL `http://dl.acm.org/citation.cfm?id=1610075.1610135`. 1

[94] David Kirsh. The intelligent use of space. *Artificial intelligence*, 73(1):31–68, 1995. 5.1

[95] Aniket Kittur, Ed H Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proc. CHI 2008*, pages 453–456. ACM, 2008. 3.2.3

[96] Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E Kraut. Crowdforge: Crowdsourcing complex work. In *Proc UIST 2011*, pages 43–52. ACM, 2011. 3.5

[97] Aniket Kittur, Andrew M Peters, Abdigani Diriye, Trupti Telang, and Michael R Bove. Costs and benefits of structured information foraging. In *Proc. CHI 2013*, pages 2989–2998. ACM, 2013. 2, 3.3.2, 3.5, 5.1

[98] Khalil Klouche, Tuukka Ruotsalo, Diogo Cabral, Salvatore Andolina, Andrea Bellucci, and Giulio Jacucci. Designing for exploratory search on touch devices. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 4189–4198. ACM, 2015. 6.1, 6.2.2

[99] Rafal Kocielnik, Saleema Amershi, and Paul N Bennett. Will you accept an imperfect ai? exploring designs for adjusting end-user expectations of ai systems. 2019. 6.3.1

[100] Hans-Peter Kriegel, Peer Kröger, and Arthur Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM TKDD*, 2009. 3.1, 3.1.1

[101] Srikumar Krishnamoorthy. Linguistic features for review helpfulness prediction. *Expert Systems with Applications*, 42(7):3751–3759, 2015. 1

[102] Todd Kulesza, Saleema Amershi, Rich Caruana, Danyel Fisher, and Denis Charles. Structured labeling for facilitating concept evolution in machine learning. In *Proc. CHI 2014*, pages 3075–3084, 2014. 3.1

[103] Todd Kulesza, Saleema Amershi, Rich Caruana, Danyel Fisher, and Denis Charles. Structured labeling for facilitating concept evolution in machine learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3075–3084. ACM, 2014. 4.2.2

[104] Edward Lank and Eric Saund. Sloppy selection: Providing an accurate interpretation of imprecise selection gestures. *Computers & Graphics*, 29(4):490–500, 2005. 5.1

[105] Gierad Laput, Walter S Lasecki, Jason Wiese, Robert Xiao, Jeffrey P Bigham, and Chris Harrison. Zensors: Adaptive, rapidly deployable, human-intelligent sensor feeds. In *Proc. CHI 2015*, pages 1935–1944. ACM, 2015. 3.5

[106] Min Kyung Lee, Sara Kiesler, Jodi Forlizzi, Siddhartha Srinivasa, and Paul Rybski. Gracefully mitigating breakdowns in robotic services. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 203–210. IEEE, 2010. 6.3.1

[107] Seunghwan Lee and Geehyuk Lee. Design of interaction techniques for clickable touch screens. In *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: The Future of Design*, OzCHI '14, pages 568–577, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-0653-9. doi: $10.1145/2686612.2686702$. URL `http://doi.acm.org/10.1145/2686612.2686702`. 5.2.1

[108] Seungyon Claire Lee, Eamonn O'Brien-Strain, Jerry Liu, and Qian Lin. A survey on web use: how people access, consume, keep, and organize web content. In *CHI Extended Abstracts*, pages 619–628, 2012. 6.2.1

[109] Clayton Lewis and Donald A Norman. Designing for error. In *Readings in Human–Computer Interaction*, pages 686–697. Elsevier, 1995. 6.3.1

[110] Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Ying-Ju Xia, Shu Zhang, and Hao Yu. Structure-aware review mining and summarization. In *Proceedings of the 23rd international conference on computational linguistics*, pages 653–661. Association for Computational Linguistics, 2010. 1

[111] Hsuan-Tien Lin, Chih-Jen Lin, and Ruby C Weng. A note on plattâĂŹs probabilistic outputs for support vector machines. *Machine learning*, 68(3):267–276, 2007. 3.2.1

[112] Thomas Lin, Patrick Pantel, Michael Gamon, Anitha Kannan, and Ariel Fuxman. Active objects: Actions for entity-centric search. In *Proceedings of the 21st international conference on World Wide Web*, pages 589–598. ACM, 2012. 6.1, 6.2.2

[113] Asha S Manek, P Deepa Shenoy, M Chandra Mohan, and KR Venugopal. Aspect term

extraction for sentiment analysis in large movie reviews using gini index feature selection method and svm classifier. *World wide web*, 20(2):135–154, 2017. 1

[114] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008. 3.1.1, 3.3.6

[115] Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006. 1, 6.1, 6.2.1, 6.2.2

[116] Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006. 5.1, 7.1

[117] Gary J Marchionini, Gary Geisler, and Ben Brunk. Agileviews. In *Proceedings of the ASIST Annual Meeting*, volume 37, pages 271–280, 2000. 6.2.2

[118] Catherine C Marshall and Sara Bly. Saving and using encountered information: implications for electronic periodicals. In *Proceedings of the Sigchi conference on human factors in computing systems*, pages 111–120. ACM, 2005. 5.1

[119] Catherine C Marshall, Morgan N Price, Gene Golovchinsky, and Bill N Schilit. Introducing a digital library reading appliance into a reading group. In *Proceedings of the fourth ACM conference on Digital libraries*, pages 77–84. ACM, 1999. 6.1, 6.2.1, 6.4.2, 7.2

[120] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013. 4.1

[121] Douglas L Medin and Marguerite M Schaffer. Context theory of classification learning. *Psychological review*, 85(3):207, 1978. 3.1

[122] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8. ACM, 2011. 6.2.1, 6.2.2, 6.3, 6.3.1, 6.3.5

[123] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 4.3, 4.3.1

[124] Iris Miliaraki, Roi Blanco, and Mounia Lalmas. From selena gomez to marlon brando: Understanding explorative entity search. In *Proceedings of the 24th International Conference on World Wide Web*, pages 765–775. International World Wide Web Conferences Steering Committee, 2015. 6.1, 6.2.1, 6.2.2, 6.3

[125] Dan Morris, Meredith Ringel Morris, and Gina Venolia. Searchbar: a search-centric web history for task resumption and information re-finding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1207–1216. ACM, 2008. 6.2.2

[126] Meredith Ringel Morris, AJ Bernheim Brush, and Brian R Meyers. Reading revisited: Evaluating the usability of digital display surfaces for active reading tasks. In *Horizontal Interactive Human-Computer Systems, 2007. TABLETOP'07. Second Annual IEEE International Workshop on*, pages 79–86. IEEE, 2007. 5.1

[127] Susan M Mudambi and David Schuff. Research note: What makes a helpful online review? a study of customer reviews on amazon. com. *MIS quarterly*, pages 185–200, 2010. 1, 4.1, 6.1

[128] K O'Hara. Towards a typology of reading goals rxrc affordances of paper project. *Rank Xerox Research Center, Cambridge, UK*, 1996. 6.1, 6.2.1, 6.3.3, 6.4.2, 7.2

[129] Panupong Pasupat and Percy Liang. Zero-shot entity extraction from web pages. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 391–401, 2014. 6.3.2

[130] Emily S Patterson, Emilie M Roth, and David D Woods. Predicting vulnerabilities in computer-supported inferential analysis under data overload. *Cognition, Technology & Work*, 3(4):224–237, 2001. 6.2.2

[131] Jaakko Peltonen, Kseniia Belorustceva, and Tuukka Ruotsalo. Topic-relevance map: Visualization for improving search result comprehension. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 611–622. ACM, 2017. 4.3.3

[132] Jaakko Peltonen, Jonathan Strahl, and Patrik Floréen. Negative relevance feedback for exploratory search with visual interactive intent modeling. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 149–159. ACM, 2017. 4.1, 4.2.1

[133] Sarah E Peterson. The cognitive functions of underlining as a study technique. *Literacy Research and Instruction*, 31(2):49–56, 1991. 5.4

[134] Peter Pirolli and Stuart Card. Information foraging. *Psychological review*, 106(4):643, 1999. 1, 2, 3.1.1, 3.2.1, 5.1, 6.1, 7.1

[135] Peter Pirolli and Stuart Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, volume 5, pages 2–4, 2005. 1

[136] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999. 3.2.1

[137] Henning Pohl and Roderick Murray-Smith. Focused and casual interactions: allowing users to vary their level of engagement. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2223–2232. ACM, 2013. 5.4

[138] The Open Directory Project. DMOZ: The Open Directory Project. https://www.dmoz.org/, http://dmoz-odp.org/, 1998-2017,2017-. Accessed: 2017-09-10,2019-01-13. 7.1

[139] Pradeep Racherla and Wesley Friske. Perceived 'usefulness' of online consumer reviews: An exploratory investigation across three services categories. *Electron. Commer. Rec. Appl.*, 11(6):548–559, November 2012. ISSN 1567-4223. doi: 10.1016/j.elerap.2012.06.003. URL http://dx.doi.org/10.1016/j.elerap.2012.06.003. 1

[140] Reddit. I have a serious problem with browser tab hoarding. https://www.reddit.com/r/declutter/comments/1jpw13/i_have_a_serious_problem_with_browser_tab_hoarding/, 2013. Accessed: 2017-09-10. 7.1

[141] Reddit. I'm a digital hoarder. I opened chrome to find all my tabs gone. I feel relieved. https://www.reddit.com/r/declutter/comments/4qkomc/im_a_digital_hoarder_i_opened_chrome_to_find_all/, 2016. Accessed: 2017-09-10. 7.1

[142] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. http://is.muni.cz/

publication/884893/en. 4.3.5

[143] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009. 4.3.4, 4.4.1

[144] Daniel E Rose and Danny Levinson. Understanding user goals in web search. In *Proceedings of the 13th international conference on World Wide Web*, pages 13–19. ACM, 2004. 6.1

[145] Volker Roth and Thea Turner. Bezel swipe: conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1523–1526. ACM, 2009. 5.1

[146] Yong Rui, Thomas S Huang, Michael Ortega, and Sharad Mehrotra. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on circuits and systems for video technology*, 8(5):644–655, 1998. 4.1

[147] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American society for information science*, 41(4):288–297, 1990. 4.1

[148] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001. 4.5

[149] Bill N Schilit, Gene Golovchinsky, and Morgan N Price. Beyond paper: supporting active reading with free form digital ink annotations. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 249–256. ACM Press/Addison-Wesley Publishing Co., 1998. 2, 5.1

[150] Barry Schwartz. The paradox of choice: Why more is less. Ecco New York, 2004. (document), 1

[151] Julia Schwarz, Jennifer Mankoff, and Scott E Hudson. An architecture for generating interactive feedback in probabilistic user interfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 2545–2554. ACM, 2015. 5.1

[152] Xuehua Shen, Bin Tan, and ChengXiang Zhai. Implicit user modeling for personalized search. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 824–831. ACM, 2005. 4.2.1

[153] Pao Siangliulue, Joel Chan, Steven P. Dow, and Krzysztof Z. Gajos. Ideahound: Improving large-scale collaborative ideation with crowd-powered real-time semantic modeling. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 609–624, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4189-9. doi: 10.1145/2984511.2984578. URL http://doi.acm.org/10.1145/2984511.2984578. 1

[154] Herbert A Simon. Designing organizations for an information-rich world. *The Johns Hopkins Press*, 1971. (document), 1

[155] Rashmi Sinha. A cognitive analysis of tagging, 2005. 5.1

[156] Mirco Speretta and Susan Gauch. Personalized search based on user search histories. In *Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on*, pages 622–628. IEEE, 2005. 4.2.1

[157] Michael Steinbach, George Karypis, Vipin Kumar, et al. A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526. Boston, 2000. 3.1.1

[158] Anselm Strauss and Juliet Corbin. *Basics of qualitative research: Techniques and procedures for developing grounded theory .* Sage Publications, Inc, 1998. 7.1

[159] Jeffrey Stylos, Brad A Myers, and Andrew Faulring. Citrine: providing intelligent copy-and-paste. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 185–188. ACM, 2004. 5.1, 6.2.1

[160] Atsushi Sugiura and Yoshiyuki Koseki. Internet scrapbook: automating web browsing tasks by programming-by-demonstration. *Computer networks and ISDN systems*, 30(1-7):688–690, 1998. 6.2.1

[161] Rohail Syed and Kevyn Collins-Thompson. Optimizing search results for human learning goals. *Information Retrieval Journal*, 20(5):506–523, 2017. 6.3

[162] Omer Tamuz, Ce Liu, Serge Belongie, Ohad Shamir, and Adam Tauman Kalai. Adaptively learning the crowd kernel. In *In ICMLâĂŹ11*, 2011. 2, 3.1.1

[163] Craig S Tashman and W Keith Edwards. Active reading and its discontents: the situations, problems and ideas of readers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2927–2936. ACM, 2011. 5.1

[164] Craig S Tashman and W Keith Edwards. Liquidtext: a flexible, multitouch environment to support active reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3285–3294. ACM, 2011. 2, 5.1, 6.1, 6.2.1, 6.3.3, 6.4.2, 7.2

[165] Jaime Teevan, William Jones, and Benjamin B Bederson. Personal information management. *Communications of the ACM*, 49(1):40–43, 2006. 5.2.1

[166] Jaime Teevan, Susan T Dumais, and Zachary Gutt. Challenges for supporting faceted search in large, heterogeneous corpora like the web. *Proceedings of HCIR*, 2008:87, 2008. 2

[167] Simon Tretter, Gene Golovchinsky, and Pernilla Qvarfordt. Searchpanel: A browser extension for managing search activity. In *EuroHCIR*, pages 51–54, 2013. 6.2.2

[168] Max G Van Kleek, Michael Bernstein, Katrina Panovich, Gregory G Vargas, David R Karger, and MC Schraefel. Note to self: examining personal information keeping in a lightweight note-taking tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1477–1480. ACM, 2009. 2, 6.2.1, 6.3, 6.3.3, 6.4.2

[169] Cornelis J Van Rijsbergen, Stephen Edward Robertson, and Martin F Porter. *New models in probabilistic information retrieval*. British Library Research and Development Department London, 1980. 4.3.5

[170] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proc. ICML 2009*, pages 1073–1080. ACM, 2009. 3.3.1

[171] Ryen W White and Resa A Roth. Exploratory search: Beyond the query-response paradigm. *Synthesis lectures on information concepts, retrieval, and services*, 1(1):1–98, 2009. 1

[172] Ryen W White, Mikhail Bilenko, and Silviu Cucerzan. Studying the use of popular desti-

nations to enhance web search interaction. In *Proc. SIGIR 2007*, pages 159–166. ACM, 2007. 3.1

[173] Barbara M Wildemuth and Luanne Freund. Assigning search tasks designed to elicit exploratory search behaviors. In *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*, page 4. ACM, 2012. 1

[174] Max L Wilson et al. Improving exploratory search interfaces: Adding value or information overload? 2008. 6.3.2

[175] Tom D Wilson. Models in information behaviour research. *Journal of documentation*, 55 (3):249–270, 1999. 1

[176] Ho Chung Wu, Robert Wing Pong Luk, Kam Fai Wong, and Kui Lam Kwok. Interpreting tf-idf term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, 26(3):13, 2008. 4.2.1

[177] Ting-Fan Wu, Chih-Jen Lin, and Ruby C Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5(975-1005):4, 2004. 3.2.1

[178] Jinxi Xu and W Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM, 1996. 4.5

[179] Beverly Yang and Glen Jeh. Retroactive answering of search queries. In *Proceedings of the 15th international conference on World Wide Web*, pages 457–466. ACM, 2006. 4.5

[180] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proc. of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196. Association for Computational Linguistics, 1995. 4.5

[181] Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408. ACM, 2003. 2

[182] Jinfeng Yi, Rong Jin, Anil K Jain, and Shaili Jain. Crowdclustering with sparse pairwise labels: A matrix completion approach. In *AAAI Workshop on Human Computation*, volume 2, 2012. 2, 3.1.1

[183] Jinfeng Yi, Rong Jin, Shaili Jain, Tianbao Yang, and Anil K Jain. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In *Advances in Neural Information Processing Systems*, pages 1772–1780, 2012. 2, 3.1.1

[184] Jianxing Yu, Zheng-Jun Zha, Meng Wang, and Tat-Seng Chua. Aspect ranking: identifying important product aspects from online consumer reviews. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1496–1505. Association for Computational Linguistics, 2011. 1

[185] Oren Zamir and Oren Etzioni. Grouper: a dynamic clustering interface to web search results. *Computer Networks*, 31(11):1361–1374, 1999. 2

[186] Hua-Jun Zeng, Qi-Cai He, Zheng Chen, Wei-Ying Ma, and Jinwen Ma. Learning to cluster web search results. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 210–217. ACM, 2004.

2

[187] Haoqi Zhang, Edith Law, Rob Miller, Krzysztof Gajos, David Parkes, and Eric Horvitz. Human computation tasks with global constraints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 217–226. ACM, 2012. 6.4.1

[188] Shengdong Zhao, Fanny Chevalier, Wei Tsang Ooi, Chee Yuan Lee, and Arpit Agarwal. Autocompaste: auto-completing text as an alternative to copy-paste. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 365–372. ACM, 2012. 5.1

[189] Yuxiang Zhu, David Modjeska, Daniel Wigdor, Shengdong Zhao, et al. Hunter gatherer: interaction support for the creation and management of within-web-page collections. In *Proceedings of the 11th international conference on World Wide Web*, pages 172–181. ACM, 2002. 6.2.1