DESIGNING VISUAL TEXT ANALYSIS METHODS TO SUPPORT SENSEMAKING AND MODELING

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Jason Chia-Chen Chuang

March 2013

This dissertation is online at: http://purl.stanford.edu/zs681pb0632

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Jeffrey Heer, Primary Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Christopher Manning

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Daniel McFarland

Approved for the Stanford University Committee on Graduate Studies.

Patricia J. Gumport, Vice Provost Graduate Education

This signature page was generated electronically upon submission of this dissertation in electronic format. An original signed hard copy of the signature page is on file in University Archives.

Acknowledgements

I would like to thank my advisor, Jeffrey Heer and Christopher Manning, for their continued support and encouragement. I would like to dedicate this dissertation to my parents, my sister, and my grandmother.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This dissertation examines the *co-design* of interactive visualizations and statistical analysis algorithms to *improve the process of visual analytics* — to create effective workflows where human cognition and algorithms work in tandem to yield insights about large and complex data.

Data analysis is an iterative process involving both statistical modeling and human interpretation of the identified patterns. While the two aspects go hand-in-hand, they are often studied in isolation. For example, many researchers treat statistical models as a black box when designing user interfaces for analysis. As a result, many tools are created based on a characterization of domain users and their analysis tasks but without a full consideration of available modeling capabilities. Many efforts focus on multiple prototypes and iterative refinement of the visual elements, but overlook improvements to the statistical models. In these contexts, how do we design *effective visual analysis tools* that fully utilize available modeling capabilities?

Conversely, while many models can be built using automatic or unsupervised learning techniques, applying them to real world analysis often requires intensive human-in-the-loop supervision. To support reasoning, statistical models often need to be manually verified to ensure they are semantically meaningful within the domain of analysis. Eliciting human judgment, however, is a time-consuming task and can dominate the time and cost of building high-quality statistical models. How do we reduce the amount of manual effort involved in the modeling process and *effectively*

design statistical models?

With the growth of big data, the demands for advanced statistical modeling in data analysis are growing. Developments in machine learning have produced increasingly powerful analysis algorithms, but the underlying modeling assumptions and abstractions have also become increasingly opaque to the end users. How do we promote a better understanding and more effective use of statistical analysis methods? How do we design statistical models that are responsive to user needs and support domain-specific applications?

1.1 Thesis

I demonstrate that an *iterative design process* — applied to refine the design of both a visualization and its underlying models — leads to effective data analysis tools. I apply *human-centered design methods* to examine the model-driven analytic workflow. I contribute methods, tools, and frameworks that allow users to more efficiently utilize domain expertise to assess model outputs and explore modeling options.

Visual analytics also contributes to the selection and evaluation of machine learning methods. Visualizations enable detailed inspection of model output, and help researchers understand properties of their models. Interactivity allows more rapid design cycles, so researchers can explore more design options. Visual tools also improve communication between model builders and users. Incorporating feedback from analysts and assessments by domain experts help create models that are more responsive to analysis needs and more accurately reflect domain expertise.

In Section 1.2, I discuss my focus on model-driven text analysis. In Section 1.3, I examine the human-centered iterative design process in greater detail, and outline how it corresponds to the relevant sections of this thesis. I summarize my contributions in Section 1.4.

1.2 Problem Domain

While many aspects of my thesis apply to all areas of visual analytics, for my thesis work, I focus primarily on the visualization and analysis of text data. Despite text being an abundant data type relevant to studies of human culture and communication, unstructured text is a notoriously difficult data type to analyze.

Our interpretation of a passage of text depends heavily on context, such as the domain of analysis or our prior knowledge about a subject matter. While textual understanding often requires reading the source document, for many real world analysis tasks, analysts are faced with document collections too large for any single person to read and must rely on statistical methods to guide their analysis.

To identify large-scale patterns in textual data, analysts typically need to first transform a text corpus into numeric formats suitable for statistical analysis. Analysts spend a significant amount of time building models — applying a long chain of text processing to extract relevant language features. To ensure that the models capture meaningful concepts situated in the appropriate context, analysts may consult domain experts who would scrutinize and validate the model outputs. The subjective nature of textual interpretation can thus complicate the construction, deployment, and evaluation of text analysis tools.

I focus on model-driven text analysis because it presents us with not only a rich set of research challenges but also an opportunity to impact many real-world analysis practices.

1.3 Human-Centered Iterative Design Process

Across multiple projects, this dissertation demonstrates the value of applying a human-centered iterative design process to the task of model formulation. I focus on four of the tools that have resulted from my projects, and identify the common components critical to the design process.

I begin each of my projects by characterizing effective *human strategies*. I explore the space of potential *visual designs* and available *analysis algorithms*. In particular,

I identify analysis tasks that can be aided by interactive visualizations and/or are amenable to statistical modeling. I then refine both the visualizations and the underlying models through an *iterative design process*. I discuss various relevant *evaluation measures* I applied to determine the appropriateness of my modeling assumptions and to assess the effectiveness of the resulting tools.

1.3.1 The Stanford Dissertation Browser

I developed a set of visualizations to help social scientists explore various academic publication datasets, including the Stanford Dissertation Browser (§3.2), a topic flow visualization tool (§3.3.1), and a visualization of language transfer in academia (§3.3.2). Here, I focus on the first tool, the Dissertation Browser.

At the start of the project, I conducted interviews with the social scientists (§3.2.1) to understand their analysis needs and identify the concepts suitable for modeling and visualization. My collaborating machine learning researchers proposed various modeling techniques (§3.2.2). I initially applied 2D projection to visualize large-scale patterns in the model output (§3.2.3). As the project progressed, we devised a novel "word borrowing" topical similarity measure (§3.2.5), removed a problematic "landscape" view (§3.1.1), and introduced two additional views. The model and visual design changes were informed by evaluations by domain experts (§3.2.4). I anecdotally noted the positive public reception of the system (§3.2.6).

1.3.2 Termite: Visualizations for Assessing Topical Quality

Working alongside the social scientists and machine learning researchers, in the aforementioned series of collaborations, provided me with extensive in-field observations (§3.2, §3.3.1, §3.3.2) of real-world model-driven analytic workflows. A significant amount of our analysis effort was spent on model design, as the social scientists worked closely with machine learning researchers to experiment with different modeling options. The experience helped me recognize the need to incorporate model design into the iterative design process. The experience also drew attention to time-consuming but recurring analysis tasks, such as manual verification of the extracted

topics (§3.5.1).

Leveraging interactive visualization, I developed Termite to enable more rapid assessment of topical quality. During the development of Termite (§3.5.2), I examined available text processing techniques and devised a saliency measure to promote informative vocabulary that can aid topical comparisons. I developed a novel seriation algorithm to reveal the clustering of related words and improve the readability of phrases in the visualization. I assessed the performance of Termite through informal user feedback (§3.5.3).

1.3.3 Model Diagnostics via Topical Alignment

I then examined how we might reduce the cost of acquiring domain expertise and increase its utilization in the modeling process. I began by characterizing how human experts (§4.2.1) topically organize information (§4.3.2). To acquire the dataset, I contributed a survey method (§4.2.2) as well as a method for synthesizing participant responses (§4.3.1) and a corresponding method for validating the combined results. I then assessed how well various topic models (§4.4.1) captured the expert concepts (§4.4.2) in terms of shared mutual information.

Recognizing that data analysis often involves the use of multiple statistical models, I devised the correspondence chart (§4.5.1), a visualization showing how a set of statistically extracted topics aligned with the ground truth (or a set of known reference concepts). The chart provided diagnostic feedback (§4.1.1) on how a topic model differed from expectation.

To enable large-scale assessment, I introduced a framework to determine the correspondence between any number of topics models with a common set of reference concepts. I began by obtaining human ratings of topical similarity (§4.5.2). I evaluated how well various similarity measures captured user judgments, and developed a novel technique that outperformed existing measures (§4.5.3, §4.5.4). I devised a method to improve the numerical robustness of my approach (§4.5.5), and demonstrated its effectiveness through a use case where I identified suitable modeling settings from over 10,000 parameter choices.

1.3.4 Text Summarization Using Descriptive Phrases

To investigate how visualizations can better convey summary information about a document collection, I began by looking into how people summarize text. I conducted a formal user study (§5.2.1, §5.2.2) to collect examples of human-generated keyphrases subject to three experimental conditions (§5.2.3). I examined the statistical and linguistic properties (§5.2.4) of descriptive phrases chosen by human judges. I then systematically assessed available computational features, identified ones predictive of high-quality keyphrases, and contributed a novel keyphrase extraction algorithm.

I compared my technique to existing algorithms using a benchmark keyphrase dataset (§5.3.3). While the performance of my algorithm matched that of existing algorithms, further examination suggested that the standard precision-recall measures did not fully reflect how people judge keyphrase quality (§5.3.4). In response, I introduced two post-processing steps: redundancy reduction (§5.4.1) and length adjustment (§5.4.2). I evaluated my improved technique through both inspection (§5.4.3) and crowdsourced ratings of tag cloud visualizations (§5.4.4) to obtain more ecologically valid evaluations. I demonstrate novel interactions that were afforded by my technique (§5.5).

1.4 Summary of Contributions

In Chapter 2, I review relevant literature (§2.1, §2.2) in information visualization, machine learning, and natural language processing. I examine and discuss previous work in three areas of text visualization (§2.3): text summarization with word clouds, document visualization with statistical topic models, and investigative analysis of entity-relation networks.

In Chapter 3, I examine how to effectively visualize statistical topic models. In a series of collaborations with social scientists and machine learning researchers, we apply topic modeling to study large-scale academic discourse [100, 126] (§3.1). I describe my experiences in three projects [34, 37] (§3.2, §3.3) involving different models and visual representations. Across these projects, the analysts spend a significant

amount of time modifying and validating the topic models, to ensure that the uncovered topics accurately reflect concepts in the domain of analysis [130]. I formulate two design principles, *interpretation* and *trust*, that are critical in supporting an analyst's ability to comprehend and interrogate observed patterns predicted by a model. I also present a set of design processes — *align*, *modify*, *verify*, and *progressive disclose* — and discuss their use in successful visual analysis tools. Finally, I develop Termite [35] (§3.5), a visual analysis tool for assessing topical quality.

In Chapter 4, I examine how human experts topically organize text corpora and how we can utilize domain knowledge to more efficiently design topic models. While fitting topic models typically involves unsupervised learning algorithms, incorporating these models into real-world data analysis requires a significant amount of human-in-the-loop supervision. Analysts often need to repeatedly verify, compare, and modify multiple models throughout their analytic workflow. Such manual judgment tasks are time-consuming and can dominate the time and cost of model-driven data analysis. I ask experienced researchers in information visualization to describe their field (§4.2) and analyze their responses (§4.3). I evaluate current modeling practices (§4.4) by comparing their outputs against expert categorizations. I develop a framework that enables large-scale assessment of topical relevance [33] (§4.5) and demonstrate how it can contribute to machine learning research (§4.6).

In Chapter 5, I examine text summarization using descriptive phrases [36]. I demonstrate that a human-centered iterative design process can improve the design of tools in domains beyond statistical topic modeling. I begin my investigation by analyzing how people select descriptive phrases to summarize documents (§5.2). I systematically examine computational features predictive of high-quality keyphrases (§5.3), and embed them within predictive statistical models. I contribute a novel keyphrase extraction algorithm where the specificity of the output terms can be dynamically adjusted (§5.4). The improved technique, in turn, enables novel interactive visualizations (§5.5).

Chapter 2

Current Approaches to Model-Driven Text Analysis

A rich and growing literature considers the use of statistical modeling methods to drive text visualizations and text analyses. Many techniques, such as tag clouds [163], analyze documents by their constituent words to support impression formation [177], augment search [146], reveal language structure [161, 170], or aid document comparison [41, 42]. Other tools infer latent topics [58, 59, 61, 62, 113, 152, 171], sentiment [13, 121, 168], or word relationships (e.g., overlap [147], clustering [69, 79], or latent semantics [44, 86]) from text. For large document collections, a common approach is to model thematic patterns in the corpus and visually convey uncovered patterns via dimensionality reduction [18, 19, 31, 92, 132, 174, 175]. A related literature concerns "science mapping" [14, 16, 17, 48, 138, 142], often via 2D projection of academic citation networks.

In this chapter, I review relevant literature on statistical topic models and visualization systems for exploring large text corpora. I also review the use of keyphrases for text summarization. I then examine in greater detail three classes of visual analysis tools. I analyze their visual designs and modeling assumptions, and discuss how they relate to analysis tasks.

2.1 Statistical Topic Modeling

Statistical topic models enable the exploration of large document collections by identifying co-occurring words that can capture thematic patterns. In this section, I look at Latent Dirichlet allocation, a popular topic modeling technique, examine issues with its deployment in real-world analysis tasks, and review relevant literature on how human experts organize topical information.

2.1.1 Latent Dirichlet Allocation Models

Latent Dirichlet allocation (LDA) [11] and its variations [9, 10, 128] are statistical models that extract *latent topics*, probability distributions of words that tend to co-occur in a corpus, and represent documents as topical mixtures. An active research area in machine learning, these models have been applied to examine language in social media [127], medical literature [113], academic publications [52], and even inventories of household items in the ruins of Pompeii [105]. More recently, they have also been incorporated into various visual analysis tools [58, 59, 171].

2.1.2 Topical Quality and Expert Verification

While LDA can produce some sensible topics, a prominent issue is the presence of *junk topics* [1, 110] comprised of general and uninformative terms. LDA modeling parameters are often chosen to minimize the perplexity of held out data. However, recent studies show that perplexity score does not match human judgment of topics [106, 116, 115].

LDA models are sensitive to the choice of parameter N, the number of latent topics to learn [61]. Choosing N involves a trade-off between topic quality and resolution [152]. A large value of N produces small and noisy topics due to insufficient data. A small value of N generates generic topics that do not have sufficient details for in-depth analysis. Both types of issues can occur at the same time; as the value N increases, the larger topics might still be too generic while the smallest topics already begin to take on nonsensical words. In addition, LDA models require the use of two

smoothing hyperparameters (α and β); unsuitable values can also affect the quality of topics [71, 5].

Large-scale applications of topic modeling in the real world often involve human experts in the loop. Talley et al. [152] examined 110,000 NIH grants over four years, and applied LDA to uncover 700 latent topics in the corpus. The authors documented steps taken to train and refine their model including: identification of stop words, resolution of acronyms, parameter search, manual removal of nonsensical topics, and retraining the model. Hall et al. [62] studied the history of academic discourse in Computational Linguistics over forty years by examining 14,000 papers published at multiple conferences. The authors applied LDA to analyze topical trends over time, and recruited experts to assess the quality of every topic. The experts retained 36 out of 100 automatically discovered topics, and manually inserted 10 additional topics not produced by the model. The need to validate and modify model outputs are challenging issues in application of statistical models, not only limited to LDA. For example, the Guardian Newspaper [121, 120] analyzed the spread of misinformation on London riot on Twitter based on sentiment. To ensure reliability of their algorithmic analysis, each of the 2.6 million tweets was then independently coded by three sociology Ph.D. students.

Most recently, researchers have proposed several automatic methods for assessing topical quality, by comparing topical word distributions with word occurrence in other reference corpora (e.g., Wikipedia, WordNet, etc.) [116, 110], based on alternative statistical measures (e.g., pointwise mutual information, etc.) [1, 167], or indirectly via intrusion tests [28]. Some recent variations of LDA models recognize the drawbacks of a completely automatic approach, and enable users to explicitly incorporate domain knowledge via labeled topics [128] or domain constraints [4]. Other techniques aid interpretation by automatically labeling topics [87, 102]. Ultimately, however, how meaningful a topic is depends on the user and task [172]. Assessing relevance of model output to a task requires human expertise.

2.1.3 Expert Categorization

Experts, through years of experience, develop effective strategies for solving problems in their domain of expertise. Psychologists comparing task performance between experts and novices have examined a wide range of professions from world-class chessmasters [30] to taxi drivers [29], and a wide range of tasks from medical diagnoses [56] to UNIX system operations [49]. These studies repeatedly demonstrate that experts exhibit efficient mental representation via information chunking [97, 103] and utilize categorizations [32] that support reasoning [160].

Previous research has established various theories on human categorization. Rosch et al. [137] observe that people create a *hierarchy of categories*, and that categories are created starting at a basic level before super (more general) and sub-ordinate (more specific) categories emerge. *Basic level categories* exhibit computationally favorable properties such as the strongest within-category similarity and between-category dissimilarity; they yield features [159] that are most amenable to codification [137], and enable more effective communication [43]. Members of a category exhibit a gradient of membership. *Prototypes* [84, 136] refer to common exemplars of a category, correspond to more stable and salient concepts, and are sometimes used as labels to help communicate the content of a category [135].

Expert categories can exhibit differing characteristics. Tversky shows that novices create categories based on easy-to-detect features while experts create categories based on features functionally significant to a task [160]. Tanaka et al. find that expertise increases the use of subordinate categories [153].

2.2 Visualizations for Text Summarization

Descriptive phrases aid the exploration of text collections by communicating salient aspects of documents. In this section, I examine existing text visualizations that use keyphrases to display text summaries, paying particular attention to how they select keyphrases. I compare their approaches to automatic keyphrase extraction algorithms developed by the natural language processing community. I discuss the need of more

suitable keyphrase extraction algorithms to support interactive visualizations.

2.2.1 Selection of Descriptive Terms in Text Visualizations

Many text visualization systems use descriptive keyphrases to summarize text or label abstract representations of documents [24, 42, 45, 64, 68, 144, 162, 164]. One popular way to represent a document is as a tag cloud: a list of descriptive words typically sized by raw term frequency. Various interaction techniques summarize documents as descriptive headers for efficient browsing on mobile devices [22, 23, 176]. While information visualization researchers have developed methods to improve the layout of terms [45, 164], they have paid less attention to methods for *selecting* the best descriptive terms.

Visualizations including Themail [162] and TIARA [144] display terms selected using variants of tf.idf (term frequency by inverse document frequency [140]) — a weighting scheme for information retrieval. Rarely are more sophisticated methods from computational linguistics used. One exception is Parallel Tag Clouds [42], which weight terms using G^2 [51], a probabilistic measure of the significance of a document term with respect to a reference corpus.

Other systems, including Jigsaw [148] and FacetAtlas [24], identify salient terms by extracting named entities such as people, places, and dates [57]. These systems extract specific types of structured data, but may miss other descriptive phrases. In this paper, I first score phrases independent of their status as entities, but later apply entity recognition to group similar terms and reduce redundancy.

2.2.2 Automatic Keyphrase Extraction

As indicated above, the most common means of selecting descriptive terms is via bag-of-words frequency statistics of single words (unigrams). Researchers in natural language processing have developed various techniques to improve upon raw term counts, including removal of frequent "stop words," weighting by inverse document frequency as in tf.idf [140] and BM25 [134], heuristics such as WordScore [88], or probabilistic measures [81, 131] and the variance-weighted log-odds ratio [108]. While

unigram statistics are popular in practice, there are two causes for concern.

First, statistics designed for document retrieval weight terms in a manner that improves search effectiveness, and it is unclear whether the same terms provide good summaries for document understanding [12, 42]. For decades, researchers have anecdotally noted that the best descriptive terms are often neither the most frequent nor infrequent terms, but rather mid-frequency terms [94]. In addition, frequency statistics often require a large reference corpus and may not work well for short texts [12]. As a result, it is unclear which existing frequency statistics are best suited for keyphrase extraction.

Second, the set of good descriptive terms usually includes multiword phrases as well as single words. In a survey of journals, Turney [158] found that unigrams account for only a small fraction of human-assigned index terms. To allow for longer phrases, Dunning proposed modeling words as binomial distributions using G^2 statistics to identify domain-specific bigrams (two-word phrases) [51]. Systems such as KEA++ or Maui use pseudo-phrases ("phrases" that remove stop words and ignore word ordering) for extracting longer phrases [101]. Hulth considered all trigrams (phrases up to length three) in her algorithm [74]. While the inclusion of longer phrases may allow more expressive keyphrases, systems that permit longer phrases can suffer from poor precision and meaningless terms. The inclusion of longer phrases may also result in redundant terms of varied specificity [53], such as "visualization," "data visualization," and "interactive data visualization."

Researchers have taken several approaches to ensure that longer keyphrases are meaningful and that phrases of the appropriate specificity are chosen. Many approaches [6, 47, 53, 74] filter candidate keyphrases by identifying noun phrases using a part-of-speech tagger or a parser. Of note is the use of so-called *technical terms* [78] that match regular expression patterns over part-of-speech tags. To reduce redundancy, Barker [6] chooses the most specific keyphrase by eliminating any phrases that are a subphrase of another. Medelyan's KEA++ system [101] trains a Naïve Bayes classifier to match keyphrases produced by professional indexers. However, all existing methods produce a *static* list of keyphrases and do not account for task- or application-specific requirements.

2.2.3 Supporting Interactive Visualizations

Recently, the Semantic Evaluation (SemEval) workshop [80] held a contest comparing the performance of 21 keyphrase extraction algorithms over a corpus of ACM digital library articles. The winning entry, named HUMB [93], ranks terms using bagged decision trees learned from a combination of features including frequency statistics, position in a document, and the presence of terms in ontologies (e.g., MeSH, WordNet) or in anchor text in Wikipedia. Moreover, HUMB explicitly models the structure of the document to preferentially weight the abstract, introduction, conclusion, and section titles. The system is designed for scientific articles and intended to provide keyphrases for indexing digital libraries.

The aims of my research are different. Unlike prior work, I seek to systematically evaluate the contributions of individual features to keyphrase quality, allowing system designers to make informed decisions about the trade-offs of adding potentially costly or domain-limiting features. I have a particular interest in developing methods that are easy to implement, computationally efficient, and make minimal assumptions about input documents.

My goal is to improve the design of text visualization and interaction techniques, not indexing of digital libraries. This orientation has led me to develop techniques for improving the quality of extracted keyphrases as a whole, rather than just scoring terms in isolation (c.f., [6, 158]). I propose methods for grouping related phrases that reduce redundancy and enable applications to dynamically tailor the specificity of keyphrases. I also evaluate my approach in the context of text visualization.

2.3 Visual Designs, Model Abstractions, and Analysis Tasks

I now discuss in greater detail a subset of the existing work on model-driven text analysis. I choose three classes of visual analysis tools due to their widespread use and significant research attention: summaries via word clouds, document visualization using latent topic models, and investigative analysis of entity-relationship networks.

I pay particular attention to visual designs and model abstractions, and discuss how they relate to analysis tasks.

2.3.1 Text Summarization with Word Clouds

Word clouds are a popular visualization method used to summarize unstructured text. A typical word cloud shows a 2D spatial arrangement of individual words with font size proportional to term frequency. Despite documented perceptual issues [133], word clouds are regularly found both in analysis tools and across the web [163]. Though simple, a word cloud rests on a number of modeling assumptions. Input text is typically treated as a "bag of words": analyses focus on individual words ignoring structures (e.g., word position, ordering) and semantic relationships (e.g., synonym, hypernym). Most implementations assume raw term counts are a sufficient statistic for indicating the importance of terms in a text.

The ostensible goal of most word clouds is to provide a high-level summary of a text. Is the visualization well suited for the task? A strength of word clouds is that they are highly *interpretable* and directly display the *units of analysis*, words and word-level statistics. Users can readily assess word distributions and identify key recurring terms. Studies found summary information provided by a word cloud can help form meaningful impressions [38] and answer broad queries [146].

To enable more specialized tasks, however, changes are required to the underlying language model. For decades, researchers have anecdotally noted that the most descriptive terms are often not the most frequent terms [94]. Significant absence of a word can be a distinguishing indicator of a document's content relative to a corpus. To better support document comparison, Parallel Tag Clouds [42] apply G^2 statistics to surface both over- and under-represented terms. Others note that single words account for only a small fraction of descriptive phrases used by people [158]. To better capture sentiment in restaurant reviews, Review Spotlight [177] extends the bag-of-words model to consider adjective-noun pairs ("great service" vs. "poor service", instead of just "service"). By modifying the unit of analysis, the tool improves impression formation while retaining a familiar visual design.

In-depth analyses may require more than inspection of individual words. Analysts may want additional context in order to *verify* observed patterns and *trust* that their interpretation is accurate. For example, does the presence of the word "matrix" indicate an emphasis on linear algebra, the use of matrices to represent network data, or a scatterplot matrix for statistical analysis? Interactive techniques can provide *progressive disclosure* across modeling abstractions, e.g., selecting a word in a cloud can trigger highlighting of term occurrences in a view of the source text. In other tools, changes in visual design are accompanied by corresponding changes in the model. WordTree [170] discloses all sentences in which a term occurs using a tree layout. Taking into account the frequency of adjacent terms, WordTree expands branches in the tree to surface recurring phrase patterns. DocuBurst [41] applies radial layout to show word hierarchy; the tool infers word relationships by traversing the WordNet hypernym graph.

2.3.2 Document Visualization with Statistical Topic Models

A growing body of visual analytics research attempts to support document understanding using topic modeling. Latent Dirichlet allocation (LDA) [11] is a popular method of discovering latent topics in a text corpus by automatically learning distributions of words that tend to co-occur in the same documents. Given as input a desired number of topics N and a set of documents containing words from a vocabulary V, LDA derives N topics β_k, each a multinomial distribution over words V. For example, a "physics" topic may contain with high probability words such as "optical," "quantum," "frequency," "laser," etc. Simultaneously, LDA recovers the per-document mixture of topics α_d that best describes each document. For example, a document about using lasers to measure biological activity might be modeled as a mixture of words from a "physics" topic and a "biology" topic.

Latent topics are often presented to analysts as a list of probable terms [28], which imposes on the analysts the potentially arduous task of inferring meaningful concepts from the list and verifying that these topics are responsive to their goals. In this case, modeling abstraction increases the gulf of evaluation [75] required to interpret

the visualization.

Evaluations of existing visualizations indicate that an analysis of "topical concepts" can provide an overview of a collection [46], but that the value of the model decreases when the analysis tasks become more specific [79]. Beyond "high-level understanding," many existing systems (e.g., [59, 171]) stop short of identifying specific analysis tasks or contexts of use. This omission makes it difficult to assess their utility.

Notable issues of *trust* arise in the application of topic models to specific domains. Talley et al. [152] examined the relationships between NIH-supported research and NIH funding agencies. To characterize research output, the authors applied LDA to uncover 700 latent topics in 110,000 grants over a four-year period. To *verify* that the topics accurately capture significant research fields, the authors manually rated individual topics and noted the presence of a large number of "junk" or nonsensical topics. The authors *modified* the model by removing 1,200 non-informative words from the analysis and inserting 4,200 additional phrases. The authors then performed extensive parameter search and removed poor topics from the final model before incorporating model output into their analysis. Hall et al. [62] studied the history of Computational Linguistics over forty years. The authors applied LDA on 14,000 papers published at multiple conferences to analyze research trends over time, and recruited experts to verify the quality of every topic. The experts retained only 36 out of 100 automatically discovered topics, and manually inserted 10 additional topics not produced by the model. In many real-world analyses, extensive research effort is spent on validating the latent topics that support the analysis results.

2.3.3 Investigative Analysis of Entity-Relation Networks

One particularly successful class of visual analysis tools uses entity-relation models to aid investigative analysis. In the context of intelligence analysis, "entities" may include people, locations, dates, and phone numbers; "relationships" are modeled as connections between them. Example systems include FacetAtlas [24], Jigsaw [149] (a VAST'07 challenge winner), and Palantir [119] (a VAST'08 challenge winner).

In contrast to other text visualization systems, these tools exhibit clearly-defined

units of analysis and provide strong support for model verification, model modification, and progressive disclosure of model abstractions. First, the units of analysis (people, places, events) are *well-aligned to the analysis tasks*. The entity-relationship model provides an interpretable analytical abstraction that can be populated by statistical methods (e.g., using automated entity extraction [57]) and modified by manual annotations (e.g., selecting terms in source text) or other override mechanisms (e.g., regular expressions). Jigsaw uses a simple heuristic to determine relations among entities: co-occurrence within a document. This model assumption is readily interpretable and verifiable, but might be revisited to infer more meaningful links. To foster trust, Palantir provides an auditable history for inspecting the provenance of an observed entity or relation.

Progressive disclosure, particularly in the form of linked highlighting, is used extensively by both Jigsaw and Palantir to enable scalable investigation and verification. According to Jigsaw's creators, the "workhorses" of the tool are the list view (which groups entities by type and reveals connections between them) and the document view (which displays extracted entities within the context of annotated source text). In contrast, Jigsaw's cluster view receives less use, perhaps due to the interpretation and trust issues inherent in assessing an arbitrary number of automatically-generated groupings.

Across these examples, I note that successful model-driven visualizations exhibit relevant *units of analysis* responsive to delineated *analysis tasks*. However, I also find that many text visualizations fail to align model abstractions with real-world tasks; iterative design often considers interface elements, but not modeling choices. These observations emphasize a recurring lack of attention to model design and a need for principled approaches. I revisit these three classes of text visualizations in Section 3.4, and present a set of process-oriented design guidelines for model-driven systems.

Chapter 3

Visualizing Statistical Topic Models

Statistical topic models enable the exploration of large document collections by identifying co-occurring words that can capture thematic patterns. To gain actionable insights from the modeling results, analysts often need to first verify that the uncovered topical concepts are semantically meaningful within the domain of analysis.

In this chapter, I introduce *interpretation* and *trust*, two design principles for creating effective model-driven visualizations. I demonstrate that model design is just as critical as visual design in determining the effectiveness of a visual analysis tool. A user-centered iterative design process must consider the two aspects together; doing so can lead to improvements in both. Through a series of collaborations with social scientists and machine learning researchers, I applied topic modeling to study large-scale academic discourse. I describe my experiences in three projects involving different models and visual representations: the Stanford Dissertation Browser, a topic flow visualization tool, and a visualization of language transfer in academia. Finally, in response to the recurring need to inspect inferred topics, I introduce Termite, a visual analysis tool for assessing topical quality.

3.1 Model-Driven Visualizations

Analysts often use both statistical models and data visualizations to help them make sense of large and complex data. Models are abstractions that represent data in terms of entities and relationships relevant to a domain of inquiry. Visual representations may depict a model, source data, or a combination of both. A central goal of visual analytics research is to augment human cognition by devising new methods of coupling data modeling and interactive visualization [155].

Statistical modeling — and model-driven visualizations on top of which they are built — can greatly increase the scale of an analysis by automatically extracting patterns from data, based on assumptions about structures in the data. While model abstractions should ideally correspond to analysts' mental models of a domain to aid reasoning, unsuitable or unfamiliar abstractions can impede interpretation. Moreover, reliable discoveries depend on analysts' ability to scrutinize both data and model and to verify that a visualization shows real phenomena rooted in appropriate model assumptions. Abstractions, however, can prevent an analyst from inspecting the underlying computation or data transformations backing an observation.

I begin this chapter with the following example to illustrate the potentials and pitfalls of conducting visual analyses through modeling abstractions.

3.1.1 The Curious Case of Petroleum Engineering

Consider the visualizations in Figure 3.1, which depict "topical similarity" between university departments in terms of their published Ph.D. dissertations. We fit a statistical topic model (latent Dirichlet allocation or LDA [11]) to the text and compute topical similarity between departments (based on cosine similarity between topic vectors that represent each department).

In the top view, we project departments onto a two dimensional plane based on principal component analysis (PCA) projection of a matrix of all pairwise topical similarities. Using this visualization we note an unexpected trend. Over the years, Petroleum Engineering pulls away from other engineering departments, and by 2005, it is situated between Neurobiology, Medicine, and Biology. This observation comes

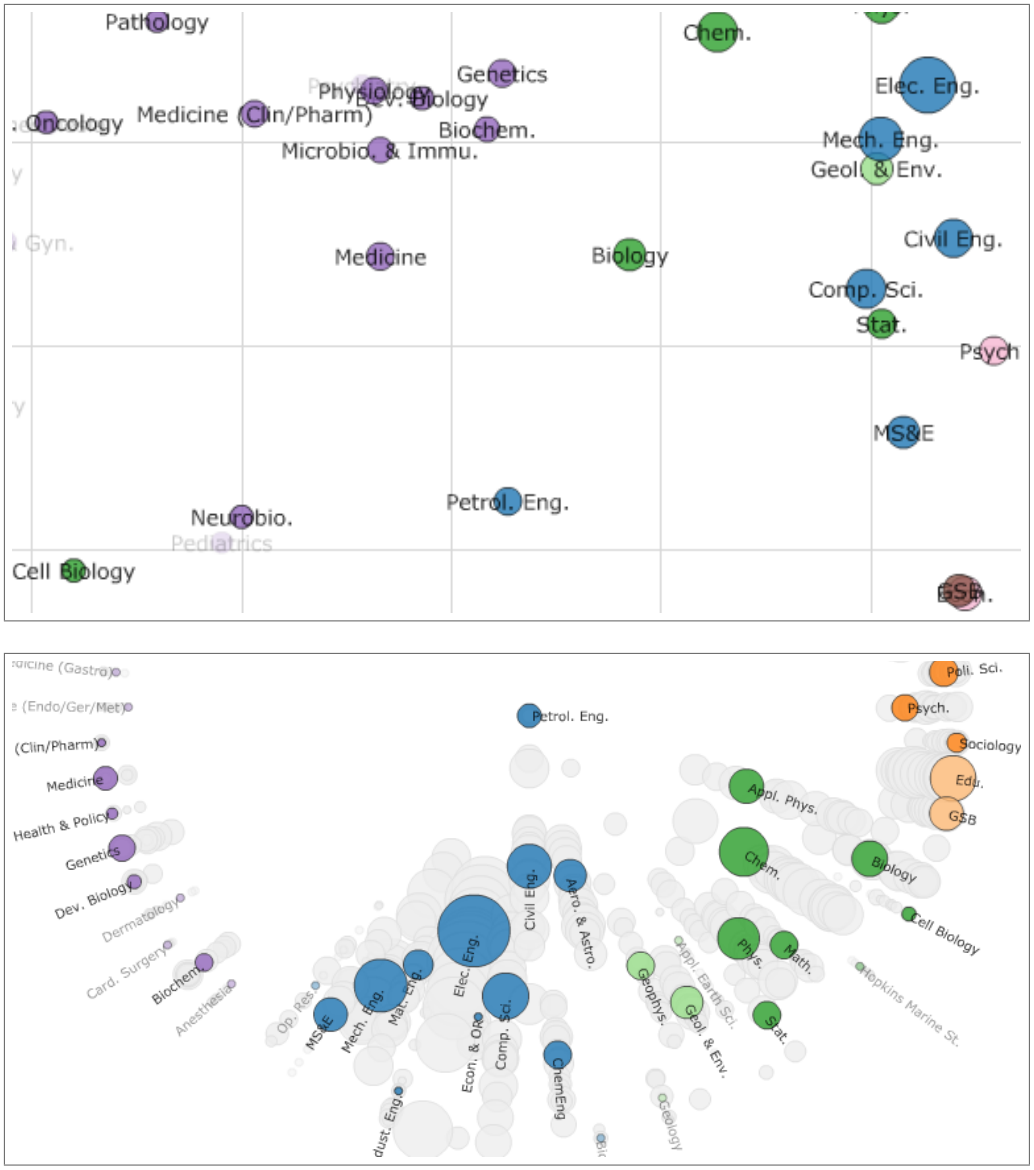

Figure 3.1: The curious case of Petroleum Engineering. The top visualization shows a 2D projection of pairwise topical distances between academic departments. In 2005, Petroleum Engineering appears similar to Neurobiology, Medicine, and Biology. Was there a collaboration among those departments? The bottom visualization shows the undistorted distances from Petroleum Engineering to other departments by radial distance. The connection to biology disappears: it was an artifact of dimensionality reduction. The visual encoding of spatial distance in the first view is *interpretable*, but on its own is not *trustworthy*.

easily as the visualization is readily *interpretable*; pixel distance on the screen ostensibly represents topical similarity. However, the display is the result of a chain of transformations — topic modeling, similarity measures, and dimensionality reduction. Can an analyst *trust* the observed pattern?

The bottom view instead shows undistorted distances from Petroleum Engineering to the other departments. The relationship with Biology evaporates; it is an artifact of dimensionality reduction. Stripping a layer of model abstraction (in this case, PCA projection) enables validation and disconfirms the initial insight.

3.1.2 Chapter Outline

In this chapter, I introduce interpretation and trust, two design considerations for model-driven visual analysis. I define interpretation as the *facility with which an analyst makes inferences about the underlying data* and trust as the *actual and perceived accuracy of an analyst's inferences*. As illustrated by the Petroleum Engineering example, designs lacking in interpretation or trust can restrict an analyst's ability to generate and validate insights derived from an analysis.

In Section 3.2, I introduce the Stanford Dissertation Browser, a visual analysis tool for exploring over 9,000 Ph.D. dissertations published at Stanford University by topical similarity. A goal of the tool is to enable the investigation of shared ideas and interdisciplinary collaboration among the academic departments at the university. We initially envisioned an interface using existing statistical models. However, we quickly arrived at a working visualization that revealed unexpected shortcomings in the underlying model. Our design work instead involved close collaboration with machine learning and natural language processing researchers to develop and evaluate models that better supported our analysis goals. We describe our experience of building the Dissertation Browser, drawing attention to issues of interpretation and trust as well as highlighting successful design decisions. We contribute a novel similarity measure for text collections based on the notion of "word-borrowing" and show how it arose from our iterative design process.

In Section 3.3, I provide selected examples from two additional projects that incorporate different models and visual representations, and highlight the recurring need for external validation in model-driven analyses. My topic flow visualization tool initially caused interpretation issues due to a difference between how the experts and the model assign importance to a citation graph. An improved visualization allowed experts to identity unexpected trends which revealed ungrounded assumptions made by the model. We then examined language transfer based on three decades of academic discourse. My tools allowed my collaborators to estimate model stability, test alternative hypotheses, and verify their discoveries based on an analysis of over one million Ph.D. dissertations.

Finally, I investigate how visualization can aid topic model assessment in Section 3.5. I present Termite, a visual analysis tool for examining the topical term distributions produced by a statistical topic model. I contribute two novel techniques. First, I describe a saliency measure for ranking and filtering terms. By surfacing more discriminative terms, my measure enables faster assessment and comparison of topics. Second, I introduce a seriation method for sorting terms to reveal clustering patterns. My technique has two desirable properties, preservation of term reading order and early termination when sorting subsets of words. I demonstrate how these techniques enable rapid classification of coherent or junk topics and reveal overlap among topics.

3.2 The Design of a Dissertation Browser

As part of the Stanford MIMIR Project, we were tasked with investigating the impact of interdisciplinary collaboration at Stanford University. Our approach adopted the idea that we could identify influences and convergent lines of research across disciplines by detecting shared language use within university-wide publications. Manually reading the document collection is infeasible due to both the size of the corpus and the expertise required to discern topical overlap between papers. The project also receives the attention of university administrators who wish to evaluate the effectiveness of various research institutes on campus. Do multi-million dollar collaborative centers

return suitable intellectual dividends? As a part of the collaboration, I designed the Stanford Dissertation Browser, a visual analysis tool for exploring 16 years of Ph.D. theses from 75 academic departments.

3.2.1 Identifying the Units of Analysis

The social scientists hypothesized that interdisciplinary collaborations foster high-impact research, and wanted to identify ideas that might bridge disciplines. For example, they posited that statistical methods are topically situated at the center of the sciences and engineering. What data, models and representations would enable rapid assessment of such hypotheses? We began by collecting 16 years of dissertation abstracts, for which text and metadata were readily available.

Early conversations with my collaborators emphasized the need to examine large scale patterns in the university's research output. A first step toward that goal is to survey research at a "disciplinary" level. Such a survey might suggest areas of horizontal knowledge transfer — such as the application of theory, methodology, or techniques across domains — that could then be verified as interdisciplinary collaborations. Because each department approximately acts as its own discipline, the university's 75 *academic departments* were suggested as a sensible baseline unit of analysis. Each department's school (such as Engineering or Medicine) provides further organizational context that is meaningful to my collaborators and target audience within the university. A visualization that demonstrates which departments share content would allow my collaborators to focus on unexpected areas of inter-disciplinary collaboration and verify known ones.

My collaborators also emphasized the need to assess the impact of interdisciplinary initiatives, which requires tracking the topical composition of involved groups over time. My collaborators want to correlate change in research output to the formation of academic ties that cross disciplinary boundaries, such as the creation of research institutes, joint grant proposals, and co-authorship. *Time*, in this case the year of filing, is therefore necessary for the analysis tasks.

Textual similarity provides one means of identifying which disciplines are sharing

information. Because each dissertation is associated with one or more departments, the content of these dissertations was seen as a reasonable basis for inferring whether two departments are working on the same content as seen through the words in their published dissertations. We thus explored various text-derived similarity measures as the basis of these similarity scores.

3.2.2 Data and Initial Models

Our dataset contains abstracts from 9,068 Ph.D. dissertations from Stanford University published from 1993 to 2008. These dissertations represent over 97% of all Ph.D. degrees conferred by Stanford during that time period. The text of the abstract could not be recovered for the remaining 263 dissertations. The advisor and department of each dissertation are included as metadata as well as the year of each publication. The abstracts average 181 words in length after tokenization, case-folding, and removal of common stop words and very rare terms (occurring in fewer than five dissertations). The total vocabulary contains 20,961 word types.

These words serve as the input to our models from which we derive scores of departmental similarity based on the text of each department's dissertations. We initially constructed two models each representing a common approach to textual similarity in the literature. The first metric is based on *word similarity* measuring the overlap of words. The second is *topic similarity* in which we measure similarity in a lower dimensional space of inferred topics.

Word Similarity Based on tf.idf

We compute the word similarity of departments based on the cosine similarity of tf.idf vectors representing each department, a standard approach used in information retrieval [141]. Each component i of the vector for a department v_D is computed by multiplying the number of times term i occurs in the dissertations from that department (tf) by the inverse document frequency (idf), computed as $\log(N/df_i)$ where N is the number of dissertations in the dataset and df_i is the number of dissertations that contain the term i. We define the word similarity of two departments D_1 and

D_2 as cosine (or the angle) between their corresponding tf.idf vectors v.

$$\cos(v_{D_1}, v_{D_2}) = \frac{v_{D_1} \cdot v_{D_2}}{\|v_{D_1}\|\|v_{D_2}\|}$$

Topic Similarity Based on Latent Topics

While tf.idf is effective for scoring similarity for documents that use exactly identical words, it cannot assign a high score to the shared use of related terms (e.g., "heat" and "thermodynamics") because each term is represented as its own dimension in the vector space. To address term sparsity issues, we apply latent Dirichlet allocation (LDA) [11] to infer latent topics in the corpus, and represent documents as a lower-dimensional distribution over the topics.

We compute the topic similarity of two departments D_1 and D_2 as the cosine similarity of their expected distribution over the topics θ_d learned by LDA. This expectation is the average distribution over latent topics for dissertations in that department, and is computed as the following.

$$\mathbb{E}\left[\theta_D\right] = \frac{1}{|D|} \sum_{d \in D} \theta_d$$

Accounting for Time

In both of the models above, we quantify the similarity of departments over time by computing a time-aware signature vector. To compute the vector for a department D within a year y, we sum across all dissertations in D either in the year y or in the preceding two years $y-1$ and $y-2$, weighting the current year by $\frac{1}{2}$, the preceding year by $\frac{1}{3}$ and the remaining year by $\frac{1}{6}$. The extra years are included in the signature to reduce sparsity and account for the influence of a student's work prior to completing a dissertation.

3.2.3 Landscape, Department, and Thesis Views

The first visualization I created was the *Landscape View* (Figures 3.1, 3.2, 3.3, and 3.4). This view was intended for revealing global patterns of change in departmental

topical compositions. I encode academic departments as circles with areas proportional to the number of dissertations filed in a given year. Distance between circles encodes one of the similarity measures, subject to principal component analysis (PCA) projection. I ensured visual stability by limiting the amount of movement between adjacent years under the projection. Time is controlled by a slider bar that enables analysts to view an animation of temporal changes or immediately access a specific year.

Consider the landscape views. In Figure 3.2, word similarity suggests a relatively uniform landscape. In Figure 3.3, however, topic similarity predicts a tight overlap of research topics in Medicine (purple) and Humanities (orange) with a relative diverse set of topics in Engineering (blue) and Sciences (green). Which measure best characterizes the university's research output? Without an interactive validation mechanism or an external ground truth, we were left with no way to choose between the similarity measures or to trust that the projection faithfully represents the similarity scores derived from each model. The social scientists were unable to confirm whether the observations — in any of the views — correspond to interdisciplinary work or to gain insight about the nature of potential collaborations.

In response to these issues of trust, I designed the *Department View* as shown in Figure 3.5 to focus on a single department at a time. This view explicitly displays the distance from a focused department to every other department (i.e., a single row in the similarity matrix) without distortion. Similarities are encoded as radial distances from the focused department at the center of the display. The remaining departments are arranged around the circle, first grouped by school, and then alphabetically within each school. A circular representation was chosen to avoid a false impression of rank-ordering among the departments and to fit into a single display without scrolling. By restricting the amount of data visible at a single time, the department view avoids projection artifacts.

This view enabled my collaborators to inspect expected patterns, such as connections between economics and business, and discover surprises. For example, contrary to their expectations, they found that statistics and computer science were not becoming consistently more similar; instead, they were the most similar in year 1999.

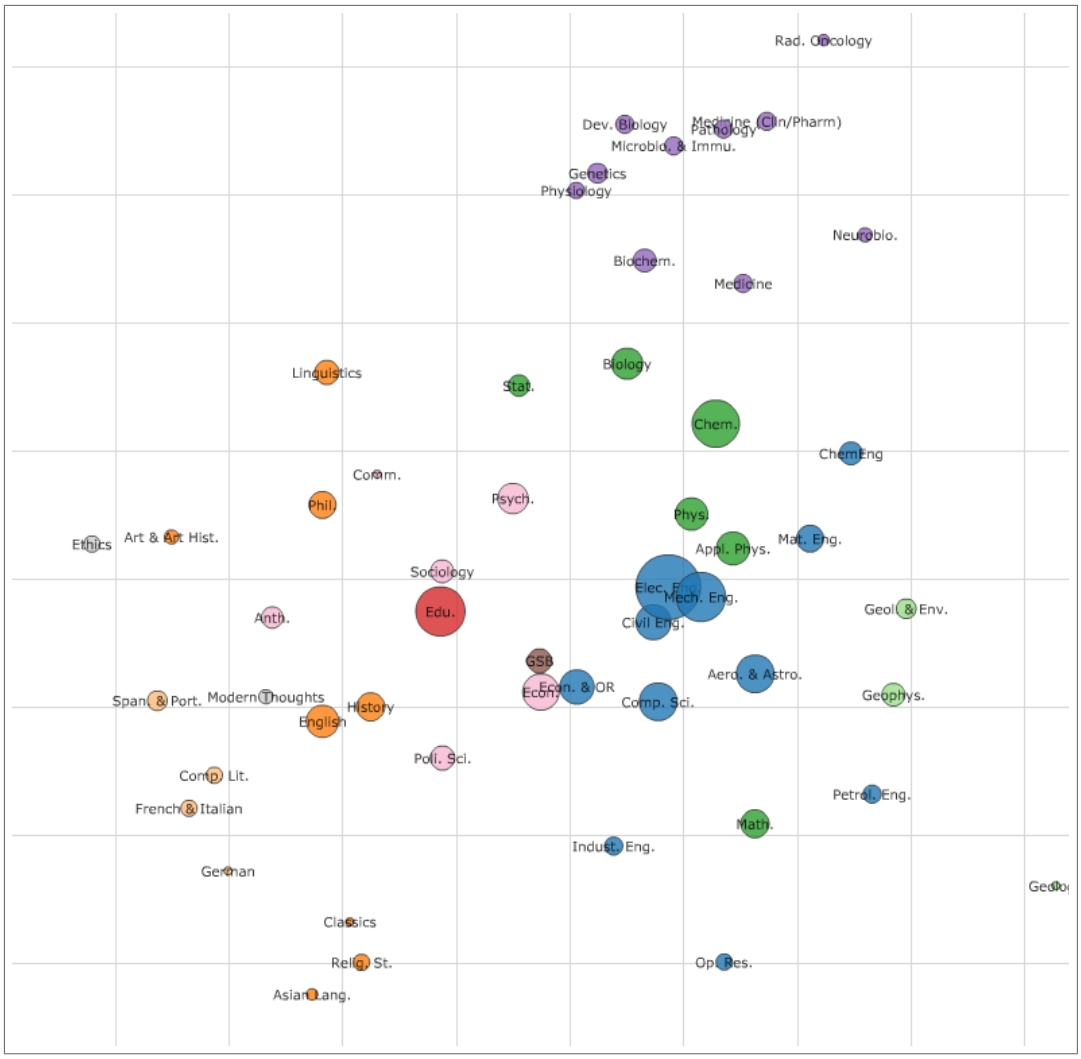

Figure 3.2: Departmental relationships based on tf.idf word similarity suggests a relatively uniform landscape. When compared with Figures 3.3 and 3.4, all three overviews seem plausible, but each makes different predictions and offers little guidance in choosing a model.

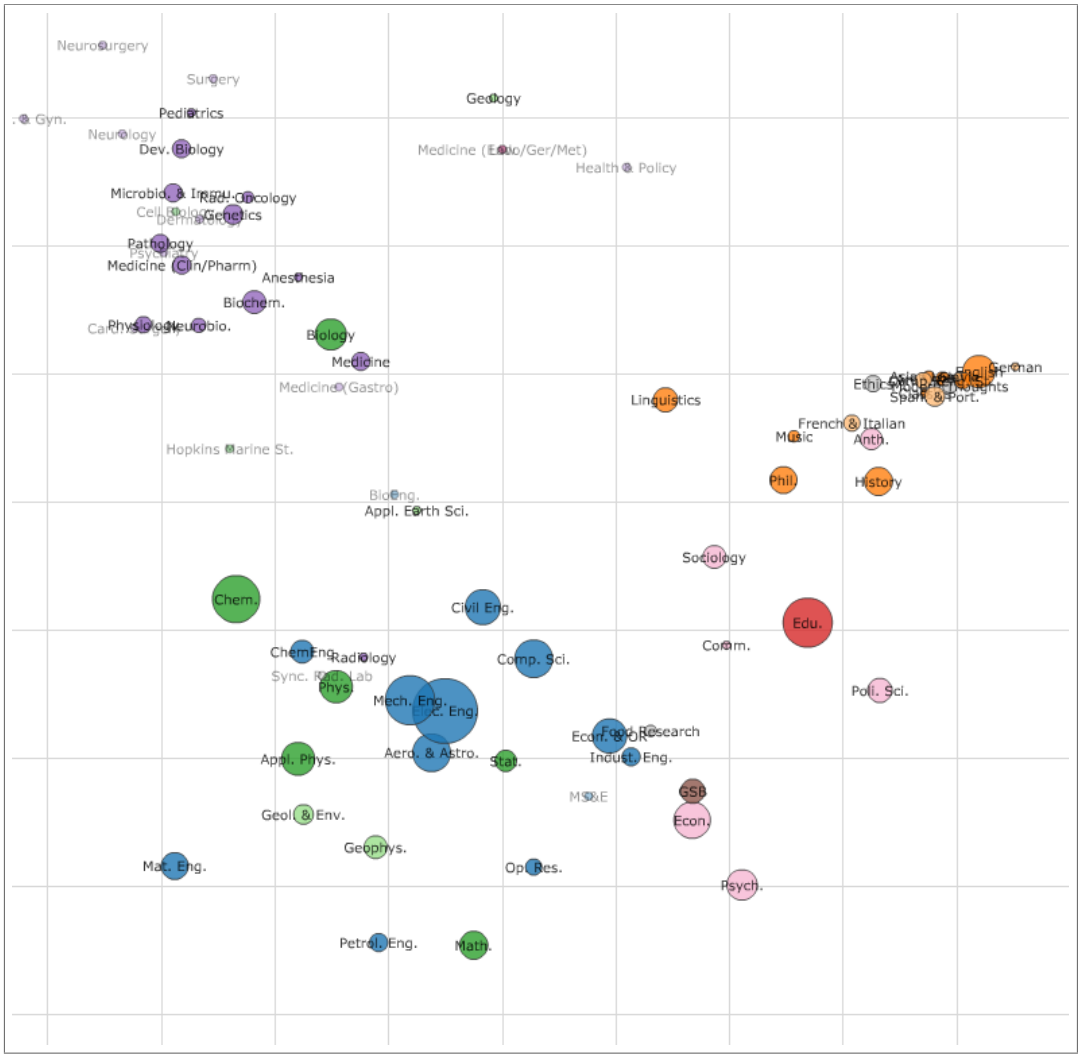

Figure 3.3: LDA topic similarity predicts a tight overlap of research topics in Medicine (purple) and Humanities (orange) with a relative diverse set of topics in Engineering (blue) and Sciences (green). Compare with Figures 3.2 and 3.4.

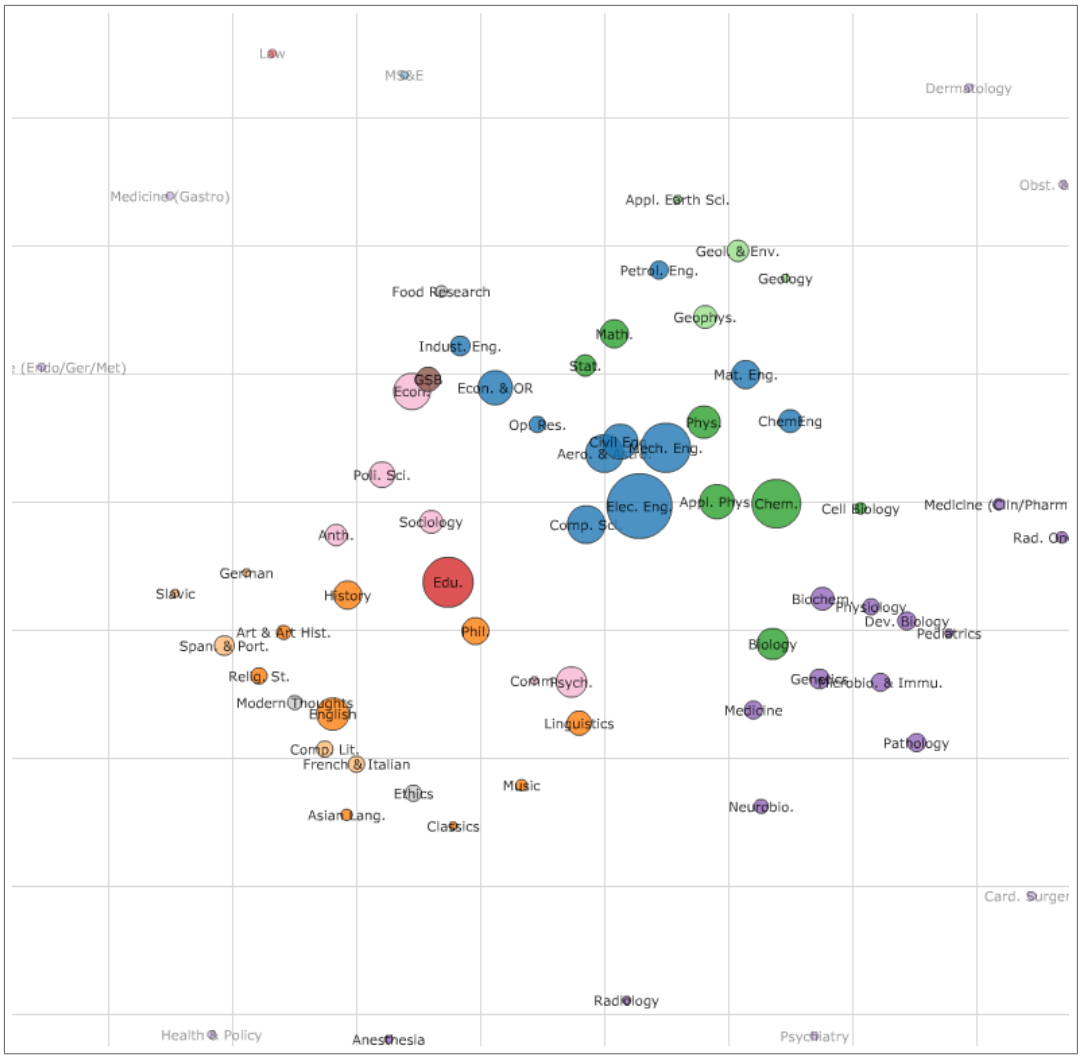

Figure 3.4: Similarities predicted by department mixture proportions best matches expert judgment. Using a supervised machine learning approach, we estimate the similarity of two departments by measuring how often dissertations from one department "borrow" words from another. Compare with Figures 3.2 and 3.3.

This surprise suggests the need for an even deeper level of verification — examining individual dissertations that contribute to the high (or low) similarity scores of two departments in a given year.

The department view also reveals peculiarities in the underlying models. Figure 3.5 centers on English and corresponds to the landscape view in Figure 3.3. This figure immediately suggests a fundamental issue in the topic similarity score derived from latent topic models: how to appropriately select the number of latent topics N used to model the corpus. For the model in Figure 3.5, we chose the topic count that maximizes the perplexity on the held-out data — the technique most commonly used to select the number of topics. However, the visualization demonstrates that the model clearly has too few topics to adequately describe variation within the humanities. A larger number of topics may mitigate this effect, but we lack data-driven metrics for making a principled selection.

As a result, I added the *Thesis View* as shown in Figure 3.6 to support validation and exploration of observed similarity scores. The thesis view is presented in response to a click on the centered department in the department view. Every dissertation from the focused department, as well as the most similar dissertations from other departments, are added to the visualization within a concentric circle between the focus and the other departments. The angular position of a thesis aligns with the most similar department excluding the focus; the radial position is a function of the ratio of the dissertation's similarity to those two departments. This encoding provides a simple means to note theses that might connect two departments.

Upon mouse over, the text of the thesis abstract is shown, enabling analysts to read the source text and judge whether the two departments are sensible anchors for the dissertation. This view allows users to explore the relationships between departments at a fine-grained level, providing texture and context to the observed department-level similarities.

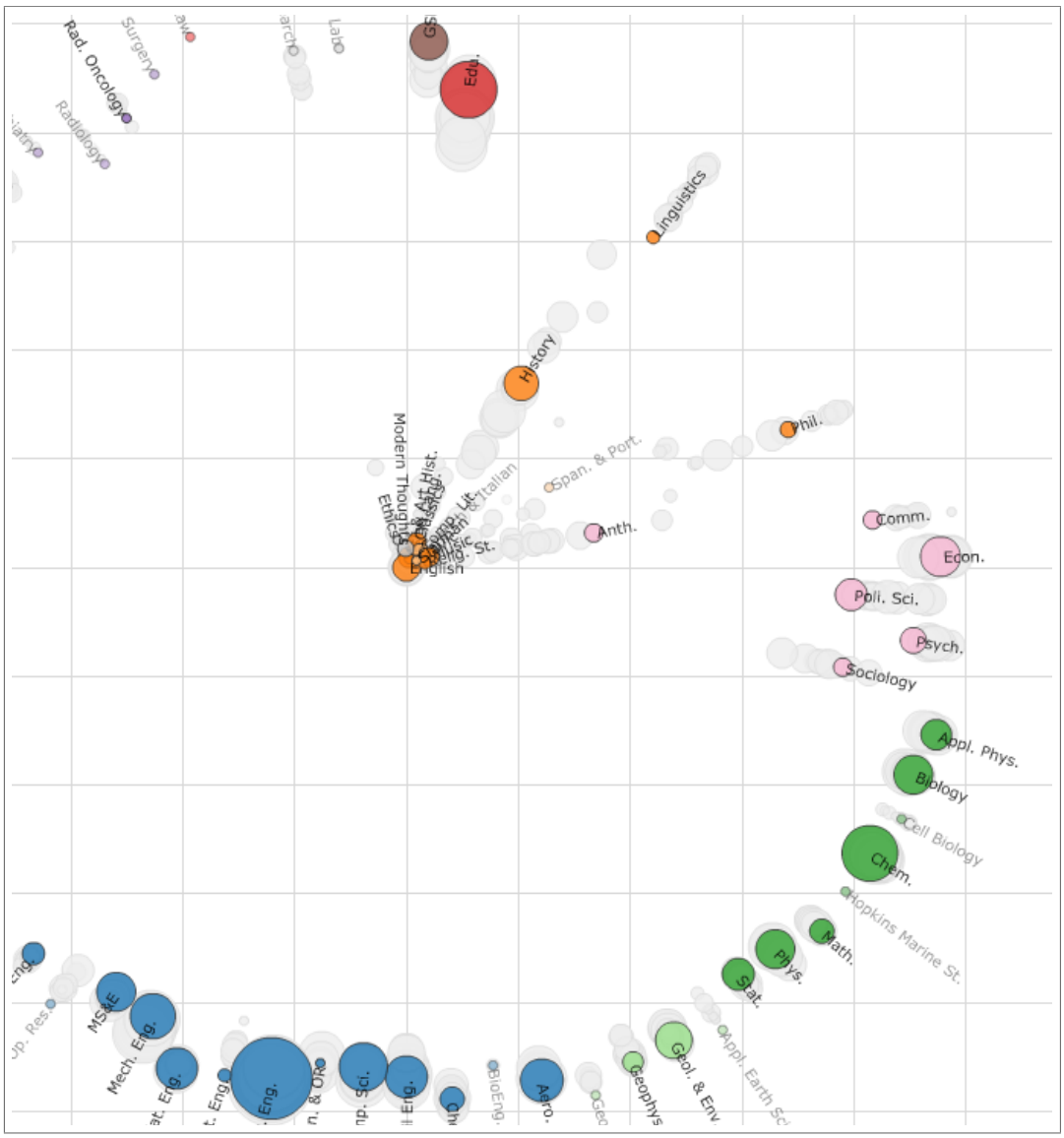

Figure 3.5: Department View using LDA topic similarity, focused on the English department. While the overview (Figure 3.3) seems plausible, we now see that the humanities have been clustered far too aggressively.

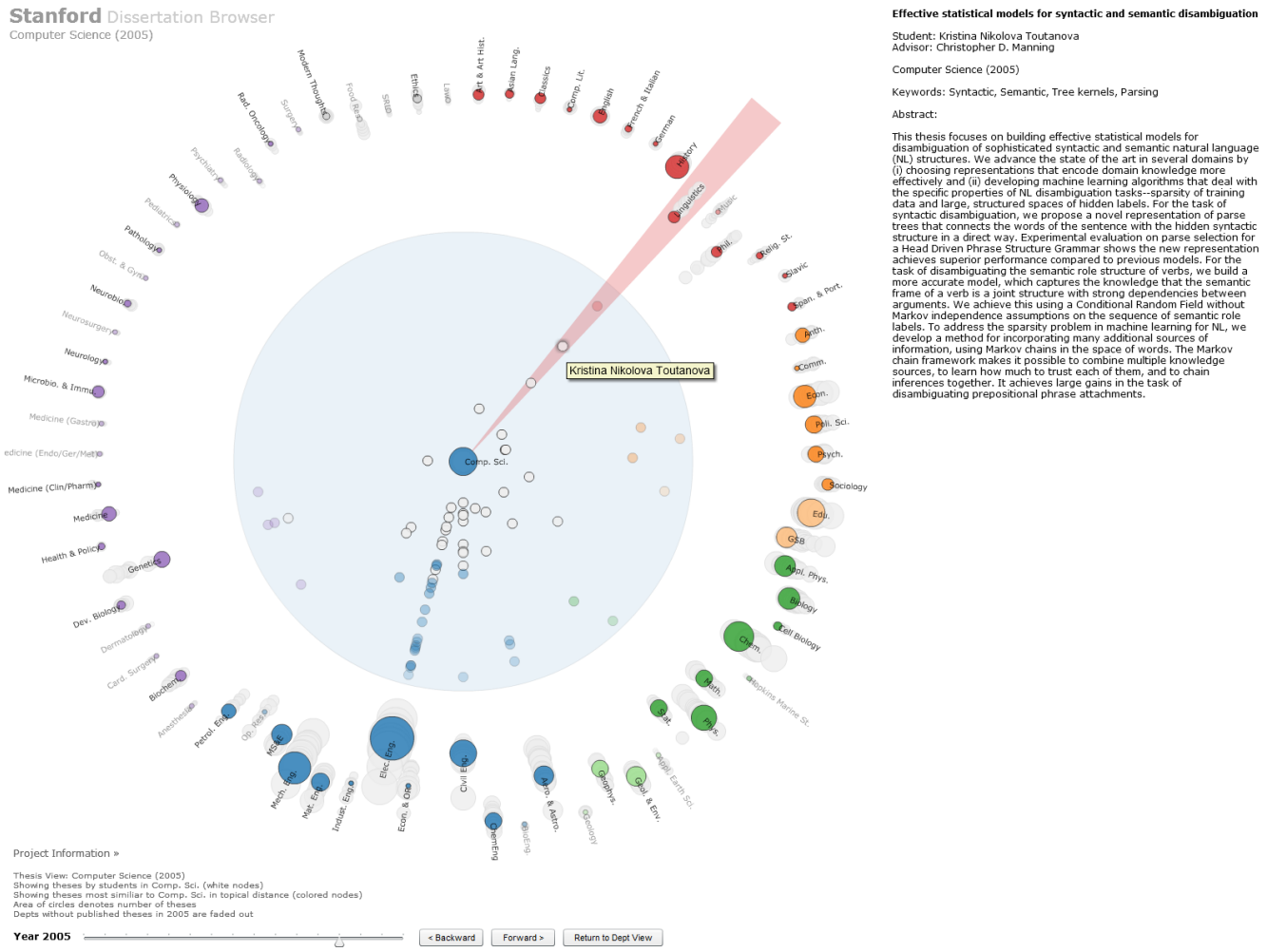

Figure 3.6: Thesis View shows individual dissertations as small circles placed between the focus department and the next most similar department. Reading the original text enables experts to evaluate observed dept-dept similarities, and confirm the placement of three computational linguistics Ph.D.s that graduated in 2005.

3.2.4 Evaluating the Models

To assess our modeling options, we conducted an expert review. We invited academic domain experts including professors and graduate students to use the interface and recorded their responses. We found that the visualizations benefited from being model agnostic. They display departmental similarity but otherwise are not constrained by other modeling assumptions. A consistent visual representation can thus be used to compare the results of different modeling approaches.

Using the landscape view, participants could not fully justify their observations. Many potentially interesting patterns turned out to be projection artifacts, ultimately leading us to remove this view from the tool. Using the department view, participants were adept at noting similarities that violated their assumptions. Both word and topic similarity led to many such instances. Rather than identifying a preferred model, we became increasingly skeptical of both approaches.

The successes and mistakes of each similarity model were revealed by the thesis view through the (mis)placement of individual dissertations with respect to the other departments. Participants were able to discover systematic errors made by topic similarity. For instance, several biology dissertations were spuriously linked to computer science and vice versa because of the existence of a computational biology topic that connected the dissertations, even though many dissertations made use of only the biology or computer science words in the computational biology topic. The tf.idf measure used for word similarity, on the other hand, often assigned documents very high similarity to departments that happened to heavily use a common rare word.

We also used our own domain knowledge to examine the relationships between dissertations and departments. The placement of three computational linguistics Ph.D.s that graduated in 2005 provides an illustrative example (Figure 3.6). We expected these dissertations to fall on the line between computer science and linguistics. In the latent topic model's similarity function, two of them did, but several unrelated dissertations were deemed substantially more similar to linguistics than the computational linguistics dissertations. We discovered this was due to a shared latent topic that covered both linguistics and information retrieval. While the tf.idf model succeeds in placing these three dissertations between computer science and linguistics, it failed

to accurately describe the relationship between the two departments. Year 2000 with only one dissertation is the year of maximum similarity even though the dissertation is not computational in nature.

3.2.5 Revising the Model: Department Mixture Proportions

The high frequency of "mismatch" between experts' mental models and our similarity scores led us to revisit our modeling assumptions. First, we wished to avoid arbitrary parameters such as the number of latent topics (N) and realized that we might better exploit the available metadata. Second, we had implicitly assumed that our similarity measure should be symmetric, as required by the mathematical definition of a metric. However, this need not be true of analysts' views of departmental similarity. In response, we formulated a novel similarity score that we call the *department mixture proportion*. This measure uses a supervised machine learning approach to directly represent the contents of each department, our primary unit of analysis. We estimate the similarity of two departments by measuring how often dissertations from one department "borrow" words from another.

To compute the department mixture proportion, my collaborating machine learning researcher utilizes the machineries of Labeled LDA [1] [128] which models each document as a latent mixture of known labels. In a two-step process, we first learn latent topics using the departments associated with each dissertation as labels. In a second inference step where labels are subsequently ignored, we infer department mixtures for each thesis.

To provide context, I briefly summarize the machineries of the unsupervised latent Dirichlet allocation algorithm.

We train a Labeled LDA model using the departmental affiliations of dissertation committee members as labels. Thus the departments themselves are the "topics". Each dissertation may have one or more labels. During training, we learn both the per-topic term distributions (β_k) and initial label-based topic mixtures (θ'_d). In Labeled LDA, topical term distributions are allowed to take on any word, as in normal

[1] The Stanford Topic Modeling Toolbox, which includes a Labeled LDA implementation, is available online at `http://nlp.stanford.edu/software/tmt/`

LDA training. However, per-document topic mixtures are restricted to only labels associated with the document. For example, the topic mixture for a thesis labeled "Biology" and "Chemistry" is zero for all topics except the two labeled departments.

Using the learned topical term distributions (β_k), we next ignore all labels and perform standard LDA inference on each dissertation (as if we were seeing it for the first time). This results in a new topic mixture (θ_d) in which the dissertation can "borrow" words from *any* department, not just the ones it was initially labeled with. We average the distributions for all dissertations in a given department to construct the department mixture proportion. The values of this averaged distribution are the desired similarity scores.

In short, we first determine the term distributions of each department and then use these distributions to answer a simple hypothetical: if we let each dissertation borrow words from *any* department, what mixture of departments would it use? The resulting mixture proportion tells us the fraction of words in each dissertation that can be best attributed to each department. The similarity of a department D_1 to D_2 is now simply the value at index D_2 in θ_{D_1}. Unlike the previous measures, this score need not be symmetric. For instance, Music may borrow more words from Computer Science than Computer Science does from Music, a pattern that we observe in several years where computational music Ph.D. dissertations are filed. This new similarity score ameliorates many of the "mismatches" identified by our earlier expert review.

3.2.6 System Deployment and Observations of Use

I first deployed the Dissertation Browser[2] outside of my research team in March 2010, as part of a presentation to the University President's Office. For convenience, I launched the tool on the web where it remained available after the presentation. My collaborators found the primary value of the tool to be in validation and communication. They noted the start of a large-scale Biophysics project connecting Biology and Physics in 2006. Several finer stories were discovered that exhibit interdisciplinary collaboration and knowledge transfer. In one case, the visualization demonstrated a

[2]The Stanford Dissertation Browser is available online at `http://vis.stanford.edu/dissertations/`

strong connection between two departments driven by a small number of individuals centered around the Magnetic Resonance Systems Research Lab. This lab graduated a series of Electrical Engineering Ph.D. students in the 1990's who worked on EE-aspects of various MRI techniques. Around the same time, a hire in Radiology held a courtesy appointment in Electrical Engineering. For the next decade, the influence of these groups strongly connected the two departments until both eventually moved onto other research areas.

As we made no effort to publicize the tool, we were taken by surprise when the system gained public attention from users on the web (e.g., in hundreds of Twitter comments) beginning in December 2010. The majority of tweets expressed interest or enjoyment in the use of the tool ("geeky and cool", "i could spend hours on this site"). Several pointed to specific patterns ("In 2003 Edu was closer to PoliSci than English", "Watch Psychology and Education PhD theses doing the hokey-pokey over time"). Later, over a dozen science and tech blogs (including Hacker News, Discover Magazine and Flowing Data) posted articles about the tool. We observed commenters interpreting specific patterns of interest: "I was not surprised to see the link between Computer Science and Philosophy. Heartened by a slight connection between dissertations in Computer Science and Genetics." and "Aha, so there are terms that are common between civil engineering and biology but not between civil engineering and religion or art history." We also observed issues of trust: "[browser] thinks neurobiology is closer to electrical engineering than to biology. It is easy to see why that might be so based on key vocabulary terms (voltage, potential, conductance, ion), but $\cdots$." From these and similar comments, we note that the ability to transition between levels of model abstractions enabled users to interrogate the model and assess unexpected correlations.

In summary, while text visualization research has traditionally focused on improving the effectiveness of a visualization, I find that the iterative design process needs to be extended to consider how the underlying model itself affects or can be adapted towards an analysis goal. I demonstrate how a novel "word-borrowing" modeling approach arose through a design process that considered task analysis, visualization

design, expert feedback, and modeling choices in a unified fashion. Moreover, machine learning research has normally been content with formal measures of model quality, with less emphasis on user- and task-centric evaluations. However, constructing a high-quality model suitable for domain-specific analysis tasks necessitates verification by experts in the domain. Aligning the units of analysis can improve the interpretability of the visualization and the underlying model — and aid verification as well as any possibly modifications to the model in response to user feedback. I observe that analysts and other users of the Stanford Dissertation Browser gained the most valuable insight — and trust in the system — by progressively inspecting the visualization and the model at multiple levels of details.

3.3 Temporal and Large-Scale Academic Discourse

After the Dissertation Browser, I continued my collaboration with researchers from the Stanford MIMIR Project to examine temporal trends in academic discourse. We also expanded our topic modeling efforts to include over one million Ph.D. dissertations in order to investigate large-scale language transfer among academic disciplines.

3.3.1 Topic Flow Visualization

My social scientist colleagues were interested in the history of academic disciplines. For example, identifying the emergence and convergence of research topics might provide insights on the factors that can give rise to a new field. Tracking the gradual decline of a research topic might answer the following two questions: Do academic disciplines die and disappear? Or, do they become mature, and in becoming a part of the fundamental language of academic research, cease to be viewed as a topic?

For this investigation, our modeling goals were to quantify the notion of *ideas* and to capture how ideas *influence* one another. To this end, we turned to topical analysis of citation networks. My collaborating machine learning researcher applied the TopicFlow model [111, 112] to the ACL Anthology Network [122, 123] consisting of 15,160 conference papers and journal articles representing the research output of

computational linguistics over the past 45 years. The dataset also contained 33,594 citations internally among these papers. Treating each publication as an idea, we sought to capture the notation of influence through citations in order to identify and track the development of *research topics* (i.e., aggregation of ideas).

TopicFlow combines network analysis with topic modeling. The algorithm analyzes the hyperlink structure of the citation graph and computes influences in a manner similar to PageRank [118]. At the same time, TopicFlow learns the topical content of each document using latent Dirichlet allocation [11]. Unlike topic-sensitive PageRank [63] where a set of topics need to be specified in advance, TopicFlow learns the topics in conjunction with flow computation and generates topic-specific flows along the citations.

The social scientists asked experts in different areas of computational linguistics to verify the model output. To facilitate exploration, validation, and communication, I created a visualization based on the citation graph (Figure 3.7). Vertical displacement in the graph represented time. Topical flows and topical compositions (i.e., the total topical flow through a node) were superimposed on the edges and nodes respectively. The interface provided the users with options to search, filter, and highlight a subset of the graph.

Misalignment between the experts', model builders', and the model's views of the data quickly surfaced. While the model's basic unit of analysis was the amount of topical flow along the citations, the experts reported that assessing the significance of individual citations was extremely difficult. During validation, the experts looked for lineages of papers or groups of authors that advanced of a topic or body of knowledge. In other words, papers and authors — not individual citations — formed the salient units by which experts judged importance and relevance. The experts described the display of flows as a hindrance as they tried to ignore the line widths imposed by the visualization. On the other hand, the model builders preferred the display of topical flows which matched the actual computational mechanism of the model. Examining the flows — without any intervening abstractions — provided them with detailed information about the performance of their model.

In response, I introduced customized views for the experts and the model builders.

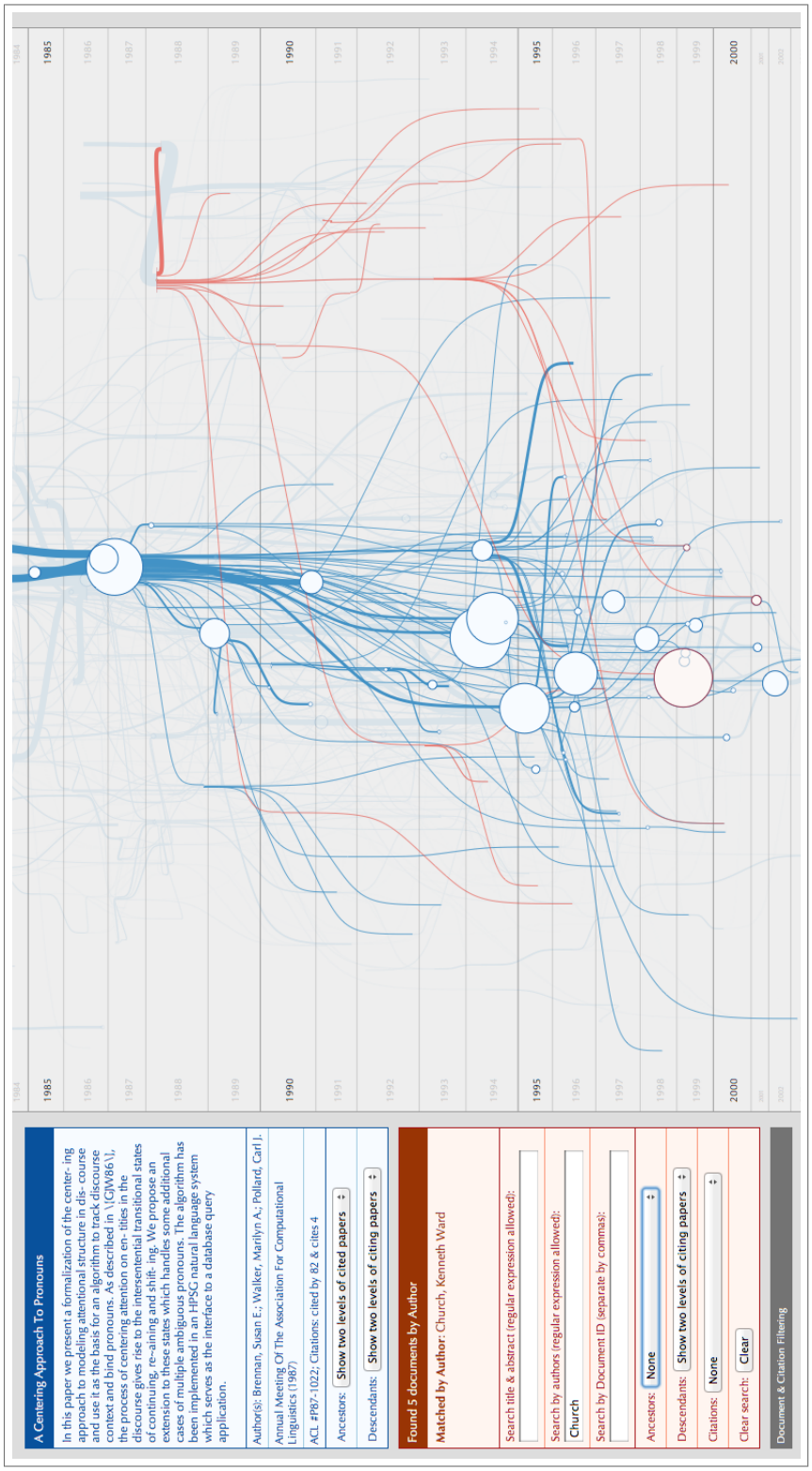

Figure 3.7: Topic Flow Visualization for exploring 45 years of computational linguistics (ACL Anthology Network) data by topical influences, based on TopicFlow Model [112] analysis.

I applied different visual encodings to the nodes and edges, such as reducing the prominence of the lines for the experts and introducing search options for expressing more complex citation or author relationships. The personalized views featured visual elements that more truthfully matched each user's respective mental units of analysis (papers and authors vs. topical flow) which in turn better supported their respective tasks (assessing the importance of papers vs. adjusting the modeling parameters to produce topical flows that best capture expert opinion). By maintaining a common graph layout among all the views, the visualization enabled my collaborators to communicate their findings despite the different visual encodings.

My final visualization revealed previously undocumented issues with the TopicFlow algorithm. During validation, the experts were troubled by the fact that a number of relatively unimportant papers received a large amount of topical assignment in multiple topics. Examining the flows, the model builders reported that they were due to flows accumulating along cyclical references in the citation graph: a condition that the model had assumed should not exist but nonetheless occurred in real-world data.

This case study further highlights the need to support interpretation and establish trust when designing data analysis tools. Statistical models are typically designed to approximate meaningful concepts in a domain of analysis. However, model outputs can deviate from the intended concepts due to a variety of reasons, such as inappropriate modeling assumptions. Here, we observe an additional possible source of modeling errors: dirty data. My collaborating social scientists are rightfully concerned about the validity of the model outputs. By modifying the visualization and improving its interpretability, my tool allows experts to more efficiently interpret the model predictions, and in doing so, spot errors in the data.

3.3.2 Visualizing Language Transfer in Academia

In a separate project, we extended our analysis on inter-disciplinary collaborations to investigate large-scale language transfer across academic disciplines. While traditional survey methods such as literature reviews, expert interviews, and questionnaires can provide detailed stories about the development in specific subjects, such

methods do not scale for our analysis goal, which was to analyze research output at a national level. Network-based analysis (e.g., citations and co-authorship) capture only formal references and might exclude influences due to informal conversations and often-uncited distant readings of work in other fields. We chose to analyze Ph.D. dissertations as they have a greater coverage of research fields than do journal publication databases. In the end, we examined abstracts from over 1.05 million Ph.D. dissertations published between 1980 and 2010 from 157 U.S. universities classified as research-intensive by the Carnegie Foundation [25].

Based on previous experience, our machine learning collaborators chose a topic model that allowed analysts to explicitly express domain expertise and prevent unnecessary modeling abstractions and complexities. Partially labeled Dirichlet allocation [129] is a semi-supervised learning algorithm based on a three-level soft clustering of words. Users are allow to assign *labels* to documents that exemplify a predetermined set of topical concepts. We chose 268 ProQuest subject codes as labels, which form the basic modeling units. Dissertations tagged with the corresponding subject codes were used as exemplary documents in the training process. We grouped the subject codes into 69 subject codes based on National Research Council (NRC) classification. These areas were then further grouped into six broad area designations chosen by our team: engineering; physical and mathematical sciences; biological sciences; earth and agricultural sciences; social sciences; and humanities. These areas and broad areas became the common units of analysis.

I created two views of the results. First, a circle view shows all areas in a ring and displays a link between areas exhibiting strong topical overlap (Figure 3.8). Second, a matrix view shows detailed language transfer (Figure 3.9). The area of a circle at row i and column j represents how likely it is that dissertations published in area j uses the language of area i. Colors denote the broad areas.

My visual analysis tool enabled the social scientists, model builders, and experts to iterate through many versions of the models based various parameter settings, identify a high-quality model, and verify that the results were robust across a wide range of assumptions and parameters. As described earlier in Section 3.2.4, a model

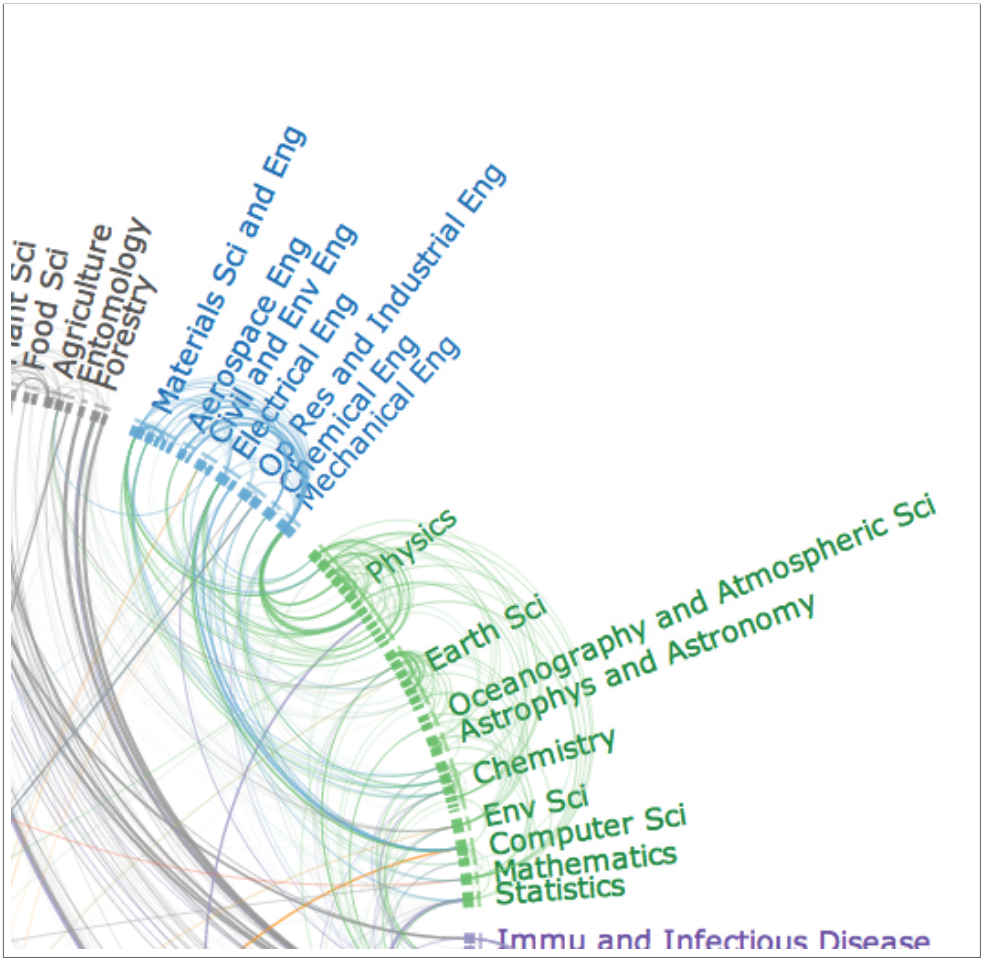

Figure 3.8: Circle view showing topical overlap between research areas. Based on partially labeled Dirichlet allocation or PLDA [129] applied to one million Ph.D. dissertations published in 157 universities in the United States.

Figure 3.9: Matrix view showing detailed topical assignments. The area of a circle at row i and column j represents how much dissertations in area j draw on the language of area i. Based on partially labeled Dirichlet allocation or PLDA [129] analysis on over one million Ph.D. dissertation abstracts published in the United States. Due to the size of the data and the visualization, labels are replicated within the matrix on mouseover to aid look up. Circles on opposing sides of the diagonal represent language exchange between two areas in opposite directions (i.e. word borrowing from i to j vs. from j to i). The two corresponding circles are always highlighted in tendon to aid the comparison on the directionality of language exchange.

agnostic representation is effective in supporting model comparison. The final visualization and model led to various observations [125], including the identification of methodological fields that export broadly, emergence of topical fields that borrow heavily and expand, and old topical fields that grow insular and retract.

In summary, across these projects, we find that successful model-driven visualizations depend on appropriate alignment of the model, visualization, analysis task, and user expertise. Exposing model abstractions can support model modification. Contextual information such as labels can aid model verification. Variables amenable to modeling, however, may differ from the ideal dimensions for analysis or presentation. In these cases, visualization can be critical in supporting collaboration — by adapting to each stakeholder's analysis task and by creating a shared representation to enable effective communication.

3.4 Design Guidelines

Across the three previous projects (Sections 3.2 and 3.3) and based on the earlier literature review on text visualizations (Section 2.3), I find that successful model-driven visualizations depend on appropriate alignment of the model, visualization, analysis task, and user expertise. Exposing model abstractions can support model modification. Drilling down/zooming out and providing context can aid model verification. Variables amenable to modeling, however, may differ from the ideal dimensions for analysis or presentation. In these cases, visualization can be critical in supporting collaboration — by adapting to each stakeholder's analysis task and by creating a shared representation to enable effective communication.

To facilitate interpretation and trust in model-driven visualizations, I distill the following process-oriented recommendations for model and visualization design:

- **Align** the analysis tasks, visual encodings, and modeling decisions along appropriate *units of analysis*.
- **Verify** the modeling decisions: ensure that model output accurately conveys concepts relevant to analysis.

- Provide interactions to **modify** a model during analysis.
- **Progressively disclose** data to support reasoning at multiple levels of model abstraction.

3.4.1 Model Alignment

I use the term *alignment* to describe the correspondences among modeling decisions, visual encoding decisions, and an analyst's tasks, expectations, and background knowledge. I consider a visual analysis system to be well-aligned when the details surfaced in the visualization are responsive to analyst's tasks, while minimizing extraneous information that might confuse or hamper interpretation. Alignment does not result from interface design alone; both the visualization and model may require iterative design.

Identify Units of Analysis

Alignment requires a sufficient understanding of users, their tasks, and the context of use. Such *domain characterization* [109] relies on methods familiar to HCI researchers (e.g., interviews, contextual inquiry, participant-observation). However, these techniques may be foreign to model designers in fields such as statistics or machine learning. To facilitate communication among stakeholders with varying backgrounds, I found it useful to frame insights in terms of *units of analysis*: entities, relationships, and concepts about which the analysts reason. These units serve as a resource for evaluating models and their fitness to the analysis task.

With the Dissertation Browser, I engaged in participatory design meetings with my collaborators to determine the units of analysis. This process led us to realize that changes in inter-department similarity could provide answers to the social scientists' research questions. In turn, I was led to depict similarity data in the visualization and avoid the potentially confusing route of trying to convey topical composition. In later iterations I further aligned my model with this unit of analysis: I reduced the number of abstractions by computing similarity directly as the department mixture

proportion. This eliminated the need to set model parameters such as the number of topics and freed analysts from unnecessarily assessing and classifying latent topics.

Assess Reliability vs. Relevance Tradeoffs

Selecting the appropriate units of analysis often involves a balance between how reliably a concept can be identified, and how relevant the concept is to the analysis task. The final units of analysis reflected in a visual analysis tool may result from a compromise: the units should correspond to the analysts' questions but must also be practical to model.

In the Dissertation Browser, I quantify "units of research" as academic departments. While my social science collaborators would ideally like to assess research at a finer granularity (e.g., trends in microbiology or evolutionary systems), I lacked reliable means to quantify such units of research. LDA models have the potential to discover unnamed research activities, but in our case collapsed all of the humanities into a single topic. Similarly, while investigating historical trends using LDA models, Hall et al. [62] found that only 36 out of 100 automatically inferred "topics" were judged relevant by experts in the field. Named organizations such as departments can be identified reliably, and correspond to concepts that the analysts can comprehend and verify during analysis. More generally, I recommend leveraging available metadata to provide reliable and relevant units of analysis.

Enumerate Model Assumptions

To assess alignment, it is valuable to explicitly enumerate the assumptions implicit in a modeling approach. Common assumptions in quantitative statistics are that data values are independently and identically distributed according to a known probability distribution (e.g., Gaussian, Poisson, etc.). Within text processing, many models are predicated on a bag-of-words assumption that ignores word ordering and relations. Understanding such assumptions is important for determining if a model is appropriate for the given units of analysis. Enumerating assumptions also provides a resource for design, suggesting potential starting points for alternative models.

While designing the Dissertation Browser, I assumed that similarity must be based on a proper metric, and hence symmetric. Once I identified this assumption, it freed us to consider the possibility of asymmetric similarity scores, ultimately leading to a "word borrowing" model based on the department mixture proportion. In Review Spotlight [177], the mismatch between the bag-of-words model and sentiment perception was resolved by making adjective-noun pairs the units of analysis, yielding improved performance.

3.4.2 Model Verification

Once candidate models have been identified, I need to assess how well they fit an analyst's goals. An analytical abstraction based on identified units of analysis can often be realized by different modeling approaches. Verification may require collaboration among designers and domain experts to assess model quality and validate model output.

Assess Model Fit

In domains with objective accuracies, one can take a quantitative approach to verification: common evaluation measures include precision (e.g., comparing model output to known ground truth data) or internal goodness-of-fit statistics (e.g., information criteria such as AIC and BIC). However, one should ensure that such metrics correlate with analysis goals. Domains such as text interpretation may be subjective in nature and so difficult to quantify. For LDA topic models, quality is typically measured in perplexity, which describes the "distinctiveness" of the learned topics. While perplexity is a sensible measure of encoding quality in an information-theoretic sense, in our case it did not correspond to our task: identifying concepts representing coherent "research topics."

Conduct End-User Evaluations

HCI evaluation methods can enable verification. For example, task-based user studies or real-world deployments may be used to assess how well a system aids analysis

tasks. Walkthroughs with representative users can help designers gauge analysts' familiarity with a presented analytical abstraction. A potential trade-off is that if analysts don't fully understand the model (e.g., higher gulf of evaluation) but gain more useful and verifiable insights, a less familiar model may be preferred. In my case, I found that expert review was a relatively lightweight means to assess model quality by cataloging instances in which users believed the model to be in error. These "mismatches" became points of comparison across modeling options. An interesting challenge for future work is to better correlate the results of user-centered evaluation with less costly model quality metrics: Can we identify or invent better metrics that reliably accelerate verification?

Enable Comparison via Model-Agnostic Views

Another method for verification is triangulation: comparing the output of multiple models or parameter settings and gauging agreement. To enable cross-model comparison in a model-driven visualization, the visualized units of analysis should be stable across modeling choices. I use the term *model-agnostic views* to describe visualizations that use a single analytical abstraction to compare the output of various underlying modeling options. To be clear, such views rely on a stable abstraction; what they are "agnostic" to is the inferential machinery of the models. For example, the Dissertation Browser uses inter-department similarity as the shared unit of analysis, enabling comparisons with any model that can generate suitable similarity scores. Interactive comparison of parameter settings and modeling options can be invaluable to model designers when assessing choices. Providing similar facilities to end users is also helpful, but might best be treated as a "last resort" when an accurate, well-aligned model can't be found.

3.4.3 Model Modification

Even with careful attention to alignment and verification, a model's output may be incorrect or incomplete. Whether due to limited training data or inaccurate yet pragmatic modeling assumptions, analysts often require mechanisms to modify a model

abstraction over time. The approaches listed below constitute ways to interactively improve model alignment.

Modify Model Parameters

A simple form of model modification is to adjust free parameters. Examples include setting the number of topics in an LDA model or adjusting threshold values for data inclusion (e.g., weights on edges in a social network). I have found that this ability is critical for early stage model exploration. While ideally this would not be necessary in a final analysis tool, in practice one rarely finds a "perfect" model. Consequently it is important for analysts to be able to assess various parameterizations. One challenge is to support real-time interactivity, as changes of model parameters may require expensive re-fitting or other operations. For such cases, visual analysis tools might provide facilities for scheduling offline, batch computation across a range of parameter values.

Add (Labeled) Training Data

Another approach to model modification is to introduce additional training data. For example, an analyst might add new text documents labeled as positive or negative examples of a category. In the context of the Dissertation Browser, new inference procedures might incorporate expert annotations into the model fitting process. To avoid costly re-fitting, designers might leverage techniques for online, interactive machine learning [2, 54]. An important research challenge is to design reflective systems that elicit the most useful training data from users, perhaps using active learning methods [39].

Adjust The Model Structure

Analysts familiar with a modeling method may wish to directly edit the model structure. An analyst might add new latent variables or conditional dependencies within a Bayesian network, or add a new factor to a generalized linear model. In this case,

the model itself becomes a unit of analysis, requiring that users possess sufficient modeling expertise.

Allow Manual Override

An alternative approach is to bypass the modeling machinery entirely to override model output. For example, to correct modeling mistakes or impose relations outside the scope of the model or source data. Analysts may wish to delete or modify inferred LDA topics. Hall et al. [62] removed 64 topics and inserted 10 hand-crafted topics in order to complete their investigation; Talley et al. [152] removed poor topics and flagged questionable topics in their visualization. Similar to model agnostic views, manual override benefits from an analytical abstraction decoupled from any inferential machinery. However, overrides may prove problematic with dynamic data: should overrides persist when modeling incoming data?

3.4.4 Progressive Disclosure

By abstracting source data, models can improve scalability, surface higher-order patterns and suppress noise. However, they might also discard relevant information. To compensate, model-driven visualizations can enable analysts to shift among levels of abstraction on-demand. *Progressive disclosure* is the strategy of drilling down from high-level overview, to intermediate abstractions, and eventually to the underlying data itself. Progressive disclosure balances the benefit of large-scale discovery using models with the need for verification to gain trust. A tool can support reasoning and improve interpretation by displaying the right level of detail when it is needed. The critical concerns are that detailed data (1) is revealed on an as-needed basis to avoid clutter and (2) highlights the connections between levels of abstraction to aid verification. I identify two primary interaction techniques for achieving progressive disclosure: semantic zooming [8] and linked highlighting (a.k.a. "brushing and linking") [7].

Disclosure via Semantic Zooming

Semantic zooming changes the visible properties of an information space based on the current "zoom" level, exposing additional detail within an existing view. Using semantic zooming for progressive disclosure entails incorporating elements across different levels of modeling abstraction. The Dissertation Browser uses semantic zooming to move from department view to thesis view: individual dissertations are visualized in relation to the higher-level departmental structure. I hypothesize that semantic zooming is particularly effective for facilitating interpretation if it can show the next level of abstraction within the context of an established abstraction. Semantic zooming relies on a hierarchical organization of relevant model abstractions or metadata.

Disclosure via Linked Highlighting

Another option is to present different levels of analytical abstraction in distinct visualizations. Linked selection and highlighting between views can then enable investigation: given distinct visualizations at different levels of abstraction (e.g., a network of extracted entities and a document viewer) highlight the cross-abstraction connections (e.g., the occurrences of the entity in the document). Perhaps the simplest case is showing details-on-demand. The Dissertation Browser shows the source text of a dissertation abstract in a separate panel when a thesis is selected. Linked highlighting is desirable if the different levels of abstraction are more effectively presented using disjoint visual encodings — that is, when combining levels via semantic zooming is either impossible or inadvisable. When faced with non-hierarchical relations or simultaneous inspection of three or more levels of abstraction, linked views are likely to be preferable to semantic zooming.

Choosing Levels of Analytical Abstraction

A primary design challenge for progressive disclosure is to select the proper levels of abstraction. I consider this an instance of (vertical) model alignment that depends on

the identified units of analysis. Another outstanding question is how "deep" progressive disclosure should go. For example, comments from Dissertation Browser users suggest that my design would be further improved by incorporating word-level details to aid verification of thesis-level similarities (e.g., what words does Civil Engineering "borrow" from Biology?). In most instances, I find that progressive disclosure should terminate in the original source data, enabling analysts to connect model abstractions to the raw input.

3.5 Visualizations for Assessing Topical Quality

In this section, I demonstrate how visualizations can enable effective use and deployment of statistical topic models. When applying topic modeling to real-world analysis, a recurring task in my own experiences and as documented in the literature is the evaluation of topical quality. An effective means for assessing topical quality is thus an important step toward making topic models more useful for analyses.

Existing literature suggests that the quality of a topic is often determined by the coherence of its constituent words [1] and its relative importance to the analysis task [172] in comparison to other topics. However, in many documented cases, evaluation is done by users visually inspecting lists of words — a representation that is ill-suited for quality assessment or topical comparisons.

In response, I developed Termite (Figure 3.10), a visual analysis tool designed for inspecting the topical term distributions produced by a topic model. My tool contributes two novel techniques to aid topical quality assessment. First, I describe a *saliency measure* for ranking and filtering terms. By surfacing more discriminative terms, my measure enables faster assessment and comparison of topics. Second, I introduce a *seriation method* for sorting terms to reveal clustering patterns. My technique has two desirable properties: preservation of term reading order and early termination when sorting subsets of words. I demonstrate how these techniques enable rapid classification of coherent or junk topics and reveal topical overlap.

Figure 3.10: The Termite system consists of a matrix of term-topic distributions (left), with support for filtering and ordering by terms, ordering by topics, and drilling down to a specific topic. When a topic is selected in the term-topic matrix, the system displays word frequency distribution relative to the full corpus (middle) and the most representative documents (right).

3.5.1 Design Goals

My goal is to create a tool that supports the effective evaluation of term distributions associated with latent topics. The tool should help with assessing the quality of individual topics and all topics as a whole. In particular, I examine how to select appropriate descriptive terms to aid rapid impression formation; incorporate term relatedness to reveal high-level patterns and improve readability; and provide context for more in-depth analysis.

LDA topics are multinomial probability distributions over terms. At present, the evaluation of topical quality relies heavily on experts examining lists of most probable words for a given topic in descending order [28, 62, 106, 116] (e.g., "dna, replication, rna, repair, complex, interaction, base, ..."). I highlight relevant visualization design and describe the potential techniques that may better support the task.

Prior work shows that an appropriate *choice of descriptive terms* can aid comparison and understanding, and suggest design alternatives to displaying the most frequent words. Parallel tag clouds [42] apply G^2 statistics [51] to identify words that are over-represented and under-represented (i.e., not just merely frequent in absolute counts), and find that they aid comparison between groups of documents. Review Spotlight [177] presents adjective-noun pairs (i.e., phrases instead of words) to better capture the notion of sentiments in restaurant reviews (e.g., "good service" instead of "service"); participants are able to form more detailed impressions faster. Tag clouds [163] are a natural generalization of displaying top terms; both are aimed at supporting initial assessment of word distributions. Notably, tag clouds presents text as a bag of words, which matches the underlying language model.

Word relatedness can be incorporated into the visualization to surface high-level patterns in the text. DocuBurst [41] leverages hypernyms (from WordNet [104]) to radially layout words in a document to show hierarchical relationships between terms. The layout reveals patterns within or between texts, and enables comparison across multiple documents. TileBar infers word relatedness by co-occurrence [65]. The presentation enables users to make better judgments about the potential relevance of a document.

Reading and language comprehension often requires *context*. Interpreting individual words in isolation can be difficult and error-prone. A common technique is to display the original text, and highlight relevant terms within the source document. Concordance [169] shows all sentences in which a word occurs, and enable analysts to read the original text to gain deeper understanding. WordTree [170] shows all words following a selected term based on tree layout. Branches of the tree can reveal frequent word sequences.

3.5.2 The Termite System

Termite consists of a matrix of term-topic distributions, with support for filtering and ordering by terms, ordering by topics, and drilling down to a specific topic to reveal related documents. All results described in this paper are based on the LDA models [124] with 25 to 50 topics, trained on abstracts from 372 papers published in IEEE Information Visualization Conferences from 1995 to 2010 [148].

Term-Topic Matrix

The term-topic matrix (Figures 3.10 and 3.11) shows term distributions for all latent topics. Unlike lists of per-topic words (the current standard practice), matrices support comparison across both topics and terms. For a matrix view to be effective I must address multiple design criteria.

I use circular area to encode term probabilities. Texts typically exhibit long tails of low probability words. Any encoding choice must deal with the long tail in a term distribution. Area has a higher dynamic range than length encodings (quadratic vs. linear scaling). Curvature enables perception of area even when the circles overlap; overlap is unavoidable in order to retain sufficient resolution for a list including less-frequent terms. I also experimented with a parallel tag cloud [42] presentation where text is displayed directly in the matrix; the resulting visualization was not sufficiently compact for even a modest number of terms.

Displaying Informative Terms

Showing all words in the term-topic matrix is neither desirable nor feasible due to large vocabularies with thousands of words. Termite can filter the display to show the most probable or salient terms. Users can choose between 10 and 250 terms. For most reasonable large displays, setting N over 250 causes significant amount of scrolling and reduces the effectiveness of the visualization.

I define *term saliency* as follows. For a given word w, I compute its conditional probability $P(T|w)$: the likelihood that observed word w was generated by latent topic T. I also compute the marginal probability $P(T)$: the likelihood that any randomly-selected word w' was generated by topic T. I define the *distinctiveness* of word w as the Kullback-Leibler divergence [83] between $P(T|w)$ and $P(T)$:

$$distinctiveness(w) = \sum_{T} P(T|w) \log \frac{P(T|w)}{P(T)}$$

This formulation describes (in an information-theoretic sense) how informative the specific term w is for determining the generating topic, versus a randomly-selected term w'. For example, if a word w occurs in all topics, observing the word tells us little about the document's topical mixture; thus the word would receive a low distinctiveness score. The *saliency* of a term is defined by the product:

$$saliency(w) = P(w) \times distinctiveness(w)$$

As shown in Figure 3.11, filtering terms by saliency can aid rapid classification and disambiguation of topics. Given the same number of words, the list of most probable terms contains more generic words (e.g., "based, paper, approach") than the list of distinctive terms (e.g., "tree, context, tasks"). My saliency measure speeds identification of topical composition (e.g., Topic 6 on focus+context techniques). By producing a more sparse term-topic matrix, my measure can enable faster differentiation among the topics and identification of potential junk topics lacking salient terms.

Ordering the Term-Topic Matrix

Termite provides three options for *term ordering*: alphabetically to aid scanning, by frequency, or using seriation. Seriation permutes the presentation order to reveal clustering structure, and are commonly used to improve visualizations of matrices [90] or cluster heatmaps [173].

Termite uses a novel *seriation* method for text data. First, I define an asymmetric similarity measure to account for co-occurrence and collocation likelihood between all pairs of words. Collocation defines the probability that a phrase (sequence of words) occurs more often in a corpus than would be expected by chance, and is an asymmetric measure. For example, "social networks" is a likely phrase; "networks social" is not. Incorporating collocation favors adjacent words that form meaningful phrases, in the correct reading order. I compute the likelihoods using G^2 statistics [51].

G^2 estimates the likelihood of an event v taking place when another event u is also observed. The likelihood is computed using the following 2×2 contingency table:

events	u	$\neg u$
v	$a = P(u\|v)$	$b = P(\neg u\|v)$
$\neg v$	$c = P(u\|\neg v)$	$d = P(\neg u\|\neg v)$

The G^2 statistic is then defined as:

$$G^2 = a \log \frac{a(c+d)}{c(a+b)} + b \log \frac{b(c+d)}{d(a+b)}$$

For word co-occurrences, G^2 represents the likelihood of a word v appearing in a document/sentence when another word u also appears in the same document/sentence. For bigrams, G^2 examines all adjacent pairs of words, and estimates the likelihood of v being the second word when u is the first word.

I then place the terms according to their similarity scores by applying the Bond Energy Algorithm (BEA) [99]. I terminate BEA whenever a sorted sub-list with the desired number of terms is generated. Assessing topical composition typically requires examining only a subset of the common or mid-frequency words [94], and does not require seriating the full vocabulary. I use BEA because it accepts asymmetric

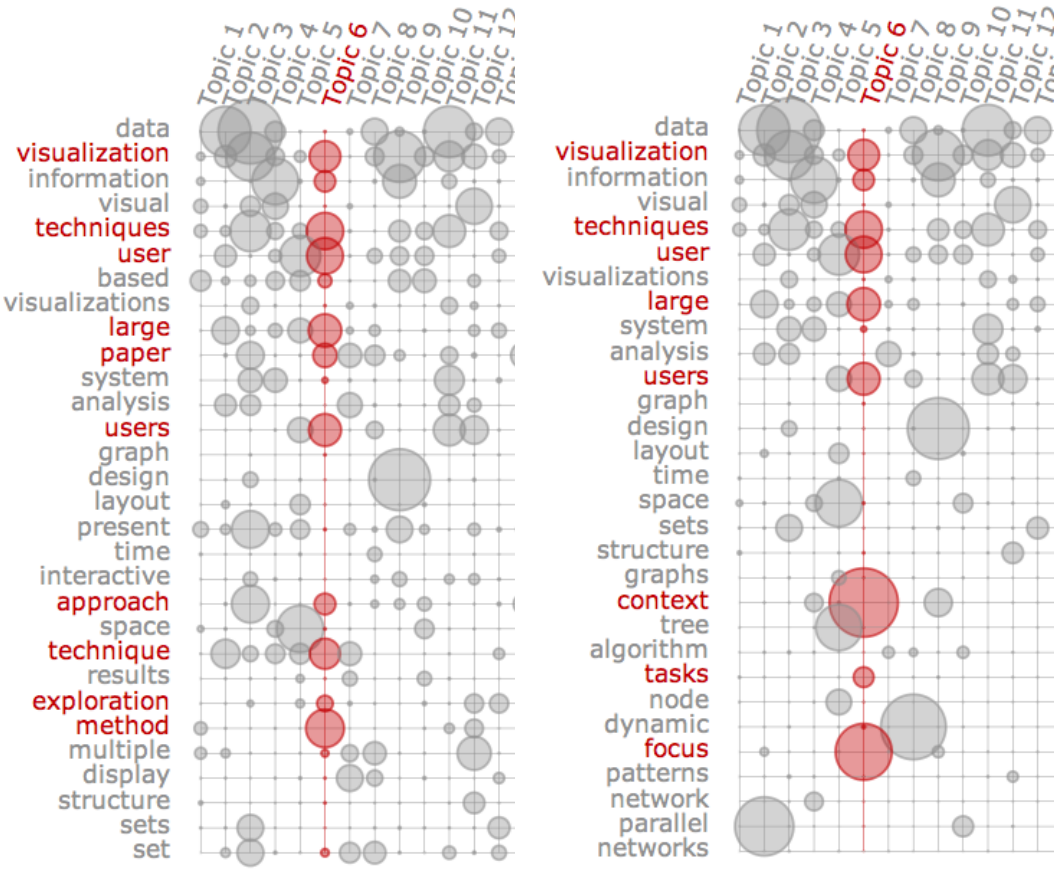

Figure 3.11: Top 30 frequent (left) vs. salient (right) terms. My saliency measure ranks "tree, context, tasks, focus, networks" above the more frequent but less informative words "based, paper, approach, technique, method." Distinctive terms enable speedier identification: Topic 6 concerns focus+context techniques, but this topical composition is ambiguous when examining the frequent terms.

similarity measures as input and is a greedy algorithm; early termination does not affect the quality of its results.

As shown in Figures 3.12 and 3.13, my seriation algorithm reveals topical clusters of terms. For example, my visualization enables rapid identification of coherent concepts, such as Topic 2 on parallel coordinates. Term grouping reveals shared properties between topics, e.g., "maintaining stability" in both treemaps and force-directed graph layout. My technique preserves reading order down the list of terms; examples include "online communities," "social networks," and "aspect ratio." Seriating terms in reading order facilitates scanning and a sense of term use in context.

Qualitatively, I observe that seriating terms using a combined similarity measure based on both document and sentence level co-occurrence is preferable to either statistics alone. Bigram likelihood produces a significantly sparser matrix than does document co-occurrence alone. As a result, adding bigram likelihood doesn't significantly change the global seriation order. Instead, it affects local orderings and places words such as "parallel coordinates," "user interface," "social networks," and "small multiples" in the correct reading order. I experimented with trigram statistics, but find that it degrades the overall seriation quality. Longer phases such as "node link diagram" are already produced by bigram statistics. Adding trigrams yields marginal gains and produces phrases such as "graph layout algorithm," "large data set," and "social network analysis." However, adding trigram likelihood leads to false positives: because the stop word "of" is omitted, the recurring trigram "level of detail" adds undesirable weight to the word sequence "level detail."

Termite also provides two options for *topic ordering*: default (i.e., order in which topic is generated by LDA) and by topical weight. Prior work suggests that small topics tend to contain more nonsensical and incoherent terms. Topic ordering by weight may surface such patterns.

Examining a Single Topic

When a topic is selected in the term-topic view (i.e., clicking on a circle or topic label in the matrix), the visualization show two additional views. Word frequency view (middle of Figure 3.10) shows the topic's word distribution relative the full

corpus. Document view (right of Figure 3.10) highlights topical terms within the most representative documents.

3.5.3 User Feedback

Distinctiveness Measure

I observe that filtering terms by distinctiveness supports faster assessment of topical quality as shown in Figure 3.11. Given the same number of words, the list of probable terms contains more generic words (e.g., "based, paper, approach") than the list of distinctive terms (e.g., "free, context, tasks"). My distinctive measure enables quick identification of topical composition for single topics, e.g. Topic 6 on focus+context techniques. By producing a more sparse term-topic matrix, my distinctive measure also enables faster differentiation of topics, and identification of potential "junk topics" that lack any significant descriptive terms. One initial concern is that distinctiveness measure might over-compensate and produce too many rare words, but I did not observe any such issues over the range of model settings.

I do observe value in retaining the option of showing frequent terms. While frequent terms do not yield sufficient details, they can reassure users that the model is doing reasonably well (e.g., top words in InfoVis are "data" and "visualization") at initial inspection whenever a new model output is loaded into the visualization.

Term Seriation

Seriation reveals a much clearer clustering structure among terms. In Figure 3.12, my visualization enables rapid identification of coherent concepts, such as Topic 2 on parallel coordinates, etc. Term grouping reveals shared properties between topics, e.g. "maintaining stability" between treemaps and force-directed graph layout, etc. Also, my technique preserves reading order down the list of terms, e.g. "online communities," "social networks," etc. Seriating terms in reading order facilitates scanning and imparts some sense of term use in context.

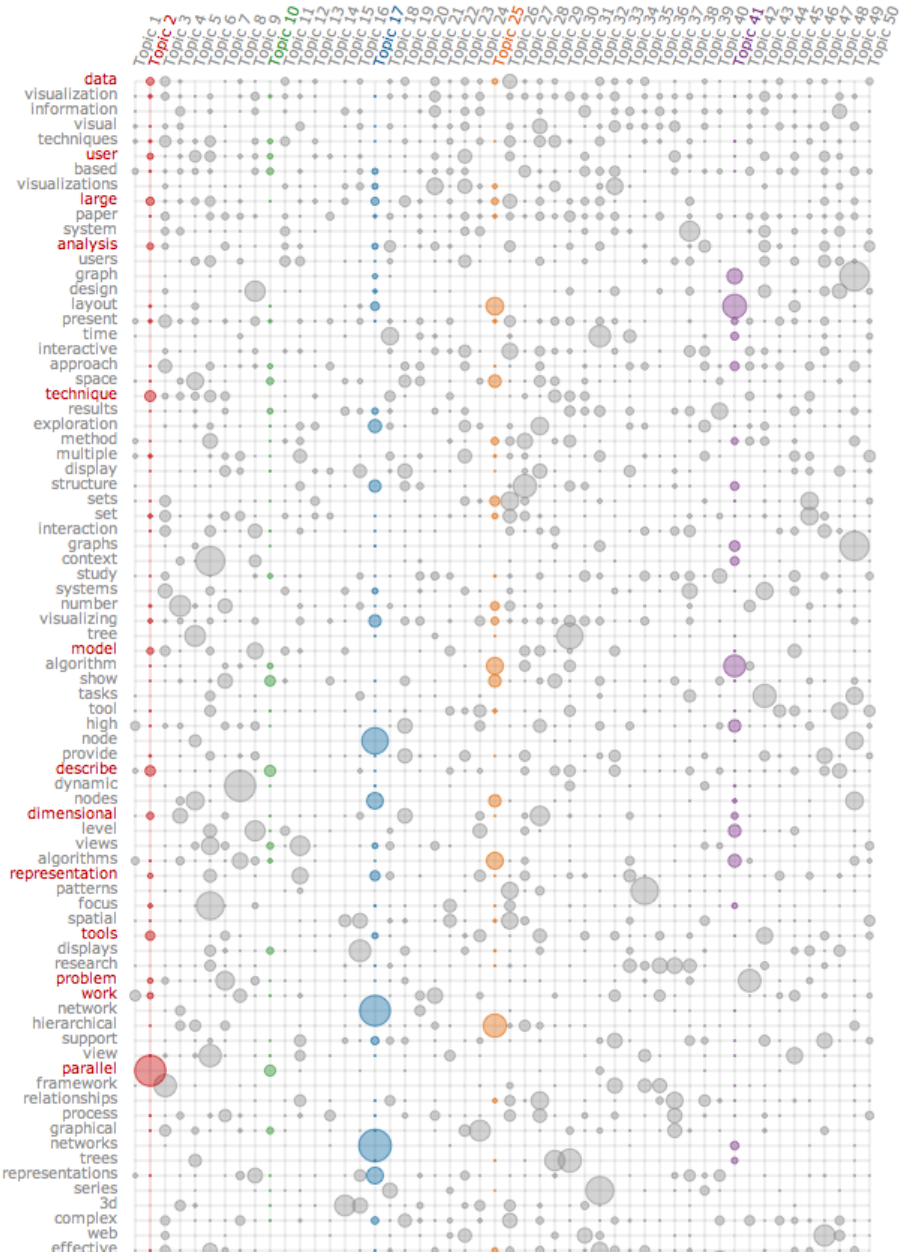

Figure 3.12: Terms ordered by frequency. Compare with my seriation technique in Figure 3.13. Discerning high-level patterns can be difficult when words are listed by decreasing frequency.

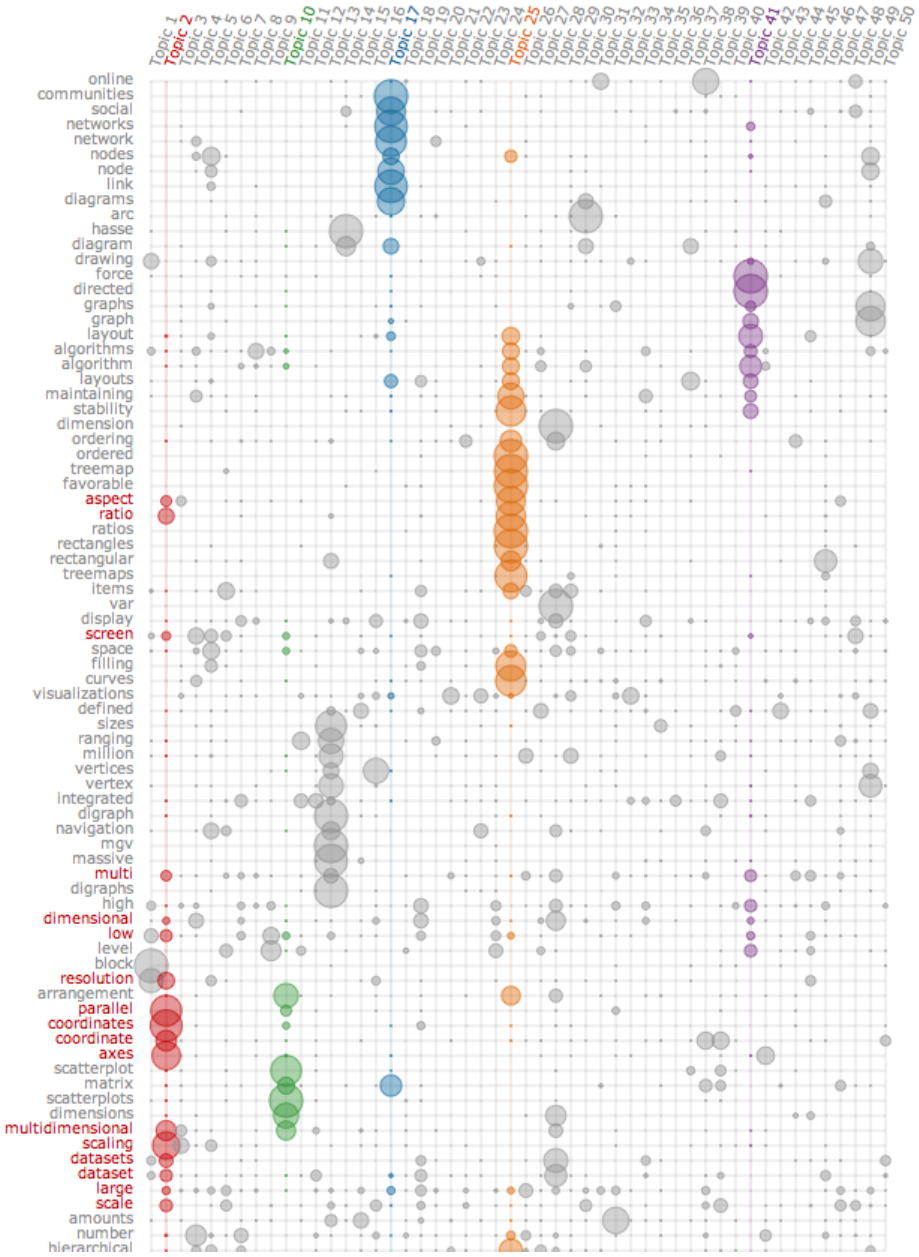

Figure 3.13: My seriation technique. Compare with term ordering by frequency in Figure 3.12. Seriation reveals clusters of terms and aids identification of coherent concepts such as Topic 2 (parallel coordinates), Topic 17 (network visualization), Topic 25 (treemaps), and Topic 41 (graph layout). My term similarity measure embeds word ordering and favors reading order (e.g., "online communities," "social networks," and "aspect ratio").

Qualitatively, I observe that seriating terms based on combined similarity measures from document and sentence co-occurrence seems to be better than either co-occurrence statistics alone.

The similarity measure for bigram statistics results in a significantly more sparse matrix than co-occurrence counts. As a result, bigram statistics don't significantly change the global seriation order much. However, they do affect local orderings: it will place words such as "parallel coordinates", "user interface," "social networks," "node link diagrams," and "small multiples" adjacent to each other in the correct reading order. Adding trigram statistics doesn't add much beyond what bigram statistics provide, either algorithmically or semantically. Adding trigram statistics leads to term orders such as "node link diagram" (already produced by bigram statistics) and less informative examples such as "graph layout algorithm," "large data set," and "social network analysis." However, there are also false negatives: because "of" is omitted, the recurring trigram "level of detail" adds undesirable weight to the term sequence "level detail."

Based on usage by members of my research group, I observed that users are able to meaningfully comprehend topical composition with Termite. Example quotes include: "The current [dataset] seems to overfit in places... much more so than the 30 topic example I used in [a previous iteration]" and "We may have single-doc topics!" I also received initial feedback requesting the ability to label and organize topics and examine document-topic probabilities.

3.5.4 Deployment and Future Releases

Termite is a first step towards a visual analysis system for human-centered iterative topic modeling. In this section, I focused on understanding terms and term-topic distributions. We publicly released the source code for Termite in February 2013, and are currently expanding Termite to visualize the topical composition of documents and adding interactions to support user inputs (e.g., adjusting model parameters, deleting junk topics, merging related topics). I believe supporting interactive model refinement can significantly improve the utility and reduce the cost of applying topic

models to make sense of large text corpora.

3.6 Gulfs of Evaluation and Next Steps

In this chapter, I examined how visualizations can be applied to support model-driven analysis as well as the design and deployment of the models themselves. I proposed *interpretation* and *trust* as criteria to guide the design of model-driven visualizations. I demonstrated that creating effective model-driven visualization requires considerations of both the visualization and the underlying model and that a *human-centered iterative design process* can produce effective tools.

3.6.1 Visual Assessment, Modeling Error, and Bias

Researchers in information visualization and machine learning have traditionally focused on the design of effective visualizations and the design of high-performance models in their respective fields. In the context of supporting model-driven visual analysis, however, the *analytic process* requires that all components work together.

My principle of interpretation can be viewed as a measure of *gulf of evaluation* [75] in the analytic process. As illustrated in Figure 3.14, model-driven visual analysis depends on a chain of data transformation, visual assessment, and communication tasks. Efficient visual encoding minimizes the *gulf of visual assessment* between a user and the visual presentation — which might depict either the source data or the data transformed according to some modeling abstraction. Model performance refers to the amount of *modeling error* which measures how well the source data fit a model's abstraction under some parameter setting. High-performance models can minimize modeling error under a suitable choice of parameters. The quality of a model-driven visualization, however, also depends on how well the modeling abstractions match an intended analysis task in a specific domain, a gap I refer to as the *modeling bias*. All three types of discrepancies — visual assessment, modeling error, and bias — contribute to the gulfs of evaluation. To support effective visual reasoning, a tool must account for interpretability issues at all levels.

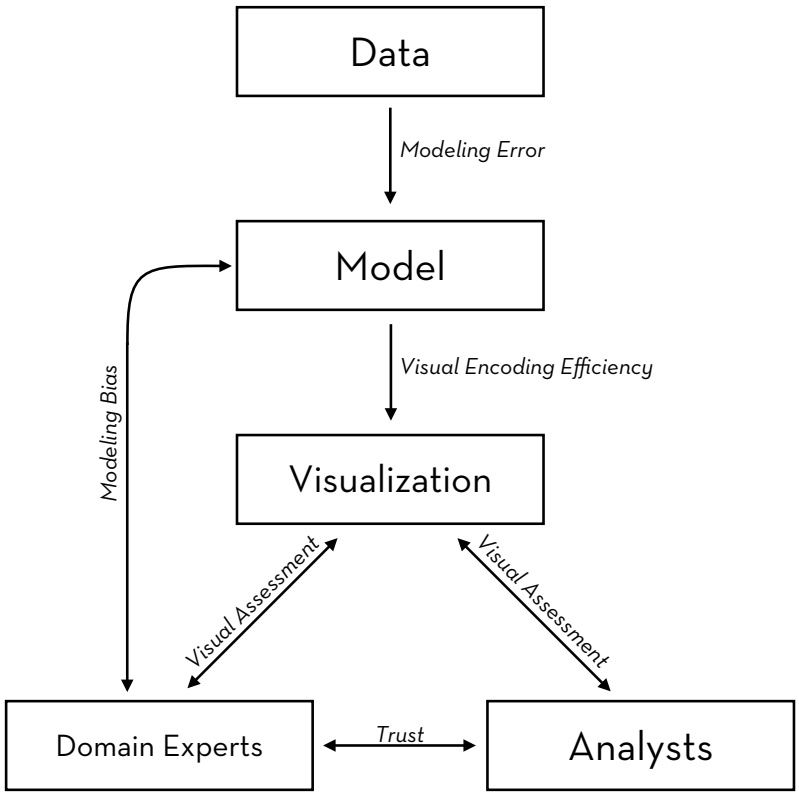

Figure 3.14: Interpretation, trust, and gulfs of evaluation in the model-driven visual analysis process. Visual assessment, modeling error, and bias all contribute to gulfs of evaluation in the analytic process.

Applying human-centered design methods to the design of models can reduce modeling bias. In my projects, by eliciting expert feedback, I identified discrepancies between modeling assumptions and known characteristics of the a domain (for example, symmetric similarity measures vs. directional departmental relationships in Section 3.2.5); revealed problems when real-world data deviated from idealized model representation (for example, cyclical citations in Section 3.3.1); and traded off modeling error to reduce bias (for example, LDA model under optimal settings grouped humanities into a single topic which minimized error according to the model's intrinsic measures but violated common sense in Figure 3.5). Expert verification also helps instill trust that the modeling results accurately reflect meaningful facts in context.

An iterative design process allows us to examine the effectiveness of all components

of a visual analysis tool. In many cases, I arrived at my design, not by piecewise optimizing the visualization or model performance but by designing an appropriate model that reduced the gulfs of evaluation at multiple levels.

Finally, the need for models will continue to grow in the face of big data applications. Statistical models enabled our analysis of language transfer at the national level (Section 3.3.2) which otherwise would not be possible using only traditional survey methods. My model design strategies (align, verify, modify, and progressive disclosure) introduced in Section 3.4 can serve as practical aids for designers and practitioners who wish to achieve interpretability and trustworthiness in their model-driven visual analysis tools.

3.6.2 Next Steps

Going forward, in the next chapter, I revisit the process of statistical topic modeling. While expert verification can help validate specific model and visualization designs, inspecting individual models requires human attention to every model instance and does not scale. I rethink the model design process and examine how we might elicit domain knowledge once so that we can use the information to evaluate any number of models afterward. I build on Termite and examine how we might leverage interactive visualization to support machine learning research, demonstrate its effective use, and improve its relevance in domain-specific analyses.

Chapter 4

Expert Organization of Text Corpora

As demonstrated in the previous chapter, while fitting statistical topic models typically involves unsupervised learning algorithms, applying these models to real-world analysis tasks requires a significant amount of human-in-the-loop supervision. For example, automatically extracted topics often need to be manually verified to ensure they are semantically meaningful within the domain of analysis. Eliciting human judgment, however, is a time-consuming task and can dominate the total amount of effort involved in building high-quality topic models.

In this chapter, I investigate how we can reduce the cost of acquiring domain expertise and increase its utilization in the modeling process. First, I conduct a survey experiment in which I ask ten experienced information visualization researchers to characterize the significant research topics in their field. My analysis of the resulting the topical concepts enables domain-specific evaluations of topic modeling practices. I then introduce a framework that enables large-scale assessment of topical relevance by aligning the outputs from any number of topic models to a common set of reference concepts. Diagnostic information generated by this framework can contribute to topic modeling research (e.g. studying the effects of model parameters).

4.1 Human Supervision in the Analytic Process

Human experts often utilize categorization [32] to process a large quantity of data and support effective reasoning [160]. Categories represent how people mentally organize or *chunk* information [97, 103] into groups comprised of items that share common attributes or functions. Established design principles [117] and case studies [157] suggest that incorporating informative categories [84, 137] into analysis tools can enable effective sensemaking [139] and efficient communication [43].

On this basis, an often stated goal of statistical topic modeling is to extract a semantic space [50] or structured representation [40] that corresponds to human information organization [85]. Expectedly, data analysts are eager to utilize topic models to analyze document collections too large for any one person to read. However, while topic modeling may be an unsupervised learning algorithm, applying them to real-world analysis tasks requires a significant amount of human-in-the-loop supervision. I begin this chapter with an examination of the manual effort involved in model-driven data analysis.

4.1.1 The Need for Reusable Diagnostic Feedback

As demonstrated in Chapter 3, discrepancies between statistically extracted topics and domain concepts abound. Creating domain-relevant models often requires that latent topics be manually inspected and model assumptions verified. In many cases, analysts may construct multiple models or re-train them using different parameter settings. Expert evaluation is then needed to compare the models and select a suitable one. In relation to other stages of the model design process that can be automated, these human judgment tasks can dominate the time and cost of building high-quality topic models.

At the present, when experts are employed to evaluate models [62, 152], they are typically tasked with validating latent topics after a model is created. In such a workflow, expert responses are tied to a specific model and cannot be reused in subsequent analyses. Even though tools such as Termite can aid the interpretation of topics, some tasks (e.g., model modification) may require experts to express their

knowledge about a subject matter. For example, analysts frequently remove terms or *stopwords* that are deemed low in information content from a model's vocabulary. As illustrated in Section 3.3.1, experts may organize their knowledge about a discipline based on units of analysis (e.g., authors and seminal papers) that differ from a model's representation (e.g., flows of words). One might reasonably suspect whether people can efficiently or accurately articulate domain concepts as a bag of words. Rethinking when, what, and how to elicit user input might reduce the cost of acquiring domain expertise and increase its utilization.

While automatic evaluation methods are available, such as statistical [11] or coherence [28, 115] measures, they can be problematic in domain-specific settings because they do not account for the notion of relevance. Many of these techniques target the identification of junk topics [1] comprised of a nonsensical collection of words. However, poor topical quality can be attributed to various other factors: for example, words that represents a mix of two distinct concepts [116] or words that are deemed irrelevant to the domain [1]. Also, many evaluation techniques typically produce only a single goodness-of-fit likelihood measure. As the analytic process is iterative, interpretable diagnostic feedback on how (i.e., not just how much) a model differs from expectation can be valuable in informing analysts on possible approaches for improving the model.

4.1.2 Chapter Outline

In this chapter, I propose an alternative workflow in which we begin the model design process by first eliciting topical categorizations from human experts. Using a computational framework that measures topical correspondence, I then apply the acquired concepts to explore a large space of model designs. Not only does my approach enable large-scale assessment of topical relevance, my work also contributes to various aspects of topic modeling research including the evaluation of current practices, potential future models, and parameter choices.

To support the investigation, I develop a survey method to collect expert topical organization in Section 4.2. I identify and address issues (i.e., bias, recall, accuracy,

participant exhaustion) associated with eliciting free-form categorization responses. I conducted a survey and collected 202 topical responses from ten experienced researchers in information visualization (InfoVis).

In Section 4.3, I introduce a method for topical aggregation of InfoVis survey responses and their subsequent validation. I synthesize a set of 28 most coherent concepts in InfoVis, based on three high-precision, low-recall predictors of topical similarity. My analysis reveals that human topical concepts may be defined through a multifaceted set of attributes.

In Section 4.4, I demonstrate that establishing a reference set of expert-provided concepts can enable novel approaches to evaluating topic modeling practices. I construct three sets of theoretically optimal word-based topic models. By measuring the amount of mutual information between model outputs and the expert concepts, I quantify the limits of both existing latent Dirichlet allocation (LDA) [11] models and potential future models. The results allow me to place an upper bound on the proportion of the expert concepts that can be recovered using only word co-occurrence statistics, based on document abstracts vs. the full text. Finally, I evaluate LDA model outputs directly in terms of experts' organization of a domain.

In Section 4.5, I introduce a framework for measuring the topical alignment between a set of latent topics and a set of reference concepts. My framework enables large-scale assessment of topical relevance by enabling comparison of any number models to a common set of expert concepts. I say a topic *resolves* to a concept if a one-to-one correspondence exists between the two, and recognize four types of misalignment: when models produce *junk* or *fused* topics or when reference concepts are *missing* or *repeated* among the latent topics. To compute an alignment, I estimate the likelihood that a topic-concept pair would be considered equivalent by human judges, based on a user study on Amazon Mechanical Turk. To ensure the stability of my alignment measures for large-scale comparisons, I also contribute a method for estimating and discounting topical correspondences that can be attributed to random chance via a generative probabilistic process.

Finally, in Section 4.6, I present the findings from an exploratory process of topic model construction. I create LDA models trained using 10,816 parameter settings.

By evaluating the resulting 569,000 latent topics against the 202 InfoVis concepts provided by experts using my framework, I observe that a small change in term smoothing (β) can significantly alter the ratio of resolved and fused topics. In many cases, increasing the number of latent topics (N) leads to more junk and fused topics with a corresponding reduction in resolved topics. About 10% of the concepts are only uncovered within a narrow range of parameters. Treating a model's outputs as reference concepts, my framework can also provide diagnostic information on how two models differ.

4.2 Eliciting Expert Categorizations

In this section, I introduce a survey method for eliciting expert topical organization based on freeform responses. Through preliminary studies, I identify four issues (i.e., bias, recall, input accuracy, and participant exhaustion) associated with eliciting open-ended categorization responses, and devise user interface and survey design modifications to address these issues. Using the survey method, I asked ten experienced researchers to describe topics of information visualization research and received 202 hand-crafted topical responses, each consisting of a title, keyphrases, and representative documents.

4.2.1 Topical Domain and Participants

I focused on InfoVis research due to relevance, scope and familiarity. Analysis of academic publications is one of the common real-world uses of topic modeling [61, 130]. Our familiarity with the InfoVis community allowed us to contact experts capable of exhaustively enumerating its research areas. InfoVis has a single primary conference, simplifying the construction and analysis of its publications.

Survey recruitment was by invitation only. I contacted 23 researchers (12 past chairs of the IEEE Information Visualization Conference, six faculty members, two senior industry researchers, and three Ph.D. students within a year of graduation) on a rolling basis over four months from March to June 2012. Of the 14 surveys that were

sent out, I received ten completed results from four past chairs, two faculty members, one industry researcher, and three Ph.D. students. I initially limited the survey to only past conference chairs, but expanded the criteria to established researchers (including final year Ph.D. students) to enable greater participation.

4.2.2 Survey Design

I asked participants to describe topics using labels, terms, and documents they would use if communicating with a peer. Representative terms should *exemplify a topic and differentiate the topic from other areas of research.* Terms could be any notable techniques, methods, systems, or people. Both words and phrases (multi-word terms) were allowed. Representative documents should *exemplify the core contributions of a topic.* Pilot studies showed that citing a paper using freeform text was time consuming, disruptive to the recall process, and prone to errors. In response, I limited the representative papers to those published at IEEE InfoVis Conferences. To associate a paper with a topic, participants could drag-and-drop a paper entry into the topic boxes in the main panel. I requested that participants enter ten or more terms and three or more papers per topic, though fewer responses were permissible. I asked participants to complete the survey in a single session if possible.

Design Considerations

Conducted using a single webpage (Figure 4.1), I designed the survey to: (1) elicit expert responses with minimal bias, (2) support recall, (3) enable accurate data collection, and (4) balance between maximizing the value of available expert time and preventing participant exhaustion. To avoid artificially limiting what they consider to be the scope of information visualization research, the participants were instructed to consider work published anywhere when creating the research topics. Participants were provided with multiple blank boxes in the survey user interface. I asked subjects to list all areas they consider to be significant. The interface contained twenty boxes by default, but subjects could add additional boxes if desired.

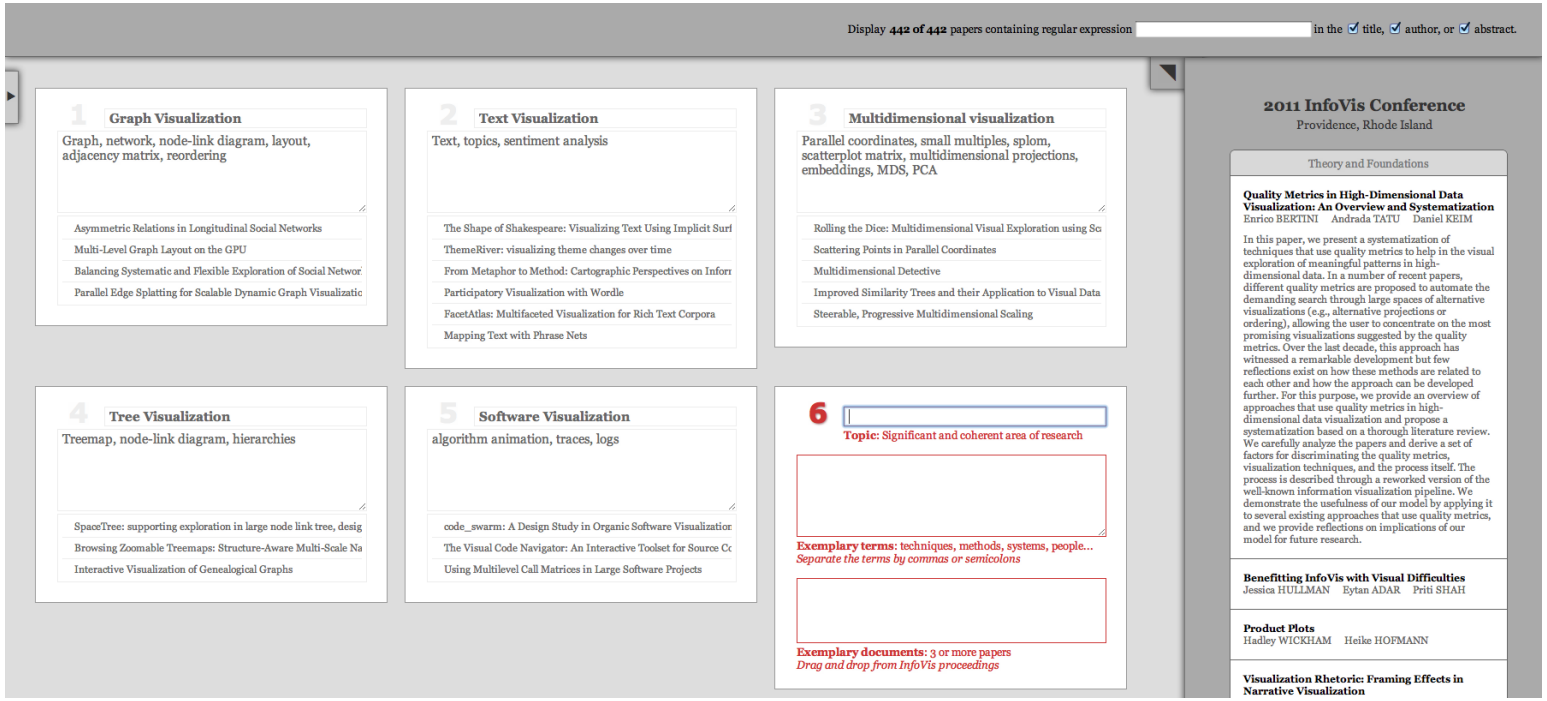

Figure 4.1: Survey user interface. Participants were provided with blank boxes in a single webpage, and asked to identify all *coherent and significant* areas of research in information visualization. An optional panel on the right shows 17 years of IEEE InfoVis Conference proceedings grouped first by year then by session.

In pilot studies, the single most prominent issue was recall. Exhaustively identifying all concepts in a domain purely from memory was difficult. In response, I added a panel on the right that contains a list of all 442 papers published at the IEEE InfoVis Conferences (1995 to 2011), grouped by year. As InfoVis is a single track conference, I grouped papers within each year by session, so the ordering of sessions and papers were consistent with the actual conference program. Participants could browse through the proceedings or search for specific papers by title, author, or abstract.

The most scarce resource in conducting the survey was acquiring available time from the experts. To maximize the value of their responses, I chose exemplary words and documents as the means to express a concept. Prior cognitive psychology studies on categorization typically identify categories by their labels. I worried that labels alone would not sufficiently capture the abstract concept of a research area. Based on pilot studies, the two chosen properties — freeform typing of a list of terms, and drag-and-drop specification of papers — minimized input complexity and allowed experts to focus on the construction of the categories. I omitted other descriptive attributes, such as summary sentences, which took pilot participants much longer to enter. I displayed twenty topic boxes by default to provide reasonably exhaustive coverage of the domain while bounding the length of the survey. In a preliminary study, my primary advisor and I exhaustively annotated every document in the corpus with multiple tags. The overlap between the two sets of annotations indicated that the domain was covered by approximately twenty shared topics.

Survey Data

I received a total of 202 topical responses (maximum of 22 and minimum of 18 per subject). The participants specified an average of 5.71 terms (max 19, min 1, median 8) and 5.15 documents (max 25, min 1, median 7) per topic. The subjects also provided 171 distinct topic labels (158 using case-insensitive comparison) and 769 distinct terms (747 case-insensitive). Together, the experts cited a total of 342 distinct documents (77% of all papers published at IEEE InfoVis Conferences) which I considered to be a reasonable coverage of the field.

I analyzed timing information for the seven of ten participants who had active internet connections for the full duration of their survey. The survey user interface automatically saved responses every minute, allowing me to track changes at that granularity. On average, the experts spent 91.7 minutes (max 162, min 42) editing their responses within a maximum of five sessions. The amount of editing time suggested that the survey taxed the experts' attention and available contiguous time.

4.3 Synthesizing Coherent Concepts in InfoVis

In this section, I synthesize and analyze the survey responses from the InfoVis experts. I create a topical similarity measure to identify matching topics from the different participants. My similarity measure is comprised of three independent high-precision, low-recall predictors using topic label, textual descriptions, and exemplary documents. The combined measure resolves matching topics with 92% precision as verified by four additional experts. In an analysis of the resulting set of 28 combined topics, I find that each attribute alone captures about 69% to 73% of the total topical contents in terms of mutual information.

In contrast to previous methods [137, 153] which examine only named categories, my synthesis approach can resolve topics without well-defined labels. During validation, many such unnamed topics are deemed equivalent by independent human judges, suggesting that topical concepts may be defined through a multifaceted set of attributes. As some topics lack a shared vocabulary, I hypothesize that content-based analyses cannot characterize the complete set of expert categories. I also discuss the lack of hierarchical organization in the dataset in relation to the cognitive psychology literature [84, 135, 136, 137].

4.3.1 Topical Resolution

To provide data for constructing the similarity measures, I examined 23 randomly-selected pairs of participants (half of 45 possible pairs). For each participant pair, I identified pairs of matching topics under the constraint that each topic can only

be matched once. I manually examined 9,117 topic pairs (out of 18,134 possible), finding 280 matching topics. On average, I find 12.2 matching topics per participant pair (max 16, min 9, median 12).

Prefixes and suffixes	Examples
... data	multidimensional data → multidimensional
... visualization(s)	graph visualization → graph
... data visualization(s)	text data visualization → text
... analysis	network analysis → network
... data analysis	social data analysis → social
... method(s)	navigation methods → navigation
... view(s)	focus and context views → focus and context
... paper(s)	evaluation papers → evaluation
... issues(s)	database issues → database
... technique(s)	interaction techniques → interaction
visual ...	visual perception → perception
visualization ...	visualization toolkits → toolkits

Table 4.1: List of 12 prefixes and suffixes removed from labels and terms

Label Similarity

Based on preliminary examination, topics with matching labels typically refer to the same concept. However, I also find that some labels contain a mixture of concepts. As a pre-processing step, I split labels on conjunctions ("Maps and Geospatial"), commas ("Spatial, Temporal"), slashes, and ampersand signs, and manually duplicate any substring that grammatically applies to both concepts ("Ambient/Casual Visualization" to "Ambient Visualization" and "Casual Visualization"). I make two exceptions for the cases "Focus and Context" and "Overview and Detail" that are known to be single coherent concepts in InfoVis. After pre-processing, 13% of labels are split into two sub-labels.

For each sub-label, I apply the following text processing. My intention is to minimize modification of the user data while producing higher quality string matching than is provided by naïve string comparison. I manually correct for misspellings,

and remove common prefixes and suffixes as shown in Table 4.1. I replace punctuations ("focus+context" to "focus and context"), and fold all responses to lower case. I manually rephrase verb phrases into noun phrases ("visualizing uncertainty" to "uncertainty visualization"), and convert adjectives and plurals into singular nouns ("hierarchical" to "hierarchy"). I modify individual words in only two cases from "bioinformatics" to "biology" and from "geospatial" to "geography." Based on domain knowledge, these two pairs of words typically refer to the same concepts within the context of InfoVis research. Altogether, I convert the 171 distinct label strings into 109 distinct sub-labels. I assign a label similarity of 1 if the set of sub-labels between two topics are identical, 0.5 if the set of the sub-labels intersect, 0 otherwise.

$$\text{LabelSim}(x, y) = \begin{cases} 1 & \text{if SubLabels}(x) = \text{SubLabels}(y) \\ 0.5 & \text{if SubLabels}(x) \cap \text{SubLabels}(y) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Textual Similarity

I devise a second predictor based on textual information associated with a topic. I assign a set of tags to each topic consisting of the list of exemplary terms given by the participants, plus the sub-labels from the previous step. I apply the same text processing to the terms. In addition, I manually identify 61 named persons and 31 project titles from the list, so they are properly resolved (e.g., from "shneiderman" to "Ben Shneiderman" and from "word tree" to "The Word Tree"). These text processing steps produce a total of 653 distinct tags. Each topic is assigned an average of 6.70 tags (max 20, min 2, median 9). For each tag, I tally its overall frequency in the corpus (the most frequent tag "geography" occurs 21 times, followed by "perception" at 18). Textual similarity between two topics is defined as the set overlap (Jaccard

Index) between their tags, weighted by log-transformed tag frequency:

$$\begin{aligned}\text{TextSim}(x,y) &= \frac{|\text{Tags}(x) \cap \text{Tags}(y)|}{|\text{Tags}(x) \cup \text{Tags}(y)|_{\log \text{Freq}}} \\ &= \frac{\sum_{t \in \text{Tags}(x) \cap \text{Tags}(y)} \log \text{Freq}(t)}{\sum_{t \in \text{Tags}(x) \cup \text{Tags}(y)} \log \text{Freq}(t)}\end{aligned}$$

Document Similarity

The third predictor is based on the expert-selected exemplary documents. I compute the overall citation count of each document within the collected data (the most expert-cited papers are "Ordered Treemap Layouts" [145], "Many Eyes: A Site for Visualization at Internet Scale" [165], and "D3: Data-Driven Documents" [15] at 10 times each). The document similarity between two topics is defined as the set overlap (Jaccard Index) between their representative documents, weighted by each document's log-frequency.

$$\begin{aligned}\text{DocSim}(x,y) &= \frac{|\text{Docs}(x) \cap \text{Docs}(y)|}{|\text{Docs}(x) \cup \text{Docs}(y)|_{\log \text{Freq}}} \\ &= \frac{\sum_{d \in \text{Docs}(x) \cap \text{Docs}(y)} \log \text{Freq}(d)}{\sum_{d \in \text{Docs}(x) \cup \text{Docs}(y)} \log \text{Freq}(d)}\end{aligned}$$

Topic Similarity

The final topical similarity between two topics is a linear combination of the three predictor outputs:

$$\text{Sim}(x,y) = \text{LabelSim}(x,y) + \text{TextSim}(x,y) + \text{DocSim}(x,y)$$

$$\text{TopicSim}(x,y) = \begin{cases} 1 & \text{if } \text{Sim}(x,y) \in (0.75, 3.00] \\ 0 & \text{if } \text{Sim}(x,y) \in [0.00, 0.30) \\ (\text{Sim}(x,y) - 0.3)/0.45 & \text{otherwise} \end{cases}$$

The thresholds 0.75 and 0.30 are chosen to achieve targets of 90% precision and

90% recall respectively. In other words, I design the topic similarity so that I can expect topic pairs with a similarity of 1.0 to be truly matching at least 90% of the time, and topic pairs with non-zero similarity to contain 90% of all matching topics. Empirically, of the 216 pairs of topics with a combined Sim score of 0.75 or higher, 195 pairs (91.1%) are annotated as matching in the training data. Of the 390 pairs of topics with a combined Sim score of 0.30 or higher, 252 pairs are annotated as matching, covering 90.0% of the 280 matching topics at a precision of 64.6%. Details are provided in Table 4.2.

Topical similarity	1	(0, 1)	0
Matching	195	57	28
Non-matching	19	119	8,699
Precision	91.1%	64.6%	
Recall	69.6%	90.0%	

Table 4.2: Precision and recall of my topical similarity measure, based on the author's annotation (training dataset). Topic pairs with a similarity of 1.0 are expected to be truly matching 90% of the time. Pairs with non-zero similarity are expected to contain 90% of all annotated matching topics.

Verification

I conducted a second survey to ensure that my similarity measure is not over-fitted to the training data and to ensure that annotations by the author are representative of expert consensus on matching concepts. Four experts (including three who were not in the original survey) were asked to verify the results. These four experts included two faculty members, one senior industry researcher, and one post doctoral researcher.

Verification participants were shown pairs of expert topic lists in a single webpage (Figure 4.2) and their corresponding lists of InfoVis topics. The participants were told that the topics are generated by fellow researchers and represent what their peers consider to the be the complete list of significant and coherent research areas in InfoVis. Pairs of topics, one from each list, were then selected and presented to the

participant in the same webpage. Selected topics were accompanied by their label, exemplary terms and documents. I asked participants if the selected topics were matching ("The two researchers are communicating the same concept."), partially matching ("The two topics have some overlapping content, but the two researchers may be referring to different concepts."), or not matching.

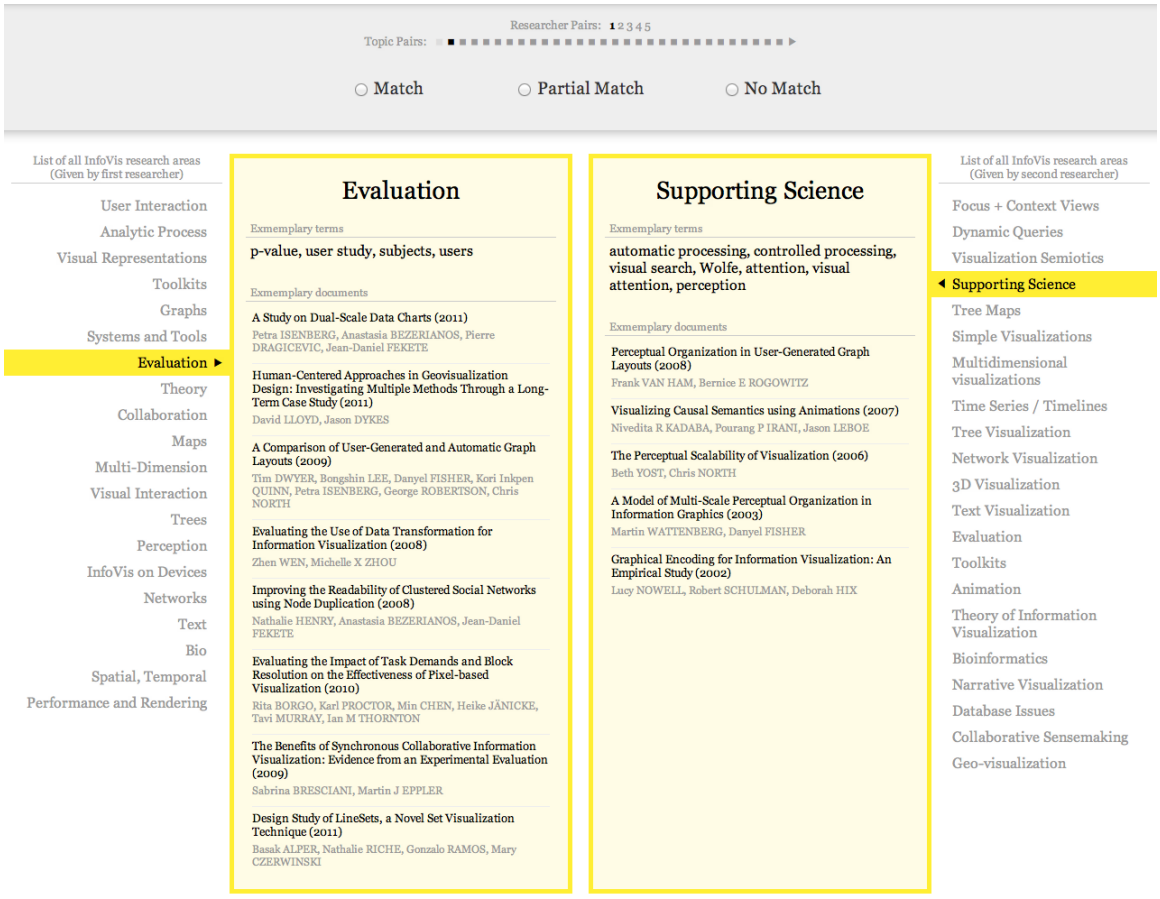

Figure 4.2: Verification user interface. Participants were provided with lists of responses from two experts at a time. Pairs of topics, one from each list, were then selected and presented to the participant who was asked to identify whether the topics were matching, partially matching, or not matching.

Each participant compared topic lists for five pairs of experts. The pairs of experts shown were not included training dataset (i.e., used to determine our similarity

metric). For each pair of experts, the participants were shown 30 pairs of randomly selected topics. The sampling process was designed so that topic pairs with a wide range of topical similarity scores are equally likely to be chosen. The results are summarized in Table 4.3. Counting each partial match as a prediction rate of 0.5, the verification results confirmed the precision of my topical similarity measure. The additional experts considered topic pairs with a similarity of 1.0 to be matching 93.1% of the time, and those with non-zero similarity to be matching 78% of the time. Applied to the full dataset, my similarity measure identified 405 pairs of matching topics and 335 pairs of partially matching topics (i.e., those with a non-zero similarity score), validating to the earlier reported results on the training dataset.

4.3.2 Coherent Concepts in Information Visualization

I create a matrix visualization that displays all pairwise topical similarities to help with identification of coherent groups of responses. Rows and columns of the matrix correspond to expert responses; topical similarities are visually encoded using circles at the interactions of rows and columns. I sort the matrix to place similar responses in close proximity and reveal high-level clustering.

I observe high levels of agreement among multiple sets of responses, which appear as blocks along on the diagonal. Examples include *Text* and *GeoVis* (Figure 4.3) and *Animation* (bottom-right of Figure 4.4). I mark each of these blocks as a coherent InfoVis topic comprised of all responses that make up the block. I also observe

Topical similarity	1	(0, 1)	0
Matching	183	40	1
Partial matching	27	66	37
Non-matching	1	27	213
Precision	93.1%	78.3%	

Table 4.3: Precision of my topical similarity measure, as verified by four additional experts. The expert find that topic pairs with a similarity of 1.0 to be matching 93% of the time, comparable to results obtained from the training dataset.

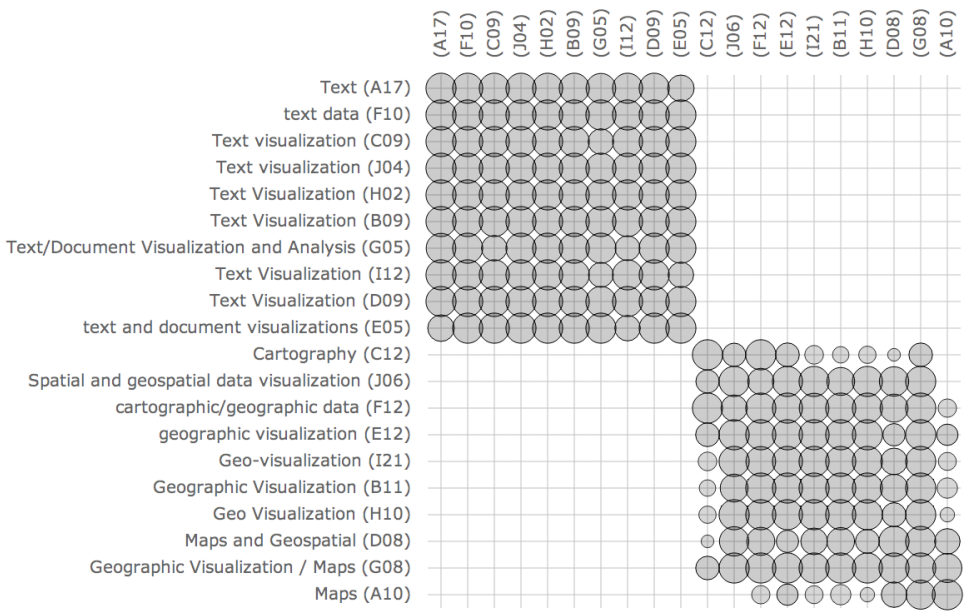

Figure 4.3: Submatrix of pairwise topical similarities. Each column and row corresponds to a single topical response provided by an expert. Areas of the circles represent similarity between the responses. Responses are seriated to surface concept grouping. Here, *Text* and *GeoVis* exhibit high levels of coherence and appear as blocks along the diagonals.

multiple pairs of overlapping blocks such as *Interaction Theories* and *Interaction Techniques* (top-left of Figure 4.4). I mark each of these blocks as a coherent topic, and note that these overlapping topics share underlying constituent responses.

In three instances, I observe a small 2×2 block attached to a larger block (i.e., *Focus and Context* attached to the larger *Overview and Detail*, *Statistical Visualization* attached to *Uncertainty Visualization*, and *Visualization beyond the Desktop* attached to *Devices*). In these cases, domain knowledge suggests that they likely refer to similar concepts; I group the smaller blocks into their corresponding larger blocks. Altogether, I identify 28 coherent InfoVis topics comprised of three or more responses. These topics consist of 14 independent topics and seven pairs of overlapping topics, as shown in Table 4.4.

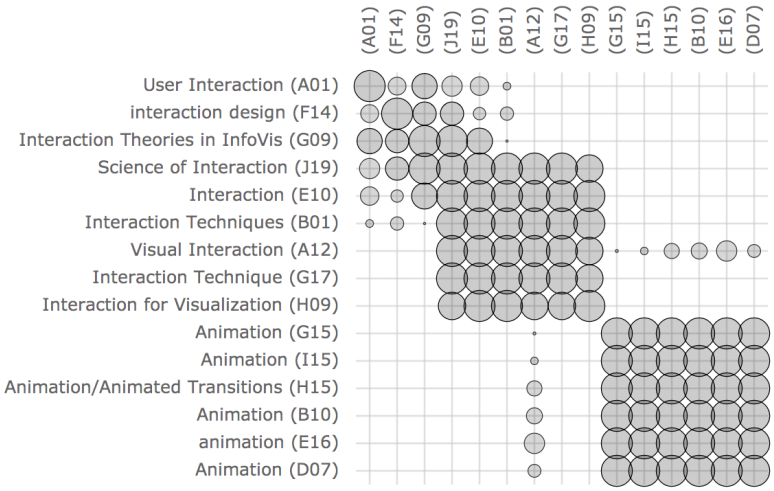

Figure 4.4: Submatrix of pairwise topical similarities. The upper-left corner contains responses that correspond to interactions. The two overlapping blocks suggest two distinct concept groupings that share common elements. Due to the lack of a coherent label, I refer to these concepts as *Interaction Theories* and *Interaction Techniques* respectively. In the bottom-right corner, I observe a coherent set of six responses corresponding to the topic *Animation*.

The Contributions of Labels, Terms, and Documents

Having identified a set of coherent concepts, I now examine the attributes with which experts define the topics. I first explore the data using visualizations and then quantify the results using information theoretic measures.

I visualize the contributions of the three predictors of topical similarity using matrix views. For example, the topic *Evaluation* shown in Figure 4.5 is well-named. Every expert but one labeled the topic "Evaluation" (the outlier labeled it "Purpose and Value"). All responses that make up the topic share a common vocabulary ("evaluation", "experiment", "qualitative", "quantitative", "user study"). This is indicated by the prominent block in the left submatrix, which shows the textual description similarities among the responses. However, the experts cite a wide variety of documents to represent the research area, without a single prominent paper, illustrated by the lack of structure in the right submatrix (document similarities).

For each coherent InfoVis topic identified in the previous section, I examine the

Topic	Size	Label	Text	Doc
Graphs [a]	9	✓	✓	
Networks [a]	9		✓	
Trees [b]	9	✓	✓	
Treemaps [b]	9		✓	✓
Multi-Dimensional [c]	10	✓	✓	✓
Parallel Coordinates [c]	7		✓	
Text	10	✓	✓	✓
GeoVis	10		✓	✓
BioVis	8		✓	✓
Time Series	8	✓	✓	✓
Uncertainty	6		✓	✓
Narrative	4		✓	✓
Software	3	✓	✓	✓
Devices	3		✓	✓
Evaluation	9	✓	✓	
Perception	8	✓	✓	
Cognition	8	✓		
Theory [d]	5	✓		✓
Collaboration [d]	5	✓	✓	✓
Social [e]	8			✓
For the Masses [e]	8			✓
Toolkits [f]	10	✓	✓	✓
Systems [f]	4		✓	
Interaction Theories [g]	5		✓	
Interaction Techniques [g]	6	✓		
Animation	6	✓	✓	✓
Overview and Detail	6		✓	
Multiple Views	4		✓	

Table 4.4: The list of 28 InfoVis topics identified by at least three experts. *Size* refers to the number of experts who identify the topic. A marker in the *Label* column indicates that the experts assign a coherent label to the topic. *Text* indicates the presence of coherent textual descriptors (label or exemplary term). *Doc* indicates the citation of a common exemplary paper. Superscripts indicate overlapping topics that share responses.

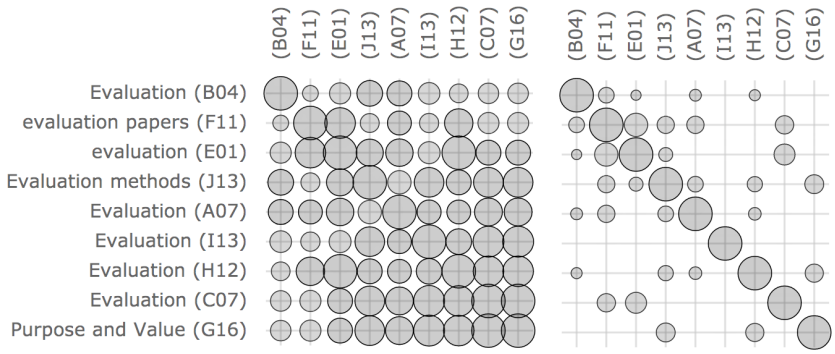

Figure 4.5: Submatrices of textual similarities (left) and document similarities (right). The label *Evaluation* is well-defined: all but one expert assigned the same name. The topic is unified by a common vocabulary, as indicated by the block on the left. However, the topic lacks a unifying document, exemplified by the lack of structure on the right.

whether its constituent responses (1) are assigned a coherent label, (2) contain a coherent set of textual descriptions, and (3) contain a common exemplary paper. I codify a topic as unified by *label*, *text*, and *document* accordingly as shown in the right columns in Table 4.4.

Previous psychology studies typically recognize categories as well-named (or labeled) concepts. Here, I find a heterogeneous combination of attributes used to define the abstract category of a research area. In fact, among the 28 InfoVis topics, I observe all seven combinations of *label*, *text*, and *document* used to define topics as shown in Table 4.5.

To better quantify the contributions of the three attributes, I examine the amount of topic information that can be communicated among the experts. Given two experts x and y, I create a joint probability distribution $P(x, y)$ that specifies how likely a concept x_i given by expert x matches a concept y_j given by expert y.

$$P(x_i, y_j) \propto \text{TopicSim}(x_i, y_i)$$

The mutual information between the marginal distributions $P(x)$ and $P(y)$ measures how much knowing the set of concepts given x informs us about the set of

Unifying attributes	Number of topics
Label + Text + Doc	7
Text + Doc	6
Text	6
Label + Text	4
Label	2
Doc	2
Label + Doc	1

Table 4.5: Attributes by which a topic is defined. Topics are defined by a heterogeneous combinations of attributes. In fact, all seven combinations of *Label*, *Text*, and *Doc* are observed among the 28 topics.

concepts given by y. I find that experts on average share 3.482 bits or 11.2 ($= 2^{3.482}$) matching topics; this is consistent with my earlier findings of 12 common topics during the annotation process.

Replacing values of the $P(x, y)$ with the three constituent predictors $LabelSim$, $TextSim$ and $DocSim$, I can now estimate the amount of shared topics if experts are allowed to communicate their concepts using only labels, textual descriptions, or exemplary documents. The average mutual information for the three attributes are 7.7, 8.2, and 8.1 topics respectively. While textual descriptions convey slightly more information, the spread among the three values is quite small, suggesting that all three contribute to topical differentiation.

Hierarchy and Basic Level Categories

Previous psychology work suggests that humans organize categories hierarchically and that categories are first created at a basic level before more general and more specific categories emerge. Examining the seven pairs of overlapping topics, I find two cases of vertical organization: *Treemap* and *Trees*; and *Parallel Coordinates* and *Multi Dimensional*. Both are examples of a specific technique within a more general class of problems. The other five pairs of overlapping topics, however, are generally

not considered to be hierarchically organized based on my domain knowledge.

Further examining the full set of expert responses (i.e., ones that are given by fewer than three experts), I do observe more general and more specific topics. Two experts provided "Visualization Techniques" and "Applications" as their responses. Within techniques, the experts identified graphs, trees, and multidimensional visualization. I also observe specific categories, such as "Color" that could be a subtopic of "Perception" and "Social Network" as a subtopic of "Network." Due to the small number of hierarchal topic pairs, I am unable to define reliable measures to detect them automatically.

One participant felt so strongly about hierarchical organization that she initiated a follow-up email message specifying the five overarching categories ("paper types", "data types", "techniques", "methodologies", and "applications") under which she would group her responses.

I also observe two additional types of categories not hierarchically organized. Two experts generated exclusive (instead of inclusive) concepts "Hierarchies/Non-treemap" and another topic that groups together "other application domains". One expert generated a horizontally organized topic "Specific Techniques" that groups together the specific subtopics such as treemaps and parallel coordinates.

While informal discussions and anecdotal feedback suggest that human categorizations are hierarchical organized, the most coherent topics in the survey appear to be at a single level of organization. I hypothesize that these topics may serve as basic level categories for InfoVis researchers, and that super- and sub-ordinate topics might emerge more strongly if I am to collect more responses.

In summary, based on 202 topic responses from ten experts, I resolve matching concepts to identify a final set of 28 coherent InfoVis topics given by at least three of the experts. These topics are defined by a combination of attributes. Though some hierarchy is present, overall these topics do not exhibit a strong hierarchical organization. In the next section, I examine how well statistical topic models — and its abstraction of representing topical concepts as a bag of words — can capture this set of concepts.

4.4 An Analysis of Word-Based Topic Models

A common computational approach to organizing documents is via word-based analysis. Representing each document as a *multinomial probability distribution of words*, a model estimates the similarities (or distances) among the documents in order to infer clustering or high-level groupings. As demonstrated in the previous section, textual descriptions are only one part of a larger set of attributes experts use to define a topic. Here I establish a theoretical upper bound on the capability of word-based analysis in capturing expert-generated topics. I then evaluate the performance of topic models trained using LDA, a popular statistical topic modeling technique.

4.4.1 Four Encoding Schemes

I calculate the proportion of the 28 expert-generated InfoVis topics that can be encoded (and decoded) using various word-based representations. I examine four encoding schemes. A minimum criteria for a model to analyze topical relationships is that the topics must have distinguishable representations under their encoding. A model has no analytical power if distinct concepts appear the same to the model.

Expert-Crafted Textual Descriptions

For my first scheme, I used the set of 533 terms provided by the experts to encode the content of the InfoVis topics. Every topic was represented as a probability distribution over these 533 terms, in proportion to how often the responses that made up the topic were tagged with one of these terms. Under this representation, the vocabulary of the model was unrestricted. During the survey, the experts were free to use any word or phrase to the describe the topics. The terms needed not appear anywhere in the documents, and there were no restrictions on the relationships between the terms.

I constructed a joint probability distribution $P(x_i, y_i)$ where x_i was the ith InfoVis topic and y_j was the jth InfoVis topic. $\text{Freq}_k(w)$ was the number of times the term

w is assigned to responses belong to the kth InfoVis topic.

$$P(x_i, y_j) \propto \sum_{w \in \text{Vocab}} \text{Freq}_i(w) \times \text{Freq}_j(w)$$

I then computed the mutual information between the marginal distributions $P(x)$ and $P(y)$ to measure the amount of information transmitted from an InfoVis topic x, encoded via the expert-crated textual descriptions that is then decoded as an InfoVis topic y. The resulting mutual information was 4.138 bits or 17.1 ($= 2^{4.138}$) topics, representing 61% of coherent concepts.

Optimal Representation Based on Abstracts

For my second scheme, I constructed a model that was aware of the expert-generated topics and exemplary documents associated with each. The model was also provided with a list of the 533 most distinctive words drawn from document abstracts that maximally distinguished the topics. The model, however, must assign a probability distribution to each of the 28 topics based on word co-occurrence in the title and abstract of the documents to represent the topics.

I constructed a joint probability distribution $P(x_i, y_i)$, as defined below, in which $\text{AbstrFreq}_k(w)$ is the number of times a word w appears in the abstracts of documents belonging to topic x. The resulting mutual information between $P(x)$ and $P(y)$ was 4.033 bits or 16.3 ($= 2^{4.033}$) topics, representing 58% of coherent concepts.

$$P(x_i, y_j) \propto \sum_{w \in \text{Vocab}} \text{AbstrFreq}_i(w) \times \text{AbstrFreq}_j(w)$$

Optimal Representation Based on Full Text

For my third scheme, the model was provided with similar information as above, except that the body text of papers were used rather than the abstracts. The model must assign a probability distribution to each of the 28 topics based on word co-occurrence in the full text of the documents, extracted from the (typically eight-page) full paper. The resulting mutual information was 3.876 bits or 15.0 ($= 2^{3.876}$) topics,

covering 54% of coherent concepts.

Latent Dirichlet Allocation Topic Models

Finally, I built latent Dirichlet allocation (LDA) topic models [124] using 225 sets of parameters. I performed a grid search over 9 values of k (between 5 and 50 numbers of topics); 5 values of α (topic smoothing hyperparameter = 0.0025, 0.005, 0.01, 0.02, 0.04); and 5 values of β (term smoothing hyperparameter = 0.0025, 0.005, 0.01, 0.02, 0.04). I then selected the model that maximized the mutual information score. These LDA models were provided with list of all documents (i.e., title and abstract) in the InfoVis corpus. The models, however, were completely unaware of the human-generated concepts, and must discover k latent topics from the corpus in an unsupervised manner where k ranged between 5 and 50.

I then constructed a joint probability distribution $P(x_i, y_j)$ where x_i was the ith LDA latent topic and y_j was the jth expert-generated InfoVis topic. $\bar{P}_i(w)$ was the probability distribution for word w for the ith latent topic. The probability was summed over all words that appear in all InfoVis abstracts.

$$P(x_i, y_j) \propto \sum_{w \in \text{Vocab}} \bar{P}_i(w) \times \text{AbstrFreq}_j(w)$$

Latent topics from the highest quality LDA topic model ($n = 50$, $\alpha = 0.0025$, $\beta = 0.02$) shared 11.6 ($= 2^{3.538}$) or 41% of the expert concepts.

The LDA model produced several redundant topics. For example, it generated four latent topics corresponding to each of the *Graph* and *Network* concepts from the experts. The pairs of topics *Cognition* and *Theory*; and *Multi-Dimensional* and *Parallel Coordinates* were not separated by the model. Two notable omissions were the *Perception* and *Animation* concepts which exhibited coherent textual descriptions in the survey but were missing from the LDA model. Another prominent issue was the lack of a recognizable *Collaboration* topic which emerged as a well-defined concept in the survey data, but was merged with *BioVis* by LDA. I also observed that 22 of the generated topics were junk topics [106] that did not usefully help organize InfoVis research areas.

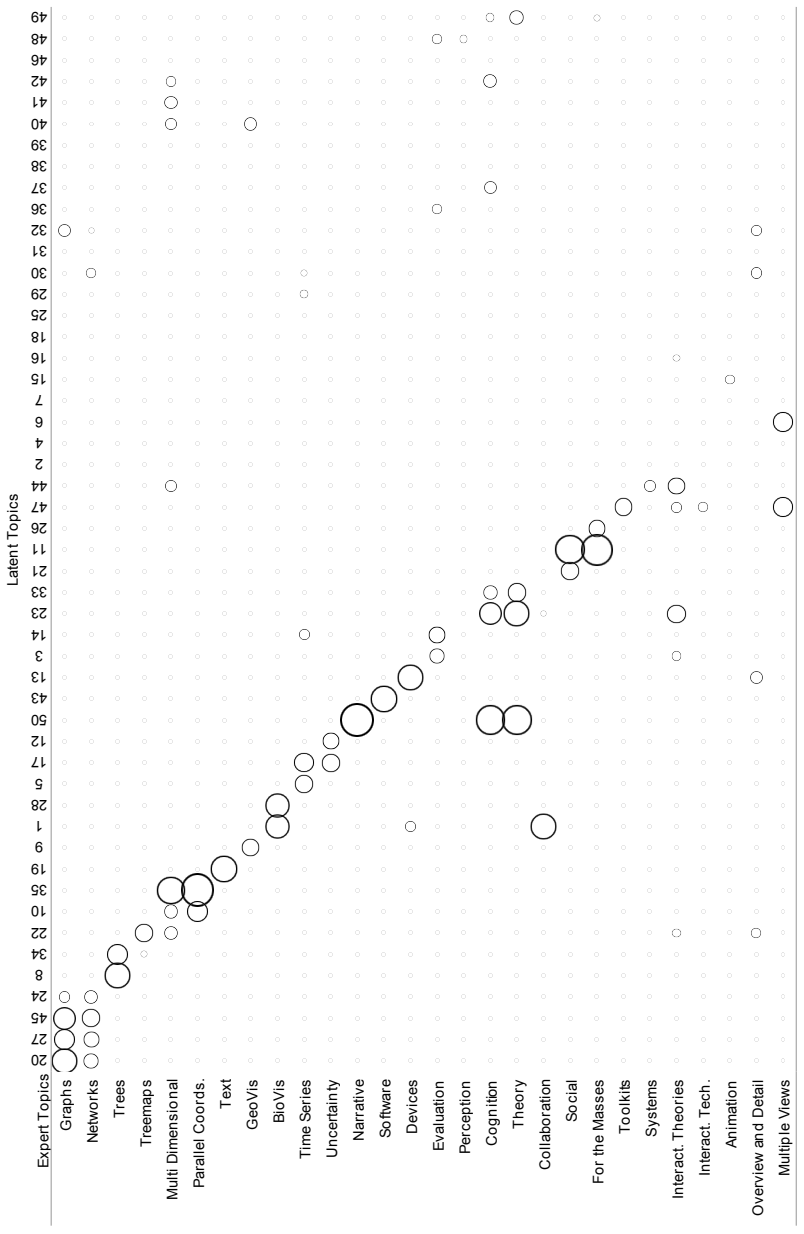

Figure 4.6: Correspondence between the highest quality LDA topic model ($n = 50$, $\alpha = 0.0025$, $\beta = 0.02$) and the set of expert-generated InfoVis topics. LDA generated multiple redundant topics (e.g., four latent topics corresponding to experts' concepts of *Graphs* and *Networks*). Two notable omissions are the exerts' *Perception* and *Animation* topics, which exhibit coherent textual descriptions in the survey but are missing from the LDA model. Another prominent issue is the lack of a recognizable *Collaboration* topic, which emerged as a well defined concept in the survey data, but is merged with *BioVis* by LDA. I note that 22 of the generated topics are "junk" topics that do not usefully help organize InfoVis research areas.

4.4.2 Discussions

At least for the domain of information visualization, these results indicate that topical categorization is multi-faceted, such that experts' descriptive terms alone can at best discern only 61% of the topics. I find that even an optimal word-based approach based on document text can at best discern 58% of expert topics, with abstract text being more informative than the larger collection of body text. I also find that latent Dirichlet allocation (LDA), a popular topic modeling algorithm, performs well below this idealized scheme, recovering only 41% of the expert topics.

While the above results await corroboration from analyses of other textual domains, they suggest that there may exist a theoretical upper bound on the quality of word-based analysis which, to the best of my knowledge, has not been established. Moreover, there may be great utility in incorporating additional data types (e.g., citations and metadata) or using semi-supervised approaches that incorporate human expertise [34, 128].

4.5 Model Diagnostics via Topical Alignment

In the previous section, by comparing LDA model output directly to expert organization, I obtain a detailed domain-specific view on how latent topics match up with the InfoVis concepts. Figure 4.6 allows us to identify high quality topics that uniquely correspond to a domain concept. The chart can also reveal domain-aware junk topics (i.e., those deemed irrelevant to the domain) and topics that fuse distinct concepts into one. Moreover, the chart allows us to determine model quality from the perspective of the experts' information organization, we identify known concepts that are missing from the model as well as those with repeated representations.

To enable large-scale assessment of topical relevance, I present a method for automatically aligning latent topics with reference concepts. At the heart of my method is the calculation of *matching likelihoods* for topic-concept pairs: the probability that a human judge will consider a latent topic and a reference concept to be equivalent. Based on human-subjects data, I examine how well various similarity measures

predict topic matches and describe how I transform similarity scores into matching likelihoods. I introduce a method to account for correspondences that occur due to random chance, to improve robustness when making a large number of comparisons. I also introduce the *correspondence chart* which visualizes the alignment between latent topics and reference concepts.

4.5.1 Correspondence Chart and Misalignments

The correspondence chart, as shown in Figure 4.7, is an $n \times m$ matrix of all possible pairings among n reference concepts and m latent topics. Each concept or topic is a multinomial distribution over words. I treat each entry $p_{s,t}$ as an independent Bernoulli random variable representing the matching likelihood that a user examining the word distributions associated with concept s and topic t would respond that the two are equivalent.

I consider a correspondence optimal when every latent topic maps one-to-one to a reference concept. Deviations from an optimal arrangement lead to four types of misalignment, as shown in Figure 4.8. I treat entries $\{p_{i,t}\}_{i=1}^{n}$ corresponding to topic t as a Bernoulli-like process: a series of independent events that can take on different probabilities. In this framework, $\dot{P}_t(k)$ is the likelihood that a user responds with exactly k matches after comparing topic t to all n reference concepts. Similarly, $\ddot{P}_s(k)$ is the likelihood of observing exactly k positive outcomes after comparing concept s to all m latent topics. The *junk* score for topic t is the probability $\dot{P}_t(0)$; the topic has no matching concept. The *fused* score for topic t is the likelihood $\sum_{k=2}^{m} \dot{P}_t(k)$; the topic matches two or more concepts. Similarly, the *missing* score for concept s is $\ddot{P}_s(0)$, and the *repeated* score is $\sum_{k=2}^{n} \ddot{P}_s(k)$.

4.5.2 Human Judgment of Topic Matches

I conducted a study to acquire data on when topics (probability distributions over terms) are considered matching by people. I trained two LDA topic models on a corpus of information visualization publications and sampled pairs of topics, one from each model. The texts were chosen to be consistent with the expert-generated

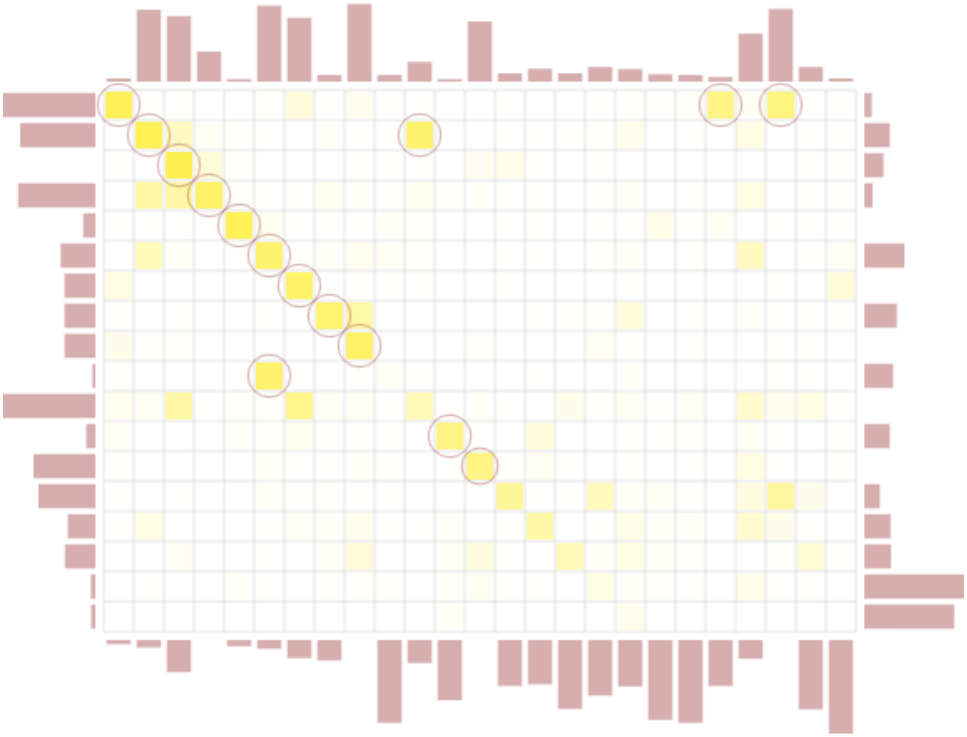

Figure 4.7: Correspondence between a set of latent topics (columns) and a set of reference concepts (rows). Shading represents the likelihood that a latent topic matches a reference concept, and circles show if the likelihood exceeds random chance. On the right, I mark reference concepts that are *missing*, on the left *repeated*, on the bottom model topics that are *junk* and on the top *fused*. The 5th topic *resolves* excellently to the 5th concept.

concepts that I collected (details in Section 4.3). Earlier analysis suggested that the corpus contained about 28 domain concepts, and thus I trained the two models with 40 and 50 latent topics using priors $\alpha = 0.01$ and $\beta = 0.01$.

I presented study subjects with topical pairs, one at a time in a webpage as shown in Figure 4.9. Each topic was displayed as a list of words, sorted by frequency, where the height of each word was scaled proportional to its frequency in the topic's distribution. I asked the subjects whether the two topics match ("represent the same meaningful concept"), partially match, or do not match ("represent different concepts or meaningless concepts"). The study was conducted on Amazon Mechanical Turk.

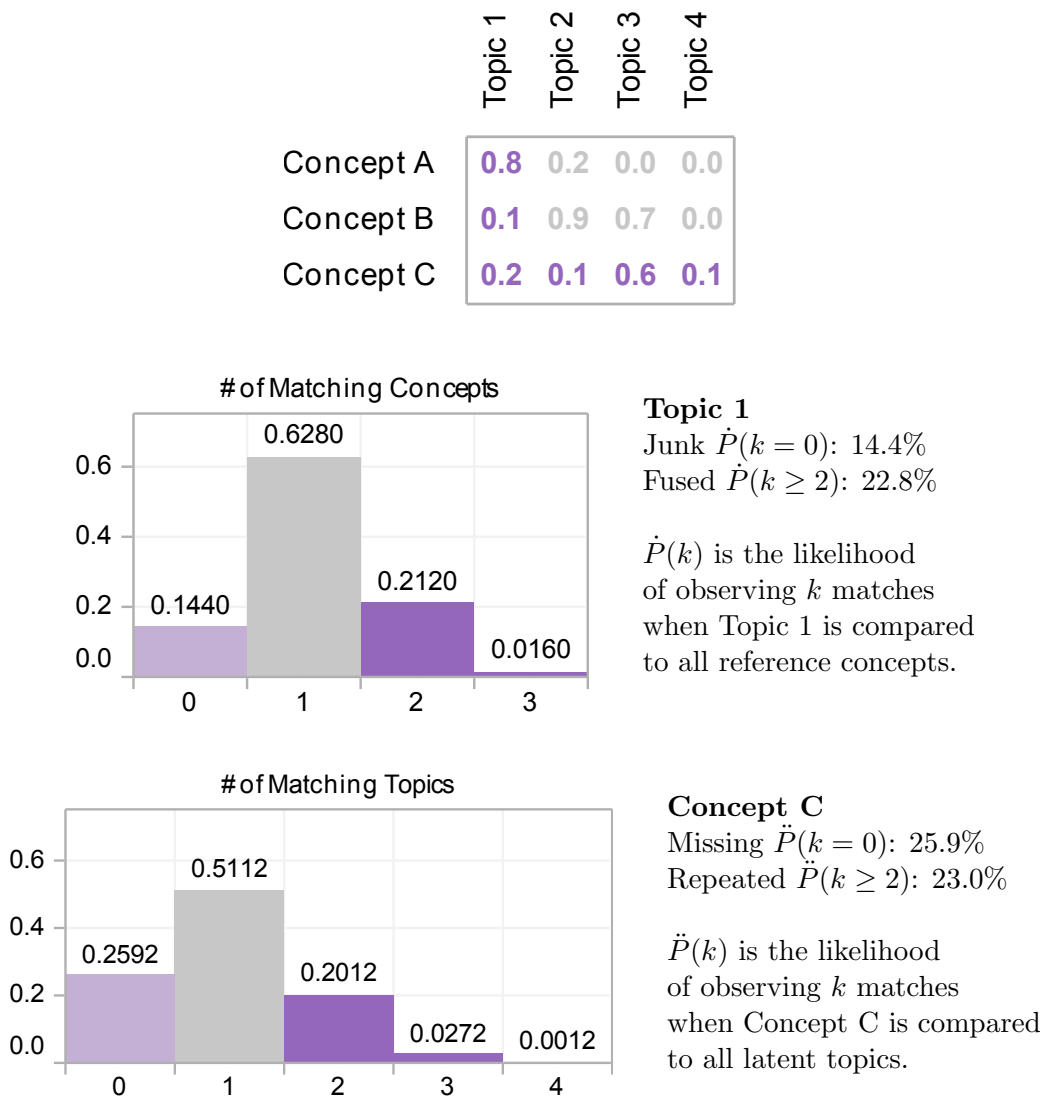

Figure 4.8: Correspondence Chart Construction. In a correspondence chart, each entry $p_{s,t}$ represents the probability that a user considers the word distributions associated with concept s and topic t as equivalent. Misalignment scores measure how much topical alignment deviates from an optimal one-to-one correspondence. Comparing a topic to all concepts, *junk* and *fused* scores measure how likely the topic matches exactly zero, or more than one reference concept. *Missing* and *repeated* scores measure how likely a concept matches exactly zero, or more than one latent topic.

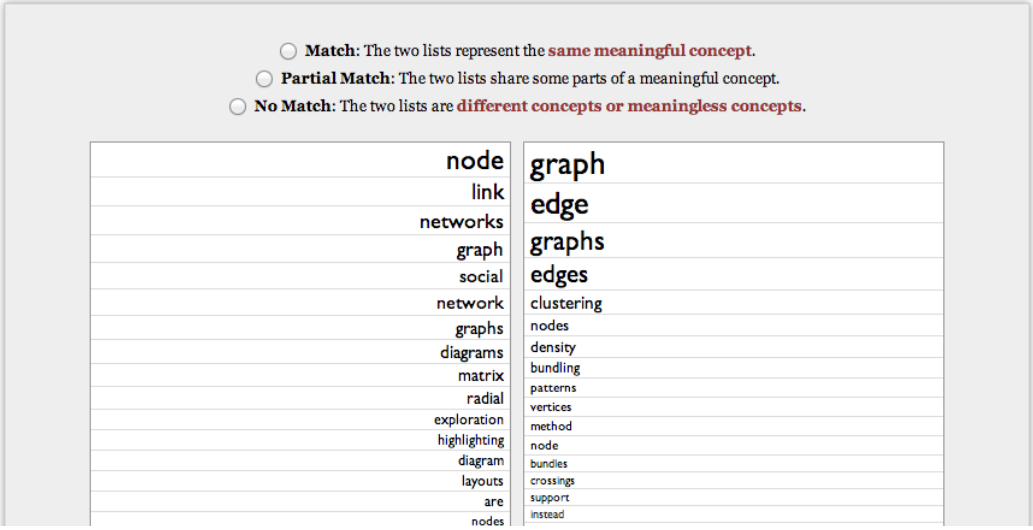

Figure 4.9: User interface for the study on human judgement of topical matches.

I included five topical pairs in each task, posted 200 tasks with a US$0.25 reward per task in December 2012, and received 1,000 ratings for 167 topical pairs.

4.5.3 Evaluating Topical Similarity Measures

I evaluated how well similarity measures can predict human judgment in terms of precision and recall. For each topical pair, I assigned it a rating of $\{1, 0.5, 0\}$ for each {match, partial match, no match} response and consider a pair as matching if it has an average rating above 0.5. I computed the similarity between topics using the four measures listed in Table 4.6. Cosine, rank, and KL-divergence represent three common approaches for measuring topical similarity. I also introduced a *rescaled dot product* to improve upon cosine.

Precision-recall scores in Figure 4.10 compare user-identified matches to the ordering of topical pairs induced by the similarity measures. The rescaled dot product achieves the highest scores for AUC, F1, F0.5, and F2 measures. I find that KL-divergence does a poor job of predicting human judgment; topical pairs ranked in the 90th percentile (among the 10% of most divergent pairs) still contain matches.

Spearman's correlation is concentrated in a narrow range $(-0.04, 0.16)$ for 96% of the data points. I observe that L_2 normalization in the cosine calculation is largely ineffective when applied to (L_1 normalized) probability distributions. Instead, given two word distributions I rescale their dot product to the range of minimum and maximum possible similarities, and find that this outperforms the other measures.

4.5.4 Mapping Similarity Scores to Likelihoods

While the rescaled dot product is predictive of human judgment, the actual similarity values deviate from the definition of matching likelihood. Figure 4.11 plots precision against the similarity score at which that precision is achieved. By definition, topical pairs ranked above a precision of 0.5 are considered matching by human judges over 50% of the time. For the rescaled dot product, this threshold occurs at 0.1485 instead of the desired value of 0.5.

Linear transformation in log-ratio likelihood space performs well for correcting this deviation. I convert similarity scores and precision values to log-ratio likelihoods, and apply linear regression to determine optimal mapping coefficients (see Table 4.7). For the rescaled dot product, the transformed scores deviate from average user ratings

Cosine	$\frac{P \cdot Q}{\|P\|_2 \|Q\|_2}$
Rescaled Dot Product	$\frac{P \cdot Q - d_{\text{Min}}}{d_{\text{Max}} - d_{\text{Min}}}$ $\quad d_{\text{Max}} = \overrightarrow{P} \cdot \overrightarrow{Q}$ $\quad d_{\text{Min}} = \overrightarrow{P} \cdot \overleftarrow{Q}$
Rank	$\text{SpearmanCorrelation}(I(P), I(Q))$
KL-Divergence	$\sum_i p_i \log \frac{p_i}{q_i}$

Table 4.6: Similarity Measures. Similarity scores for two word probability distributions P and Q. Scalar x_i denotes the probability for term i in X. For the rescaled dot product, $\overrightarrow{X}$ is a vector consisting of all x_i sorted in descending value; $\overleftarrow{X}$ is a vector consisting of all x_i sorted in ascending value. For rank, $I(X)$ denote the ranks of terms in X. For KL-divergence, I treat topics from one model as reference concepts P and the others as latent topics Q.

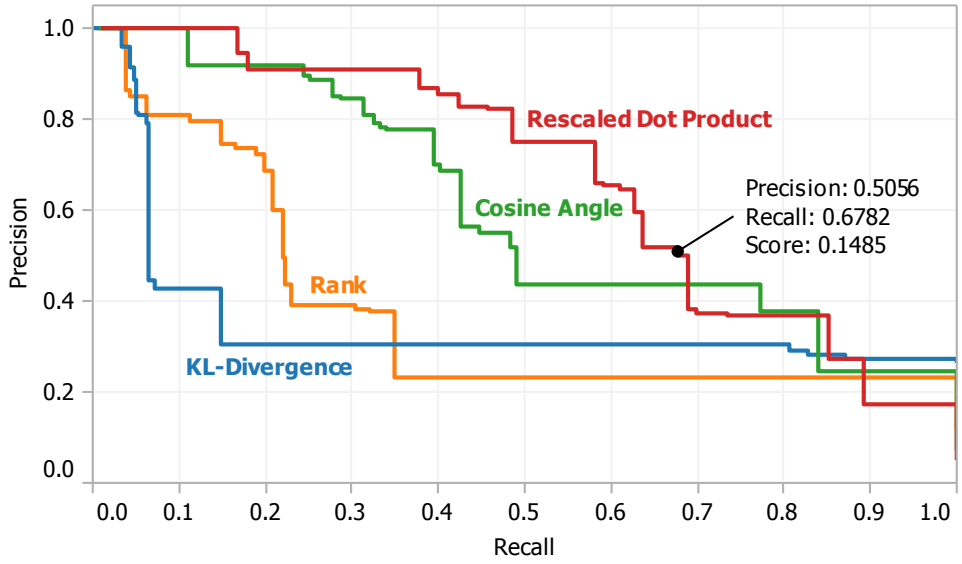

Figure 4.10: Precision and recall. Predicting human judgment of topic matches using topical similarity measures.

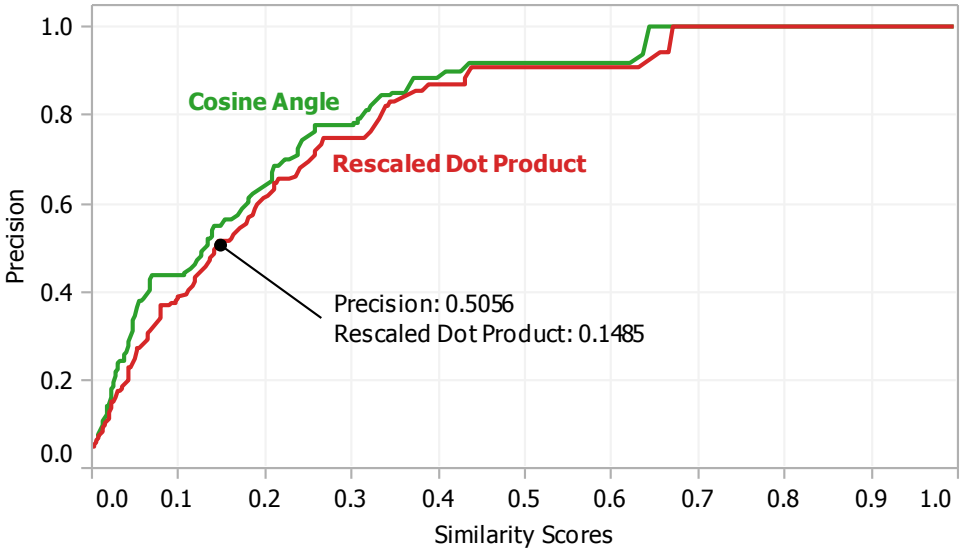

Figure 4.11: Similarity Score vs. Precision. Topical pairs with a rescaled dot product score greater than 0.148 were considered matching by human judges over 50% of the time.

by 0.0650. Transformed cosine angles deviate from user ratings by 0.1036. Provided with sets of reference concepts and latent topics, I can now populate entries of a correspondence chart using the transformed rescaled dot product scores.

Similarity Score	s
Log-Ratio Likelihood	$s' = \dfrac{\log s}{\log(1-s)}$
Linear Regression	$t' = as' + b$
Inverse of Log-Ratio Likelihood	$t = \dfrac{e^{t'}}{e^{t'}+1}$
Transformed Similarity Score	t

Table 4.7: Transformed Similarity Score. I fit similarity scores to empirically obtained precision values, based on linear regression in log-ratio likelihood space. For rescaled dot product, the coefficients are $a = 1.567088$ and $b = 2.445738$. For cosine, they are $a = 1.970030$ and $b = 4.163359$.

4.5.5 Estimating Random Chance of Matching

Matching likelihoods determined from human judgments are rarely exactly zero. As a topic model may contain hundreds of latent topics, even a small chance probability of matching can accumulate and bias misalignment scores toward a high number of repeated concepts or fused topics. To ensure the framework is robust for large-scale comparisons, I introduce a method to estimate and remove topical correspondences that can be attributed to chance.

Given a correspondence matrix, I treat it as a linear combination of two sources: a *definitive* matrix whose entries are either 0 or 1; and a *noise* matrix representing some chance probability. I assume that matching likelihoods are randomly drawn from the definitive matrix $(1-\gamma)$ of the time and from the noise matrix γ of the time, where γ is a noise factor between $[0, 1]$.

Without explicitly specifying the values of the entries in the definitive matrix, I can still construct $P^k_{\text{definitive}}$ if I know it contains k non-zero values. I compute the average row and column matching likelihoods, and create a noise matrix whose entries

equal $\hat{p}_{s,t} = 0.5 \sum_{i=1}^{n} p_{i,t}/n + 0.5 \sum_{j=1}^{m} p_{s,j}/m$. The action of sampling from the two source charts produces a corresponding $P_{\text{combined}} = P_{\text{definitive}}^{k(1-\gamma)} * P_{\text{noise}}^{\gamma}$ where $*$ is the convolution operator. A full derivation is in Appendix A.1.

I compute γ by solving the convex optimization (implementation details are in Section A.2):

$$\min_{\gamma} \text{KL}(P_{\text{definitive}}^{k(1-\gamma)} * P_{\text{noise}}^{\gamma} || P).$$

The optimal γ represents the estimated amount of matches that can be attributed to noise. I then estimate the most likely distribution of topical matches P_{denoised} without the chance matches, by solving the following constrained optimization:

$$\min_{P_{\text{denoised}}} \text{KL}(P_{\text{denoised}} * P_{\text{noise}}^{\gamma} || P)$$

subject to P_{denoised} being a proper probability distribution whose entries sum to 1 and are in the range $[0, 1]$. Implementation details are in Appendices A.2 and A.3.

I apply the above process to each row and column in the correspondence matrix, to obtain $\dot{P}_{\text{denoised}}$ and $\ddot{P}_{\text{denoised}}$ from which I estimate topical misalignment scores as described previously.

4.6 Applications of Model Diagnostic Framework

I experimented with an exploratory process of topic model construction, in which a user specifies reference concepts a priori and uses alignment scores to analyze the parameter space of models. Talley et al. [152] found that the number of latent topics affects both concept resolution and the number of poor quality topics. They arrived at this conclusion only after building a large number of models and performing an extensive manual review. In contrast, my framework allows users to map a large number of models onto predefined concepts, and immediately inspect model qualities. As shown in Figure 4.12, my misalignment measures indicate that the maximum number of resolved topics peaks at $N = 18$. While the ratio of fused topics dips at $N = 20$, the proportion of fused topics increases again for $N \geq 30$. Trends in Figure

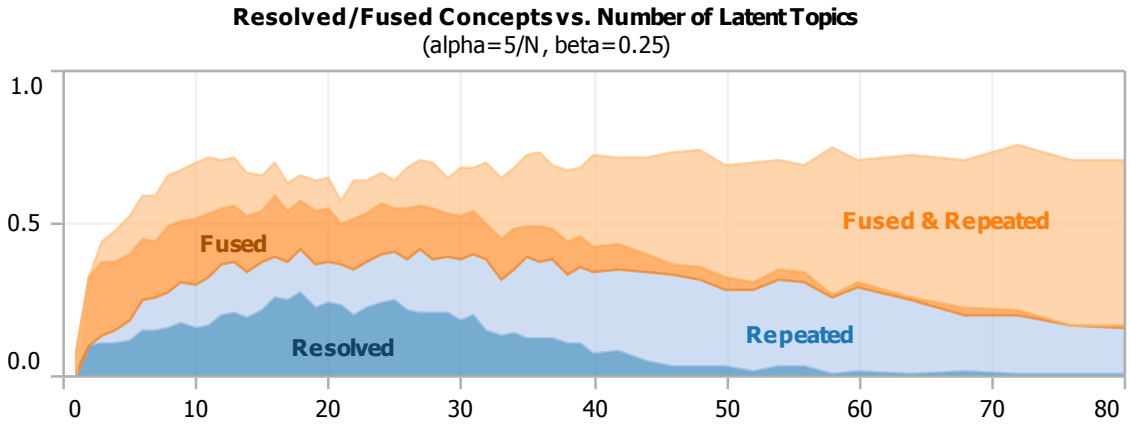

Figure 4.12: Alignment of LDA models for $\alpha = 5/N$, $\beta = 0.25$, $N \in [1, 80]$. The y-axis shows the percentage of reference concepts that have a single matching topic (*Resolved*), multiple matching topics (*Repeated*) or are subsumed by one (*Fused*) or multiple fused topics (*Fused & Repeated*). These models uncover up to 75% of the reference concepts, but coverage increases only marginally for $N \geq 10$. Further increases in N result in duplicate latent topics that correspond to concepts already uncovered. For $N \geq 40$, the models produce an increasing number of latent topics that fuse multiple concepts, and a corresponding reduction in resolved concepts.

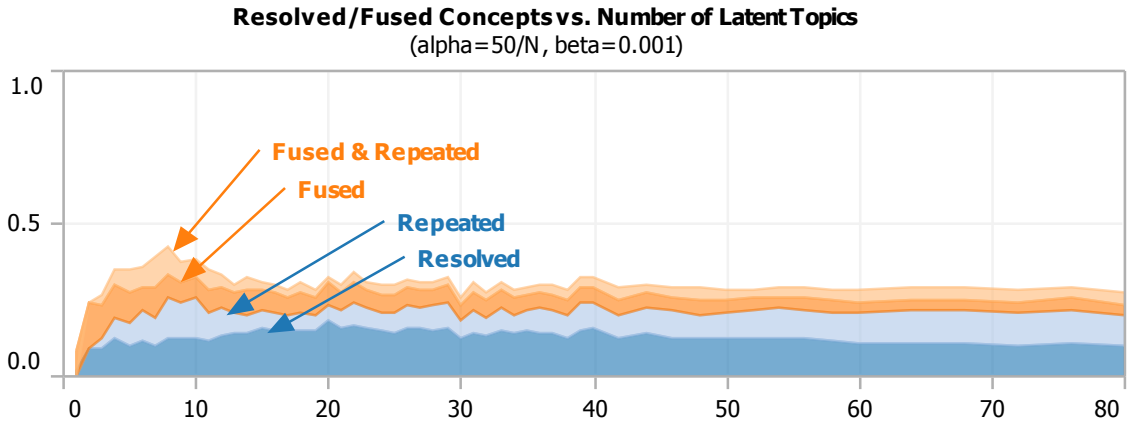

Figure 4.13: Alignment for $\alpha = 50/N$, $\beta = 0.001$, $N \in [1, 80]$. This series of LDA models uncovers up to 40% of the reference concepts. Coverage peaks at N=8. The proportion of resolved and fused topics remains stable for $N \geq 15$; increasing N produces only more junk topics.

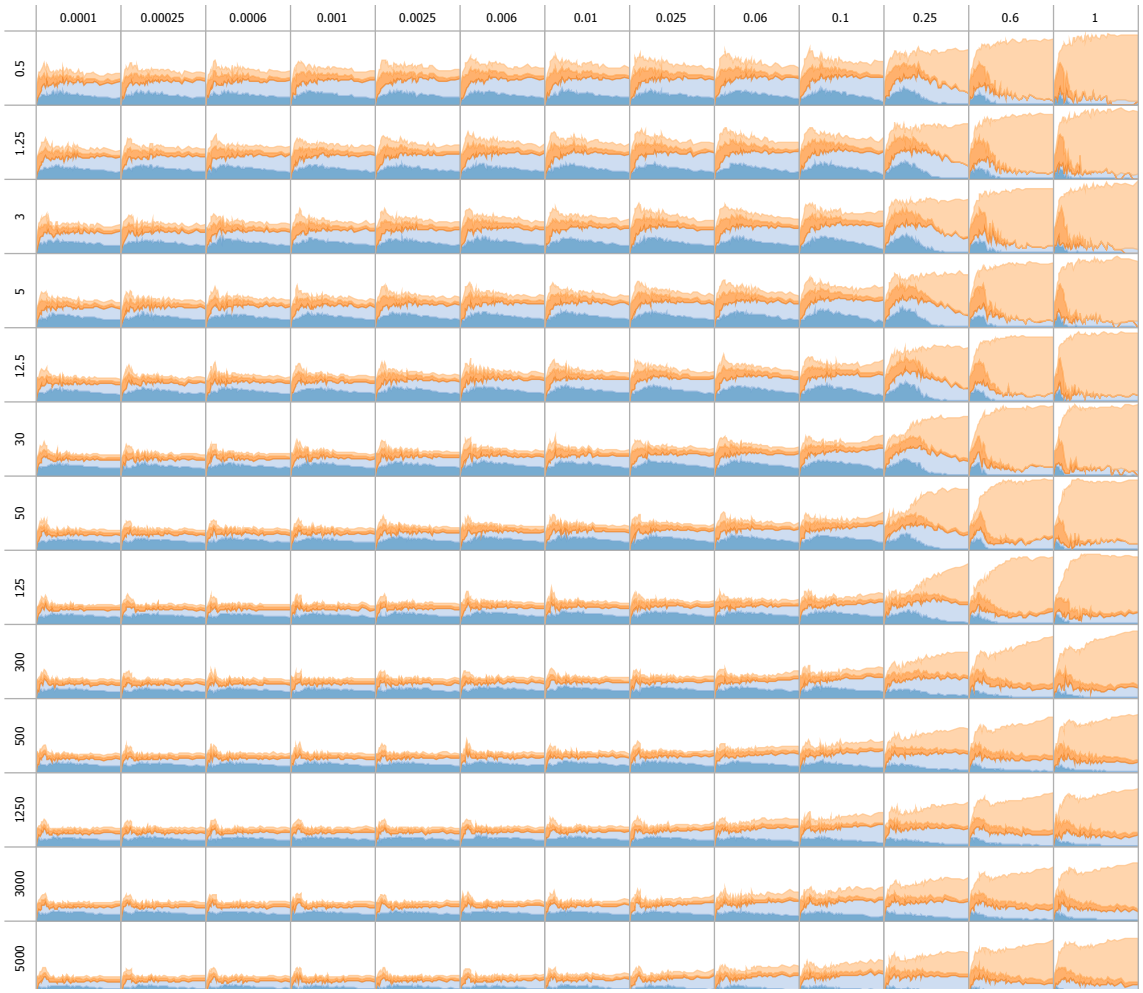

Figure 4.14: Alignment of LDA models for $\alpha \in [0.5/N, 5000/N]$, $\beta \in [0.0001, 1]$, and $N \in [1, 80]$, over a grid of α-values (vertical) and β-values (horizontal). We observe a qualitative shift in alignment at β=0.25. For β>0.25, the models generate fused topics that uncover but do not fully resolve a majority of the reference concepts as N increases. For β<0.25, the proportion of resolved and fused topics remain stable regardless of N. For β=0.25, the models resolve the most concepts at α=5 and N=18. Overall, decreasing β or increasing α leads to a decrease in coverage.

4.13 suggest that, for a different hyperparameter setting, increasing N produces only more junk topics.

In Figure 4.14, I extend the space of models to over 10,000 parameter settings, and observe additional patterns. We find a qualitative change in topic composition

around $\beta = 0.25$. For $\beta > 0.25$, the models generate fused topics that uncover but do not fully resolve a majority of the reference concepts as N increases. For $\beta < 0.25$, the proportion of resolved and fused topics remain stable regardless of N.

4.7 Summary

In this chapter, I examined how human-centered approaches and interactive visualizations can help improve the topic modeling process. I began by collecting and analyzing expert topical organization of document collections. I contributed a survey method as well as a method for synthesizing participant responses and a corresponding method for validating the combined results. I assessed how well various topic models captured the expert concepts in terms of shared mutual information.

I devised the correspondence chart, a visualization showing how a set of statistically extracted topics aligned with a set of known reference concepts. The chart provided *diagnostic feedback* that can help analysts examine and compare multiple models. To increase the utilization and *reusability* of acquired domain expertise, I introduced a framework to determine the correspondence between any number of topics models with a common set of reference concepts. I developed a matching likelihood measure to capture human judgment of topical similarity. I devised a method to improve the numerical robustness of my approach, and demonstrated its effectiveness through a use case where I identified suitable modeling settings from over 10,000 parameter choices.

Chapter 5

Descriptive Phrases for Text Summarization

Descriptive phrases aid the exploration of text collections by communicating salient aspects of documents. For statistical topic models, labeling can improve the interpretability of the discovered concepts. Keyphrases are also frequently used to create effective visualizations of text. While prior work in information visualization has proposed a variety of ways of presenting keyphrases, less attention has been paid to selecting the best descriptive terms.

In this chapter, I demonstrate how a human-centered design process leads to improved keyphrase extraction algorithms and enables novel interactive visualizations for summarizing text. Based a study of 69 graduate students describing a corpus of dissertation abstracts, I analyze the statistical and linguistic properties most predictive of high-quality keyphrases chosen by human judges. I systematically assess the contribution of potential features within statistical models of keyphrase quality. I then introduce a method for grouping similar terms and varying the specificity of displayed terms so that keyphrases can be dynamically chosen based on the available screen space or the current context of interaction. Precision-recall measures find that my technique generates keyphrases matching those selected by human judges and scores comparably to existing keyphrase extraction algorithms. Crowdsourced ratings of tag cloud visualizations rank my approach above other automatic techniques.

5.1 Chapter Outline

Document collections, from academic publications to blog posts, provide rich sources of information. People explore these collections to understand their contents, uncover thematic patterns, or find documents matching an information need.

Keywords (or *keyphrases*) aid exploration by providing summary information intended to communicate salient aspects of one or more documents. When applying statistical topic modeling to analyses, labeling can aid interpretation especially when communicating with or presenting the uncovered topical concepts to users unfamiliar with the dataset [102]. Keyphrase selection is also critical to effective visualization and interaction, including automatically labeling documents, clusters, or themes [64, 68]; choosing salient terms for tag clouds or other text visualization techniques [42, 162, 164]; or summarizing text to support small display devices [22, 23, 176]. While terms hand-selected by people are considered the gold standard, manually assigning keyphrases to thousands of documents simply does not scale.

To aid interpretation, keyphrase extraction algorithms select descriptive phrases from text. A common method is bag-of-words frequency statistics [88, 108, 131, 134, 140]. However, such measures may not be suitable for short texts [12] and typically return single words, rather than more meaningful longer phrases [158]. While others have proposed methods for extracting longer phrases [6, 51, 53, 74, 80, 101], researchers have yet to systematically evaluate the contribution of individual features predictive of keyphrase quality and often rely on assumptions — such as the presence of a reference corpus or knowledge of document structure — that are not universally applicable.

I characterize the statistical and linguistic properties of human-generated keyphrases in Section 5.2. My analysis is based on 5,611 responses from 69 students describing Ph.D. dissertation abstracts. I use the results to develop a two-stage method for automatic keyphrase extraction. I first apply a regression model to score candidate keyphrases independently (Section 5.3). I then group similar terms to reduce redundancy and control the specificity of selected phrases (Section 5.4). Throughout the analysis and modeling process, I investigate the following concerns.

Reference Corpora. Human-computer interaction (HCI) researchers work with text from various sources including data whose domain is unspecified or in which a domain-specific reference corpus is unavailable. I examine several frequency statistics and assess the trade-offs of selecting keyphrases with and without a reference corpus. While models trained on a specific domain can generate higher-quality phrases, I find that models incorporating language-level statistics in lieu of a domain-specific reference corpus produce competitive results.

Document Diversity. Interactive systems may need to show keyphrases for a collection of documents. I compare descriptions of single documents and of multiple documents with varying levels of topical diversity. I find that increasing the size or diversity of a collection reduces the length and specificity of selected phrases.

Feature Complexity. Many existing tools select keyphrases solely using raw term counts or tf.idf scores [140], while recent work [42, 108] advocates more advanced measures, such as G^2 statistics [51, 131]. I find that raw counts or tf.idf alone provide poor summaries but that a simple combination of raw counts and a term's language-level commonness matches the improved accuracy of more sophisticated statistics. I also examine the impact of features such as grammar and position information. For example, I find that part-of-speech tagging provides significant benefits, over which more costly statistical parsing provides little improvement.

Term Similarity and Specificity. Multi-word phrases identified by an extraction algorithm may contain overlapping terms or reference the same entity (person, location, organization, etc). I present a method for grouping related terms and reducing redundancy. The resulting organization enables users to vary the specificity of displayed terms and allows applications to dynamically select terms in response to available screen space. For example, a keyphrase label might grow longer and more specific through semantic zooming.

I assess the resulting extraction approach by comparing automatically and manually selected phrases and via crowdsourced ratings. I find that the precision and recall of candidate keyphrases chosen by my model can match that of phrases hand-selected by human readers. I also apply my approach to tag clouds as an example of real-world presentation of keyphrases. I asked human judges to rate the quality of tag clouds

using phrases selected by my technique and unigrams selected using G^2. Raters prefer the tag clouds generated by my method and identify other factors such as layout and prominent errors that affect judgments of keyphrase quality. Finally, I conclude the paper by discussing the implications of this research for human-computer interaction, information visualization, and natural language processing.

5.2 Characterizing Human-Generated Keyphrases

To better understand how people choose descriptive phrases, I compiled a corpus of phrases manually chosen by expert and non-expert readers. I analyzed this corpus to assess how various statistical and linguistic features contribute to keyphrase quality.

5.2.1 User Study Design

I asked graduate students to provide descriptive phrases for a collection of Ph.D. dissertation abstracts. I selected 144 documents from a corpus of 9,068 Ph.D. dissertations published at Stanford University from 1993 to 2008. These abstracts constitute a meaningful and diverse corpus well suited to the interests of the study participants. To ensure coverage over a variety of disciplines, I selected abstracts each from the following six departments: Computer Science, Mechanical Engineering, Chemistry, Biology, Education, and History. I recruited graduate students from two universities via student email lists. Students came from departments matching the topic areas of selected abstracts.

5.2.2 Study Protocol

I selected 24 dissertations (as eight groups of three documents) from each of the six departments in the following manner. I randomly selected eight faculty members from among all faculty who have graduated at least ten Ph.D. students. For four of the faculty members, I selected the three most topically diverse dissertations. For the other four members, I selected the three most topically similar dissertations.

Subjects participated in the study over the Internet. They were presented with a series of webpages and asked to read and summarize text. Subjects received three groups of documents in sequence (nine in total); they were required to complete one group of documents before moving on to the next group. For each group of documents, subjects first summarized three individual documents in a sequence of three webpages and then summarized the three as a whole on a fourth page. Participants were instructed to summarize the content using five or more keyphrases, using any vocabulary they deemed appropriate. Subject were not constrained to only words from the documents. They would then repeat this process for two more groups. The document groups were randomly selected such that they varied between familiar and unfamiliar topics.

I received 69 completed studies, comprising a total of 5,611 free-form responses: 4,399 keyphrases describing single documents and 1,212 keyphrases describing multiple documents. Note that while I use the terminology keyphrase in this chapter for brevity, the longer description "keywords and keyphrases" was used throughout the study to avoid biasing responses. The online study was titled and publicized as an investigation of "keyword usage."

5.2.3 Independent Factors

I varied the following three independent factors in the user study.

Familiarity. I considered a subject *familiar* with a topic if they had conducted research in the same discipline as the presented text, and relied on self-reports to determine subjects' familiarity.

Document count. Participants were asked to summarize the content of either a single document or three documents as a group. In the case of multiple documents, I used three dissertations supervised by the same primary advisor.

Topic diversity. I measured the similarity between two documents using the cosine of the angle between tf.idf term vectors. My experimental setup provided sets of three documents with either low or high topical similarity.

5.2.4 Dependent Statistical and Linguistic Features

To analyze responses, I computed the following features for the documents and subject-authored keyphrases. For the remainder of this chapter, I use "term" and "phrase" interchangeably. Term length refers to the number of words in a phrase; an n-gram is a phrase consisting of n words.

Documents are the texts I showed to subjects, while *responses* are the provided summary keyphrases. I tokenize text based on the Penn Treebank standard [96] and extract all terms of up to length five. I record the position of each phrase in the document and whether a phrase occurs in the first sentence. *Stems* are the roots of words with inflectional suffixes removed. I apply light stemming [107] which removes only noun and verb inflections (such as plural s) according to a word's part of speech. Stemming allows us to group variants of a term when counting frequencies.

Term frequency (tf) is the number of times a phrase occurs in the document (*document term frequency*), in the full dissertation corpus (*corpus term frequency*), or in all English webpages (*web term frequency*) as indicated by the Google web n-gram corpus [20]. I define *term commonness* as the normalized term frequency relative to the most frequent n-gram, either in the dissertation corpus or on the web. For example, the commonness of a unigram equals $\log(tf)/\log(tf_{\text{the}})$ where tf_{the} is the frequency of "the" — the most frequent unigram. When distinctions are needed, I refer to the former as *corpus commonness* and the latter as *web commonness*.

Term position is a normalized measure of a term's location in a document. A value of 0 corresponds to the first word and 1 to the last. The *absolute first occurrence* is the minimum position of a term (cf., [101]). However, as frequent terms are more likely to appear earlier due to higher rates of occurrence, I introduce a new feature — the *relative first occurrence* — to factor out the correlation between position and frequency. Relative first occurrence (formally defined in Section 5.3.1) is the probability that a term's first occurrence is lower than that of a randomly sampled term with the same frequency. This measure makes a simplistic assumption — that term positions are uniformly distributed — but allows us to assess term position as an independent feature.

I annotate terms that are *noun phrases*, *verb phrases*, or match *technical term*

Pattern	Regular Expression
Technical term	$T = (A\|N)^+ \ (N\|C) \quad \| \quad N$
Compound technical term	$X = (A\|N)^* \ N \text{ of } (T\|C) \quad \| \quad T$

Table 5.1: Technical terms. Technical terms are defined by part-of-speech regular expressions. N is a noun, A an adjective, and C a cardinal number. I modify the definition of technical terms [78] by permitting cardinal numbers as the trailing word. Examples of technical terms include the following: *hardware*, *interactive visualization*, *performing arts*, *Windows 95*. Examples of compound technical terms include the following: *gulf of execution*, *War of 1812*.

patterns [78] (see Table 5.1). Part-of-speech information is determined using the Stanford POS Tagger [156]. I additionally determine grammatical information using the Stanford Parser [82] and annotate the corresponding words in each sentence.

5.2.5 Exploratory Analysis of Human-Generated Phrases

Using these features, I characterized the collected human-generated keyphrases in an exploratory analysis. My results confirm observations from prior work — the prevalence of multi-word phrases [158], preference for mid-frequency terms [94], and pronounced use of noun phrases [6, 47, 53, 74] — and provide additional insights, including the effects of document count and diversity.

For single documents, the number of responses varies between 5 and 16 keyphrases (see Figure 5.1). I required subjects to enter a minimum of five responses; the peak at five in Figure 5.1 suggests that subjects might respond with fewer without this requirement. However, it is unclear whether this reflects a lack of appropriate choices or a desire to minimize effort. For tasks with multiple documents, participants assigned fewer keyphrases despite the increase in the amount of text and topics. Subject familiarity with the readings did not have a discernible effect on the number of keyphrases.

Assessing the prevalence of words vs. phrases, Figure 5.2 shows that bigrams are the most common response, accounting for 43% of all free-form keyphrase responses, followed by unigrams (25%) and trigrams (19%). For multiple documents

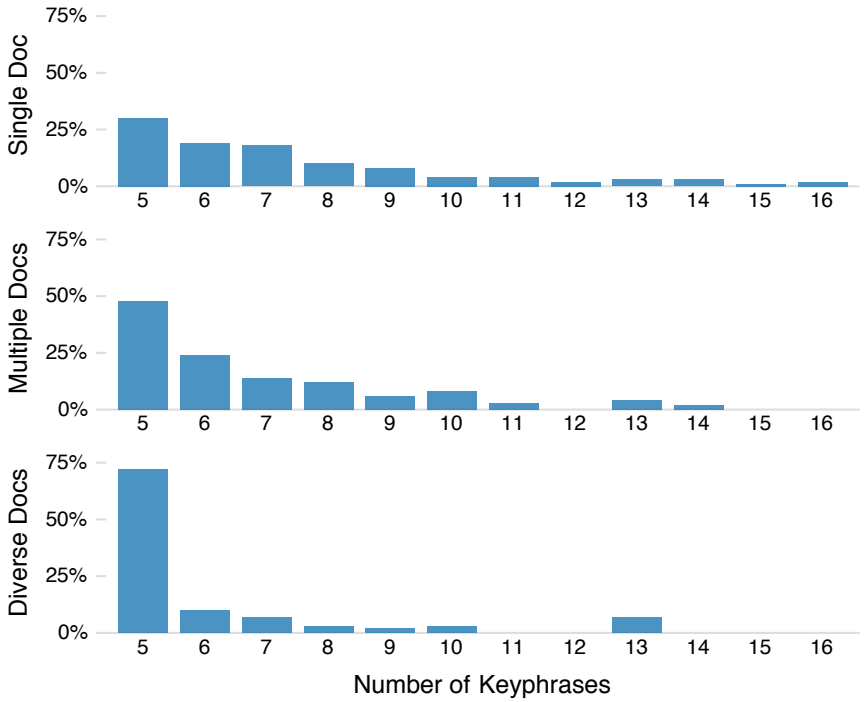

Figure 5.1: How many keyphrases do people use? Participants use fewer keyphrases to describe multiple documents or documents with diverse topics, despite the increase in the amount of text and topics.

or documents with diverse topics, I observe an increase in the use of unigrams and a corresponding decrease in the use of trigrams and longer terms. The prevalence of bigrams confirm prior work [158]. By permitting users to enter any response, my results provide additional data on the tail end of the distribution: there is minimal gain when assessing the quality of phrases longer than 5 words, which account for less than 5% of responses.

Figure 5.3 shows the distribution of responses as a function of web commonness. I observe a bell-shaped distribution centered around mid-frequency, consistent with the distribution of significant words posited by Luhn [94]. As the number of documents and topic diversity increases, the distribution shifts toward more common terms. I found similar correlations for corpus commonness.

For each user-generated keyphrase, I find matching text in the reading, and note

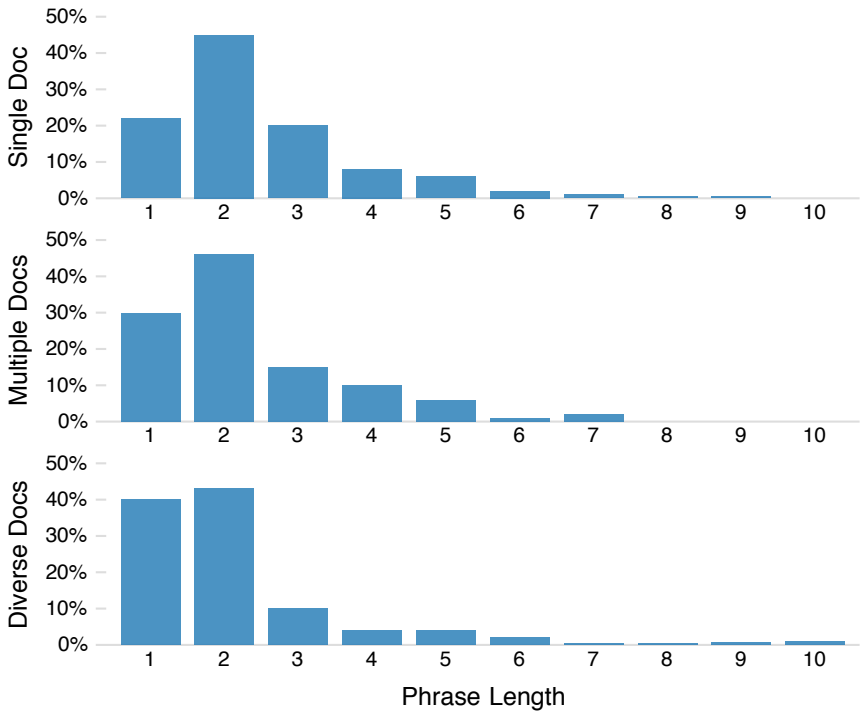

Figure 5.2: Do people use words or phrases? Bigrams are the most common. For single documents, 75% of responses contain multiple words. Unigram use increases with the number and diversity of documents.

that 65% of the responses are present in the document. Considering for the rest of this paragraph just the two thirds of keyphrases present in the readings, the associated *positional* and *grammatical* properties of this subset are summarized in Table 5.2. 22% of keyphrases occur in the first sentence, even though first sentences contain only 9% of all terms. Comparing the first occurrence of keyphrases with that of randomly sampled phrases of the same frequency, I find that keyphrases occur earlier 56% of the time — a statistically significant result ($\chi^2(1) = 88$, $p < 0.001$). Nearly two-thirds of keyphrases found in the document are part of a noun phrase (i.e., continuous subsequence fully contained in the phrase). Only 7% are part of a verb phrase, though this is still statistically significant ($\chi^2(1) = 147{,}000$, $p < 0.001$). Most strikingly, over 80% of the keyphrases are part of a technical term.

In summary, the above exploratory analysis shows that subjects primarily choose

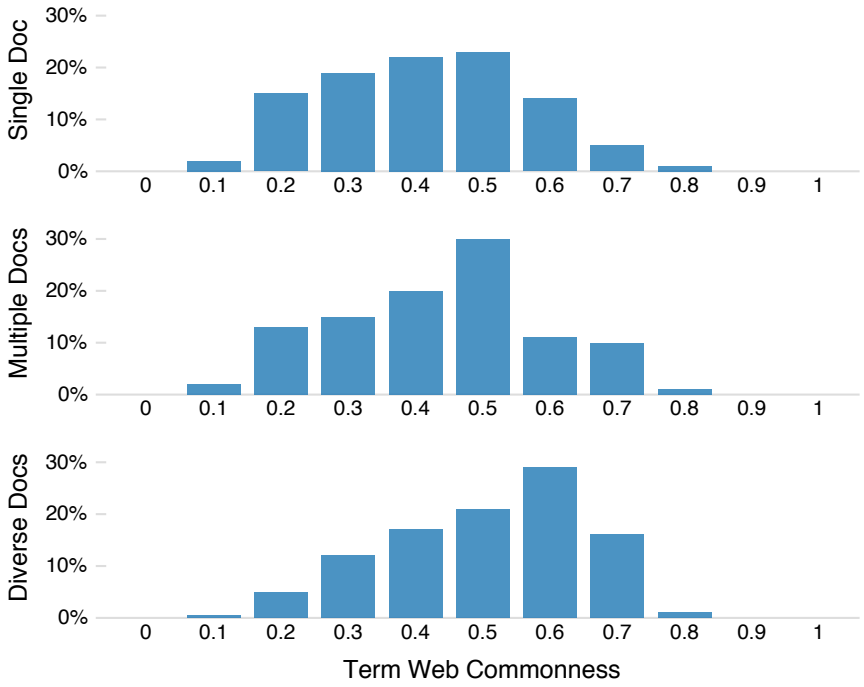

Figure 5.3: Do people use generic or specific terms? Term commonness increases with the number and diversity of documents.

multi-word phrases, prefer terms with medium commonness, and largely use phrases already present in a document. Moreover, these features shift as the number and diversity of documents increases. Keyphrase selection also correlates with term position, suggesting we should treat documents as more than just "bags of words." Finally, human-selected keyphrases show recurring grammatical patterns, indicating the utility of linguistic features.

5.3 Automatic Keyphrase Extraction

Informed by the exploratory analysis, I systematically assessed the contribution of statistical and linguistic features to keyphrase quality, resulting in a pair of regression models (one corpus-dependent, the other independent) that incorporate term

Feature	% of Keyphrases	% of All Phrases
First sentence	22.09%	8.68%
Relative first occurrence	56.28%	50.02%
Noun phrase	64.95%	13.19%
Verb phrase	7.02%	3.08%
Technical term	82.33%	8.16%
Compound tech term	85.18%	9.04%

Table 5.2: Positional and grammatical statistics. Position and grammar features of keyphrases present in a document (65% of total). Keyphrases occur earlier in a document: two-thirds are noun phrases, over four-fifths are technical terms.

frequency, commonness, position, and grammatical features. I evaluated my models in two ways. First, I compared the performance of my models with that of the human judges. Second, I compared my techniques with results from the Semantic Evaluation (SemEval) contest of automatic keyphrase extraction methods [80].

5.3.1 Statistical Modeling of Keyphrase Quality

I modeled keyphrase quality using logistic regression. I chose this model because its results are readily interpretable: contributions from each feature can be statistically assessed, and the regression value can be used to rank candidate phrases. I initially used a mixed effects model [55], which extends generalized linear models to let one assess random effects, to include variation due to subjects and documents. I found that the random effects were not significant and so reverted to a standard logistic regression model.

I constructed the models over 2,882 responses. I excluded user-generated keyphrases longer than five words (for which I am unable to determine term commonness; my data on web commonness contains only n-grams up to length five) or not present in the documents (for which I am unable to determine grammatical and positional information). I randomly selected another set of 28,820 phrases from the corpus as negative examples, with a weight of 0.1 (so that total weights for positive examples

and negative examples are equal during model fitting). Coefficients generated by logistic regression represent the best linear combination of features that differentiate user-generated responses from the random phrases.

I examine three classes of features — frequency statistics, grammar, and position — visited in order of their predictive accuracy as determined by a preliminary analysis. Unless otherwise stated, all features are added to the regression model as independent factors without interaction terms.

I present only modeling results for keyphrases describing single documents. I did fit models for phrases describing multiple documents, and they reflect observations from the previous section, for example, weights shifted toward higher commonness scores. However, the coefficients for grammatical features exhibit large standard errors, suggesting that the smaller data set of multi-document phrases (641 phrases vs. 2,882 for single docs) is insufficient. As a result, I leave further modeling of multi-document descriptions to future work.

I evaluate features using precision-recall curves. Precision and recall measure the accuracy of an algorithm by comparing its output to a known, "correct" set of phrases; in this case, the list of user-generated keyphrases up to length five. Precision measures the percentage of correct phrases in the output. Recall measures the total percentage of the correct phrases captured by the output. As more phrases are included, recall increases but precision decreases. The precision-recall curve measures the performance of an algorithm over an increasing number of output phrases. Higher precision is desirable with fewer phrases and a larger area under the curve indicates better performance. I also assessed each model using model selection criteria (i.e., AIC, BIC). As these scores coincide with the rankings from precision-recall measures, they are omitted.

Frequency Statistics

I computed seven different frequency statistics. My simplest measure was log term frequency: *log(tf)*. I also computed *tf.idf*, *BM25*, G^2, *variance-weighted log-odds ratio*, and *WordScore*. Each requires a reference corpus, for which I use the full dissertation abstract collection. I also created a set of *hierarchical tf.idf* scores (e.g.,

Statistic	Definition
log(tf)	$\log\left(t_{\text{Doc}}\right)$
tf.idf	$\left(t_{\text{Doc}}/t_{\text{Ref}}\right) \cdot \log\left(N/D\right)$
G^2	$2\left(t_{\text{Doc}} \log \frac{t_{\text{Doc}} \cdot T_{\text{Ref}}}{T_{\text{Doc}} \cdot T_{\text{Doc}}} + t_{\overline{\text{Doc}}} \log \frac{t_{\overline{\text{Doc}}} \cdot T_{\text{Ref}}}{T_{\overline{\text{Doc}}} \cdot T_{\text{Doc}}}\right)$
BM25	$3 \cdot t_{\text{Doc}} / \left(t_{\text{Doc}} + 2\left(0.25 + 0.75 \cdot T_{\text{Doc}}/r\right)\right) \cdot \log\left(N/D\right)$
WordScore	$\left(t_{\text{Doc}} - t_{\text{Ref}}\right) / \left(T_{\overline{\text{Doc}}} - T_{\overline{\text{Ref}}}\right)$
log-odds ratio (weighted)	$\left(\log \frac{t'_{\text{Doc}}}{t'_{\overline{\text{Doc}}}} - \log \frac{T'_{\text{Doc}}}{T'_{\overline{\text{Doc}}}}\right) / \sqrt{\frac{1}{t'_{\text{Doc}}} + \frac{1}{t'_{\overline{\text{Doc}}}}}$

Table 5.3: Frequency Statistics. Given a document from a reference corpus with N documents, the score for a term is given by these formulas. t_{Doc} and t_{Ref} denote term frequency in the document and reference corpus; T_{Doc} and T_{Ref} are the number of words in the document and reference corpus; D is the number of documents in which the term appears; r is the average word count per document; t' and T' indicate measures for which I increment term frequencies in each document by 0.01; terms present in the corpus but not in the document are defined as $t_{\overline{\text{Doc}}} = t_{\text{Ref}} - t_{\text{Doc}}$ and $T_{\overline{\text{Doc}}} = T_{\text{Ref}} - T_{\text{Doc}}$. Among the family of tf.idf measures, I selected a reference-relative form as shown. For BM25, the parameters $k_1 = 2$ and $b = 0.75$ are suggested by [95]. A term is any analyzed phrase (n-gram). When frequency statistics are applied to n-grams with $n = 1$, the terms are all the individual words in the corpus. When $n = 2$, scoring is applied to all unigrams and bigrams in the corpus, and so on.

as used by Viégas et al. in Themail [162]) by computing tf.idf with five nested reference corpora: all terms on the web, all dissertations in the Stanford dissertation corpus, dissertations from the same school, dissertations in the same department, and dissertations supervised by the same advisor. Due to its poor performance on 5-grams, I assessed four variants of standard tf.idf scores: tf.idf on unigrams, and all phrases up to bigrams, trigrams, and 5-grams. Formulas for frequency measures are shown in Table 5.3.

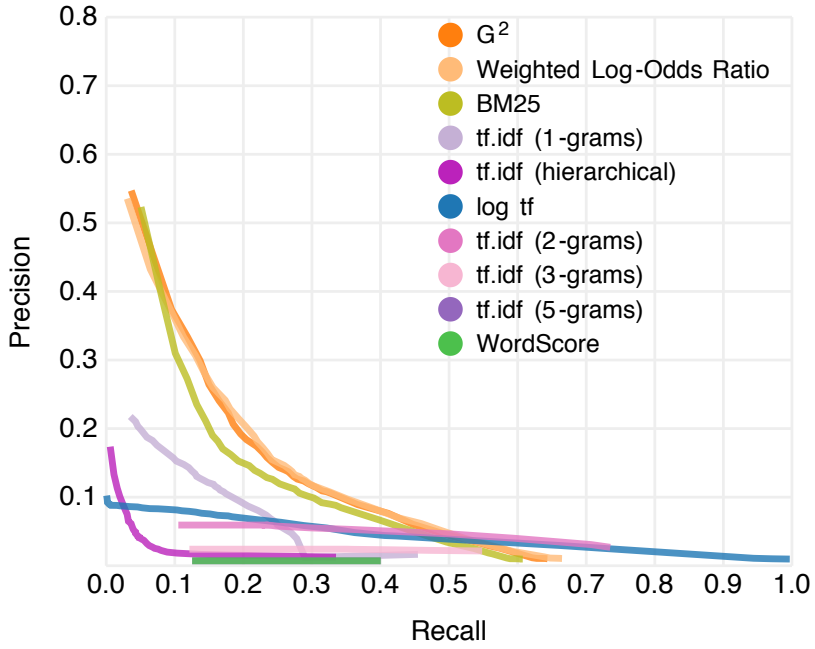

Figure 5.4: Precision-recall curves for keyphrase regression models. Among models based on only frequency statistics, G^2 and log-odds ratio perform well. Legends are sorted by decreasing initial precision.

Figure 5.4 shows the performance of these frequency statistics. Probabilistic measures — namely G^2, BM25 and weighted log-odds ratio — perform better than count-based approaches (e.g., tf.idf) and heuristics such as WordScore. Count-based approaches suffer with longer phrases due to an excessive number of ties (many 4- and 5-grams occur only once in the corpus). However, tf.idf on unigrams still performs much worse than probabilistic approaches.

Adding Term Commonness. During keyphrase characterization, I observed a bell-shaped distribution of keyphrases as a function of commonness. I quantiled commonness features into *web commonness* bins and *corpus commonness* bins in order to capture this non-linear relationship. I examined the effects of different bin counts up to 20 bins.

As shown in Figure 5.5, the performance of $\log(tf)$ + commonness matches that

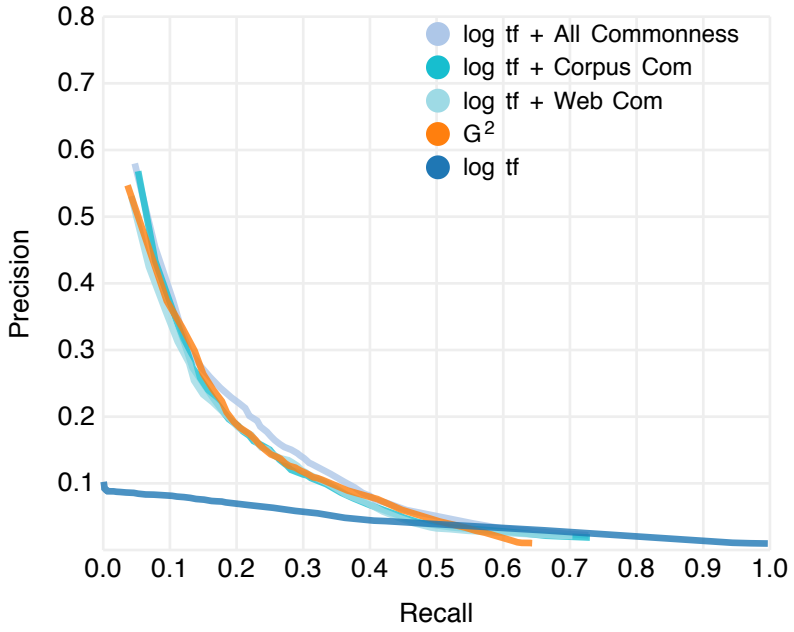

Figure 5.5: For keyphrase regression models based on frequency statistics and term commonness, a simple combination of $\log(tf)$ and commonness performs competitively to G^2. Graph shows precision-recall curves; legends are sorted by decreasing initial precision.

of statistical methods such as G^2. As corpus and web commonness are highly correlated, the addition of both commonness features yields only a marginal improvement over the addition of either feature alone. I also measured the effects due to bin count. Precision-recall increases as the number of bins are increased up to about five bins, and there is marginal gain between five and eight bins. Examining the regression coefficients for a large number of bins (ten bins or more) shows large random fluctuations, indicating overfitting. As expected, the coefficients for commonness peak at middle frequency; see Table 5.5. Adding an interaction term between frequency statistics and commonness yields no increase in performance. Interestingly, the coefficient for tf.idf is negative when combined with web commonness; tf.idf scores have a slight negative correlation with keyphrase quality.

Grammatical Features

Computing grammatical features requires either parsing or part-of-speech (POS) tagging. Of note is the higher computational cost of parsing — nearly two orders of magnitude in run time. I measure the effectiveness of these two classes of features separately to determine if the extra computational cost of parsing pays dividends.

Parser features. For each term extracted from the text, I tag the term as a *full noun phrase* or *full verb phrase* if it matches exactly a noun phrase or verb phrase identified by the parser. A term is tagged as a *partial noun phrase* or *partial verb phrase* if it matches a substring within a noun phrase or verb phrase. I add two additional features that are associated with words at the boundary of a noun phrase. Leading words in a noun phrase are referred to as *optional leading words* if their part-of-speech is one of cardinal number, determiner, or pre-determiner. The last word in a noun phrase is *head noun*. If the first word of a term is an optional leading word, or if the last word of a term is a head noun, then the term is tagged accordingly. These two features occur only if the beginning or end of the term is aligned with a noun phrase boundary.

Tagger features. Phrases that match technical term patterns (Table 5.1) are tagged as either a *technical term* or *compound technical term*. Phrases that match a substring in a technical term are tagged as *partial technical term* or *partial compound technical terms*.

As shown in Figure 5.6, adding parser-derived grammar information yields an improvement significantly greater than the differences between leading frequency statistics. Adding technical terms matched using POS tags improves precision and recall more than parser-related features. Combining both POS and parser features yields only a marginal improvement. Head nouns (cf., [6]) did not have a measurable effect on keyphrase quality. The results indicate that statistical parsing may be avoided in favor of POS tagging.

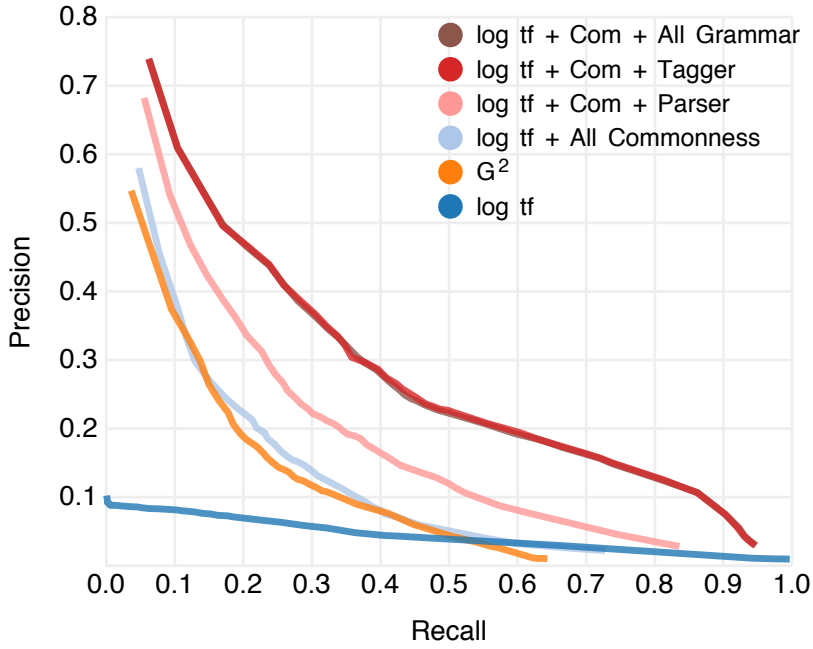

Figure 5.6: Adding part-of-speech features improve the performance of keyphrase regression models more than parser-related features. Combining both POS and parser features yields only a marginal improvement.

Positional Features and Final Models

Finally, I introduce *relative first occurrence* and *presence in first sentence* as positional features; both predictors are statistically significant.

First occurrence. The *absolute first occurrence* of a term is the earliest position in the document at which a term appears, normalized between 0 and 1. If a term is the first word of a document, its absolute first occurrence is 0. If the only appearance of a term is as the last word of a document, its absolute first occurrence is 1. The absolute first occurrences of frequent terms tend to be earlier in document, due to their larger number of appearances.

I introduce *relative first appearance* to have a measure of early occurrence of a word independent of its frequency. Relative first occurrence measures how likely a term is to initially appear earlier than a randomly-sampled phrase of the same frequency. Let $P(W)$ denote the the expected position of words W in the document. As a null

hypothesis, I assume that words are uniformly distributed $P(W) \sim \text{Uniform}[0,1]$. The expected absolute first occurrence of a randomly-selected term that appears k times in the document is the minimum of the k instantiations of the term $P(w_1), \cdots, P(w_k)$, and is given by the following probability distribution:

$$\min_{i=1}^{k} P(w_i) = \eta\,(1-x)^{k-1}$$

for position $x \in [0,1]$ and some normalization constant η. Suppose a term w' occurs k times in the document and its first occurrence is observed to be at position $a \in [0,1]$. Its relative first occurrence is the cumulative probability distribution from a to 1.

$$\text{Relative first occurrence of } w' = \int_a^1 \min_{i=1}^{k} P(w_i) = \int_a^1 \eta\,(1-x)^{k-1}\,dx = (1-a)^k$$

Combining $\log(tf)$, commonness (five bins), grammatical, and positional features, I built two final models for predicting keyphrase quality. The full model is based on all significant features using the dissertation corpus as reference. In the simplified model (Table 5.5), I excluded corpus commonness and statistical parsing to eliminate corpus dependencies and improve running time. Omitting the more costly features incurs a slight decrease in precision, as shown in Figure 5.7.

5.3.2 Comparison with Human-Selected Keyphrases

I compared the precision-recall of keyphrases extracted using my methods to human-generated keyphrases. In the previous comparisons of model performance, a candidate phrase was considered "correct" if it matched a term selected by any of the K human subjects who read a document. When evaluating human performance, however, phrases selected by one participant can only be matched against responses from the $K-1$ other remaining participants. A naïve comparison would thus unfairly favor my algorithm, as human performance would suffer due the smaller set of "correct" phrases. To ensure a meaningful comparison, I randomly sample a subset of K participants for each document. When evaluating human precision, a participant's response is considered accurate if it matches any phrase selected by another subject. I then

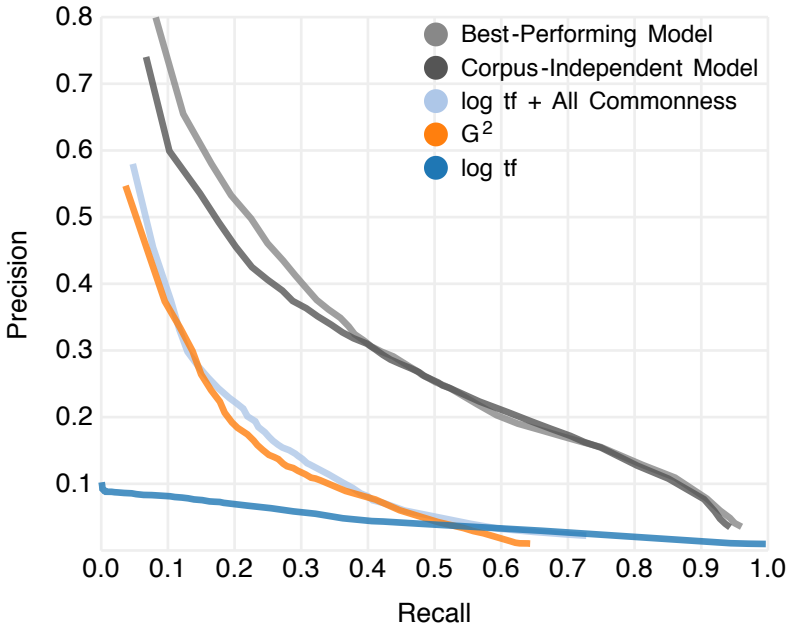

Figure 5.7: Positional features provide further gains for both a complete keyphrase regression model and a simplified corpus-independent model.

replace the participant's responses with the model's output, ensuring that both are compared to the same $K-1$ subjects. I chose $K=6$, as on average each document in the study was read by 5.75 subjects.

Figure 5.8 shows the performance of the two models versus human performance. At low recall (i.e., for the top keyphrase), the full model achieves higher precision than human responses, while the simplified model performs competitively. The full model's precision closely matches that of human accuracy until mid-recall values.

5.3.3 Comparison with SemEval 2010 Contest Task #5

Next I compared the precision-recall performance of the corpus-independent model to the results of the SemEval 2010 contest. Semantic Evaluation (SemEval) is a series of workshops focused on evaluating methods for specific text analysis problems. Task #5 of SemEval 2010 [80] compared 21 keyphrase extraction algorithms for scientific articles. A total of 244 articles from four different subdisciplines were chosen from

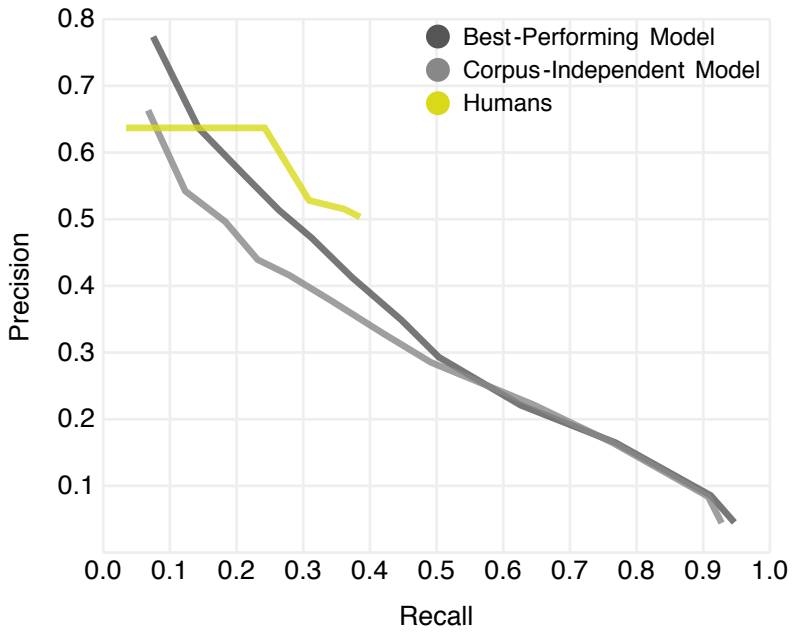

Figure 5.8: Comparison with human-selected keyphrases. My models provide higher precision at low recall values.

the ACM Digital Library. Contestants received 144 articles for training; the submitted techniques were then tested on the remaining 100 articles. Three classes of keyphrases were evaluated: author-assigned, reader-assigned, and the combination of both. Reader-assigned phrases were provided by volunteers who were given five papers and instructed to spend 10 to 15 minutes per paper generating keyphrases. For each class, precision and recall were computed for the top 5, 10, and 15 keyphrases.

I used this same data to evaluate the performance of the corpus-independent modeling approach trained on the SemEval corpus. The coefficients of the SemEval model differ slightly from those of the Stanford dissertations model (Table 5.5), but the relative feature weightings remain similar, including a preference for mid-commonness terms, a strong negative weight for high commonness, and strong weights for technical term patterns.

Figure 5.9 compares my model's precision-recall scores against the distribution of SemEval results for the combined author- and reader-assigned keyphrases. The corpus-independent model closely matches the median scores. Though intentionally

Model Feature	Regression Coefficients
(intercept)	-2.88114***
$\log(tf)$	0.74095***
WC $\in (0\%, 20\%]$	0.08894
WC $\in (20\%, 40\%]$	0.04390
WC $\in (40\%, 60\%]$	-0.19786
WC $\in (60\%, 80\%]$	-0.46664*
WC $\in (80\%, 100\%]$	-1.26714***
CC $\in (0\%, 20\%]$	0.20554
CC $\in (20\%, 40\%]$	0.39789**
CC $\in (40\%, 60\%]$	0.24929
CC $\in (60\%, 80\%]$	-0.34932
CC $\in (80\%, 100\%]$	-0.97702**
relative first occurrence	0.52950***
first sentence	0.83637**
partial noun phrase	0.14117
noun phrase	0.29818*
head noun	-0.16509
optional leading word	0.46481*
partial verb phrase	0.15639
verb phrase	1.12310*
full technical term	-0.58959
partial tech. term	1.37875*
full compound tech. term	1.09713
partial comp. tech. term	1.10565*

Table 5.4: Regression coefficients for the full (corpus-dependent) model based on the PhD dissertations. WC = web commonness. CC = corpus commonness. Statistical significance = *: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$

simplified, my approach matches or outperforms half of the contest entries. This outcome is perhaps surprising, as competing techniques include more assumptions and complex features (e.g., leveraging document structure and external ontologies) and more sophisticated learning algorithms (e.g., bagged decision trees vs. logistic regression). I believe these results argue in favor of the identified features.

Model Feature	Regression Coefficients	
	Dissertations	SemEval
(intercept)	-2.83499***	-5.4624**
$\log(tf)$	0.93894***	2.8029*
WC $\in (0\%, 20\%]$	0.17704	0.8561
WC $\in (20\%, 40\%]$	0.23044*	0.7246
WC $\in (40\%, 60\%]$	0.01575	0.4153
WC $\in (60\%, 80\%]$	-0.62049***	-0.5151
WC $\in (80\%, 100\%]$	-1.90814***	-2.2775
relative first occurrence	0.48002**	-0.2456
first sentence	0.93862***	0.9173
full tech. term	-0.50152	1.1439
partial tech. term	1.44609**	3.4539***
full compound tech. term	1.13730	1.0920
partial comp. tech. term	1.18057*	2.0134

Table 5.5: Regression coefficients for corpus-independent model. WC = web commonness. Statistical significance = *: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$

5.3.4 Lexical Variation and Relaxed Matching

While I am encouraged by the results of the precision-recall analysis, some skepticism is warranted. Up to this point my analysis has concerned only exact matches of stemmed terms. In practice, it is reasonable to expect that both people and algorithms will select keyphrases that do not match exactly but are lexically and/or conceptually similar (e.g., "analysis" vs. "data analysis"). How might the results change if we permit a more relaxed matching?

To gain a better sense of lexical variation among keyphrases, I analyzed the impact of a relaxed matching scheme. I experimented with a number of matching approaches by permitting insertion or removal of terms in phrases or re-arrangement of terms in genitive phrases. For brevity, I report on just one simple but effective strategy: I consider two phrases "matching" if they either match exactly or if an exact match can be induced by adding a single word to either the beginning or the end of the

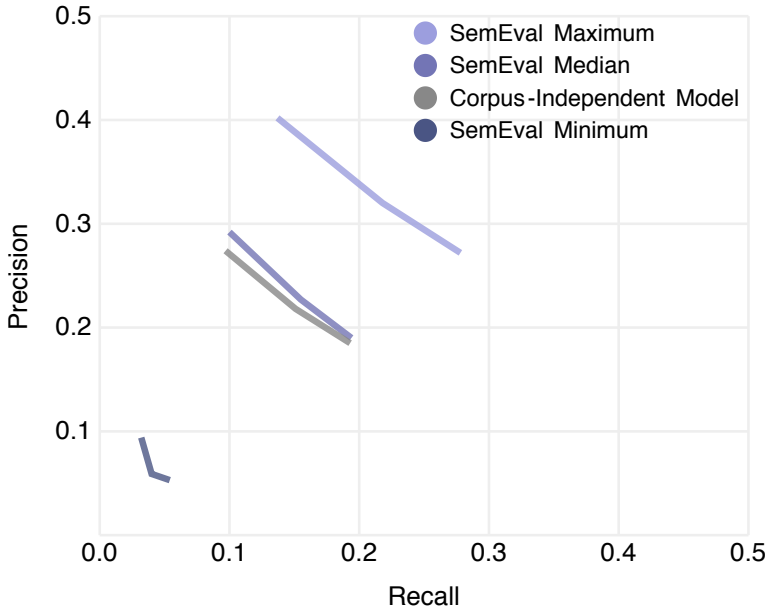

Figure 5.9: Comparison with SemEval 2010 [80] results for 5, 10, and 15 phrases. My corpus-independent model closely matches the median scores.

shorter phrase.

Permitting relaxed matching significantly raises the proportion of automatically extracted keyphrases that match human-selected terms. Considering just the top-ranked term produced by my model for each document in the SemEval contest, 30.0% are exact matches while 75.0% are relaxed matches. Looking at the top five terms per document, 27.4% exactly match a human-selected term, permitting a relaxed match increases this number to 64.2%. These results indicate that human-selected terms regularly differ from the automatically extracted terms by a single leading or trailing word. This observation suggests that (a) precision-recall analysis may not reveal the whole picture and (b) related keyphrases might vary in length but still provide useful descriptions. I now build upon this insight to provide means for parameterizing keyphrase selection.

5.4 Keyphrase Grouping and Selection

The previous section describes a method for scoring keyphrases in isolation. However, candidate keyphrases may overlap (e.g., "*visualization*", "*interactive visualization*") or reference the same entity (e.g., "*Barack Obama*", "*President Obama*"). Keyphrase selection might be further improved by identifying related terms. An intelligent grouping can also provide a means to interactively parameterizing the display of keyphrases. Users might request shorter/longer — or more general/more specific — terms. Alternatively, a user interface might automatically vary term length or specificity to optimize the use of the available screen space. Once I have extracted a set of candidate keyphrases, I can next optimize the overall quality of that set. Here I present a simple approach for filtering and selecting keyphrases — sufficient for removing a reasonable amount of redundancy and adapt keyphrase specificity on demand.

5.4.1 Redundancy Reduction

Redundancy reduction suppresses phrases similar in concept. The goal is to ensure that each successive output keyphrase provides a useful marginal information gain instead of lexical variations. For example, the following list of keyphrases differ lexically but are similar, if not identical, in concept: "*Flash Player 10.1*", "*Flash Player*", "*Flash*". I propose that an ideal redundancy reduction algorithm should group together phrases that are similar in concept (e.g., perhaps similar to synsets in WordNet), choose the most prominent lexical form of a concept, and suppress other redundant phrases.

I use string similarity to approximate conceptual similarity. I consider two phrases A and B to be similar if A can be constructed from B by prepending or appending a word. For example, "*Flash Player 10.1*" and "*Flash Player*" are considered similar. For many top-ranked keyphrases, this assumption is true. Figure 5.10 shows an example of terms considered conceptually similar by my algorithm.

I also account for the special case of names. I apply named entity recognition [57] to identify persons, locations, and organizations. To resolve entities, I consider two people identical if the trailing substring of one matches the trailing substring of

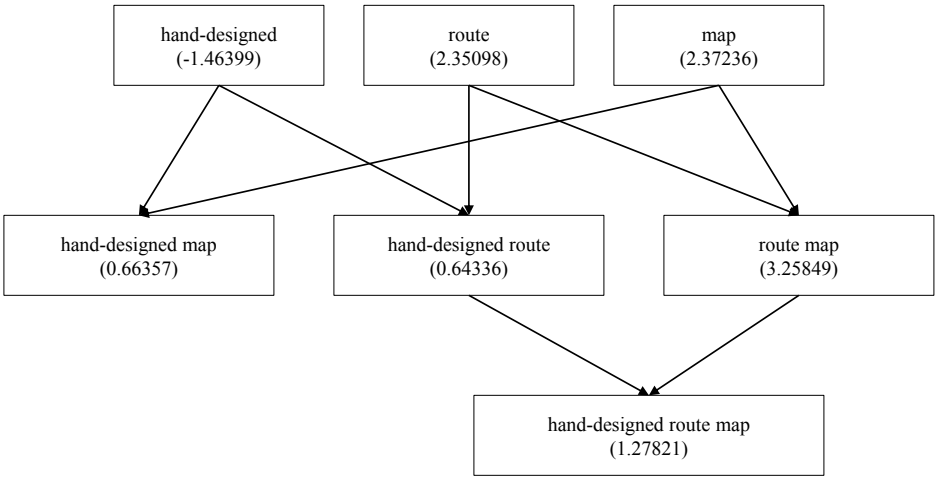

Figure 5.10: Term grouping. The above graph shows a subset of unigrams, bigrams, and trigrams considered to be conceptually similar by my algorithm. Connected terms differ by exactly one word at the start or the end of the longer phrase. Values in parentheses are the scores from the simplified model for the dissertation "Visualizing Route Maps." By default, my algorithm displays the keyphrase "*route map*" and suppresses "*route*", "*map*", and "*hand-designed route maps*". Users may choose to display a shorter word ("*map*") or longer phrase ("*hand-designed route map*") to describe this document.

the other. For example, "*Obama*", "*President Obama*", and "*Barack Obama*" are considered the same person. If the name of a location or organization is a substring of another, I consider the two to be identical, for example, "*Intel*" and "*Intel Corporation*". I also apply acronym recognition [143] to identify the long and short forms of the same concept, such as "*World of Warcraft*" and "*WoW*". For most short texts my assumptions hold. However, in general, a more principled approach will likely be needed for robust entity and acronym resolution. Figure 5.11 shows additional typed edges connecting terms that my algorithm considers as referring to the same entity.

5.4.2 Length and Specificity Adjustment

Once similar terms have been grouped, I must select which term to present. To parameterize final keyphrase selection, I allow users to optionally choose longer/shorter and more generic or specific terms. I use two simple features to determine which form of similar phrases to display: term length and term commonness. When two terms are deemed similar, I can bias for longer keyphrases by subtracting the ranking score from the shorter of the two terms and adding that to the score of the longer term, in proportion to the difference in term length. Similarly, I can bias for more generic or specific terms by shifting the ranking score between similar terms in proportion to the difference in term commonness. The operation is equivalent to shifting the weights along edges in Figures 5.10 and 5.11.

Other adjustments can be specified directly by users. For recognized people, users can choose to expand all names to full names or contract to last names. For locations and organizations, users can elect to use the full-length or shortened form. For identified acronyms, users may choose to expand or contract the terminology. In other words, for each subgraph of terms connected by named entity typed edges, the user may choose to assign the maximum node weight to any other nodes in the subgraph. In doing so, the chosen term is displayed suppressing all other alternatives.

5.4.3 Qualitative Inspection of Selected Keyphrases

As an initial evaluation of my two-stage extraction approach, I compared the top 50 keyphrases produced by my models with outputs from G^2, BM25, and variance-weighted log-odds ratio. I examined both dissertation abstracts from the user study and additional documents described in the next section. Terms from the 9,068 Ph.D. dissertations are used as the reference corpus for all methods except my simplified model, which is corpus independent. I applied redundancy reduction to the output of each extraction method.

My regression models often choose up to 50 or more reasonable keyphrases. In contrast, I find that G^2, BM25, and variance-weighted log-odds ratio typically select a few reasonable phrases but start producing unhelpful terms after the top ten results.

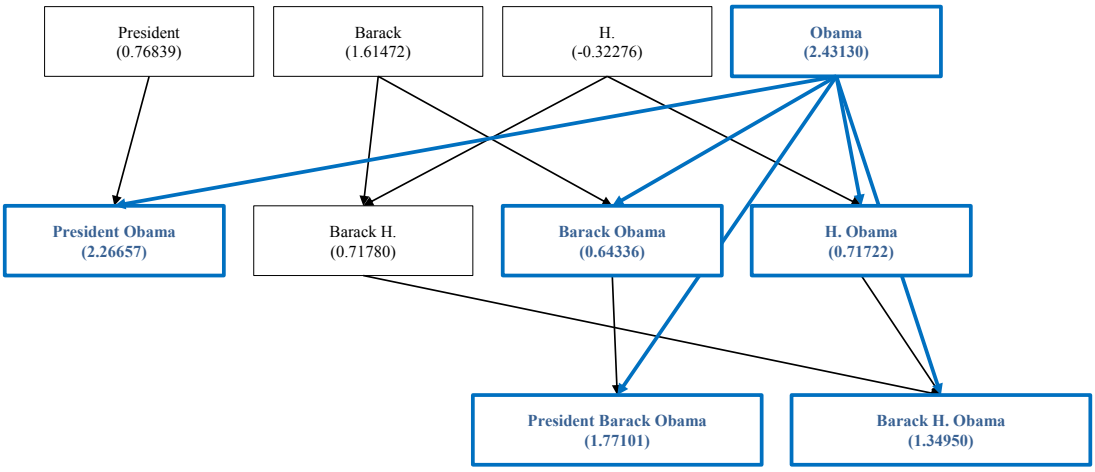

Figure 5.11: Term grouping for named entities and acronyms. The above graph shows typed edges that embed additional relationships between terms in a document about President Obama. Black edges represent basic term grouping based on string similarity. Bold blue edges represent people: terms that share a common trailing substring and are tagged as "person" by a named entity recognition algorithm. By default, my algorithm displays "*Obama*" to summarize the text. Users may choose to show a longer phrase "*President Obama*" or display a longer and more specific description "*President Barack Obama*" by shifting the scores along the typed edges. Users may also apply type-specific operations, such as showing the longest name without honorifics, "*Barack H. Obama*".

The difference is exacerbated for short texts. For example, in a 59-word article about San Francisco's Mission District, my algorithm returns noun phrases such as "*colorful Latino roots*" and "*gritty bohemian subculture*", while the other methods produce only one to three usable phrases: "*Mission*", "*the District*", or "*district*". In these cases, my method benefits from grammatical information.

My algorithm regularly extracts salient longer phrases, such as "*open-source digital photography software platform*" (not chosen by other algorithms), "*hardware-accelerated video playback*" (also selected by G^2, but not others), and "*cross platform development tool*" (not chosen by others). Earlier in the exploratory analysis,

I found that the inclusion of optional leading words degrades the quality of descriptive phrases. However, many phrases tend to be preceded by the same determiner and pre-determiner. Without a sufficiently large reference corpus, statistics alone often cannot separate meaningful phrases from common leading words. By applying technical term matching patterns, my model naturally excludes most types of non-descriptive leading words and produces more grammatically appropriate phrases such as "*long exposure*" (my models) versus "*a long exposure*" (G^2, BM25, weighted log-odds ratio). Even though term commonness favors mid-frequency phrases, my model can still select salient words from all commonness levels. For example, from an article about the technologies in Google versus Bing, my models choose "*search*" (common word), "*navigation tools*" (mid-frequency phrase), and "*colorful background*" (low-frequency phrase), while all other methods output only "*search*".

I observe few differences between the full and simplified models. Discernible differences are typically due to POS tagging errors. In one case, the full model returns the noun phrase "*interactive visualization*", but the simplified model returns "*interactive visualization leverage*", as the POS tagger mislabels "*leverage*" as a noun.

On the other hand, the emphasis on noun phrases can cause my algorithm to omit useful verb phrases, such as "*civilians killed*" in a news article about the NATO forces in Afghanistan. My algorithm chooses "*civilian casualties*" but places it significantly lower down the list. I return several phrases with unsuitable prefixes such as "*such scenarios*" and "*such systems*" because the word "*such*" is tagged as an adjective in the Penn Treebank tag set, and thus the entirety of the phrase is marked as a technical term. Changes to the POS tagger, parser, or adding conditions to the technical term patterns could ameliorate this issue. I also note that numbers are not handled by the original technical term patterns [78]. I modified the definition to include trailing cardinal numbers to allow for phrases such as "*H. 264*" and "*Windows 95*", dates such as "*June 1991*", and events such as "*Rebellion of 1798*".

Prior to redundancy reduction, I often observe redundant keyphrases similar in term length, concept, or identity. For example, "*Mission*", "*Mission District*", and "*Mission Street*" in an article about San Francisco. My heuristics based on string similarity, named entity recognition, and acronym recognition improve the returned

keyphrases (see Tables 5.6 and 5.7). As I currently consider single-term differences only, some redundancy is still present.

5.4.4 Crowdsourced Ratings of Tag Clouds

I evaluated my extracted keyphrases in a visual form, and asked human judges to rate the relative quality of tag cloud visualizations with terms selected using both my technique (i.e., simplified model) and G^2 scores of unigrams (cf., [42, 51, 131]). I chose to compare tag cloud visualizations for multiple reasons. First, keyphrases are often displayed as part of a webpage or text visualization; I hypothesize that visual features such as layout, sizing, term proximity, and other aesthetics are likely to affect the perceived utility of, and preferences for keyphrases in real-world applications. Tag clouds are a popular form used by a diverse set of people [164]. Presenting selected terms in a simple list would fail to reveal the impact of these effects. Second, keyphrases are often displayed in aggregate; I hypothesize that the perceived quality of a collective set of keyphrases differs from that of evaluating each term independently. Tag clouds encourage readers to assess the quality of keyphrases as a whole.

Parallel Tag Clouds [42] use unigrams weighted by G^2 for text analytics, making G^2 statistics an interesting and ecologically valid comparison point. I hypothesized that tag clouds created using my technique would be preferred due to more descriptive terms and complete phrases. I also considered variable-length G^2 that includes phrases up to 5-grams. Upon inspection, many of the bigrams (e.g., "*more about*", "*anyone can*") and the majority of trigrams and longer phrases selected by G^2 statistics are irrelevant to the document content. I excluded the results from the study as they were trivially uncompetitive. Including only unigrams results in shorter terms, which may lead to a more densely-packed layout (this is another reason that I chose to compare to G^2 unigrams).

Method

I asked subjects to read a short text passage and write a 1–2 sentence summary. Subjects then viewed two tag clouds and were asked to rate which they preferred on

My Corpus-Independent Model	G^2
Adobe	Flash
Flash Player	Player
technologies	Adobe
H. 264	video
touch-based devices	Flash Player is
runtime	264
surge	touch
fair amount	open source
incorrect information	10.1
hardware-accelerated video playback	Flash Player 10.1
Player 10.1	SWF
touch	the Flash Player
SWF	more about
misperceptions	content
mouse input	H.
mouse events	battery life
Seventy-five percent	codecs
codecs	browser
many claims	desktop
content protection	FLV/F4V
desktop environments	Flash Player team
Adobe Flash Platform	Player 10.1 will
CPU-intensive task	actively maintained
appropriate APIs	Anyone can
battery life	both open and proprietary
further optimizations	ecosystem of both
Video Technology Center	ecosystem of both open and
memory use	for the Flash
Interactive content	hardware-accelerated
Adobe Flash Player runtime	hardware-accelerated video playback
static HTML documents	include support
rich interactive media	multitouch
tablets	of both open
new content	on touch-based
complete set	open source and is

Table 5.6: Top 25 keyphrases for an open letter from Adobe about Flash technologies. I apply redundancy reduction to both lists.

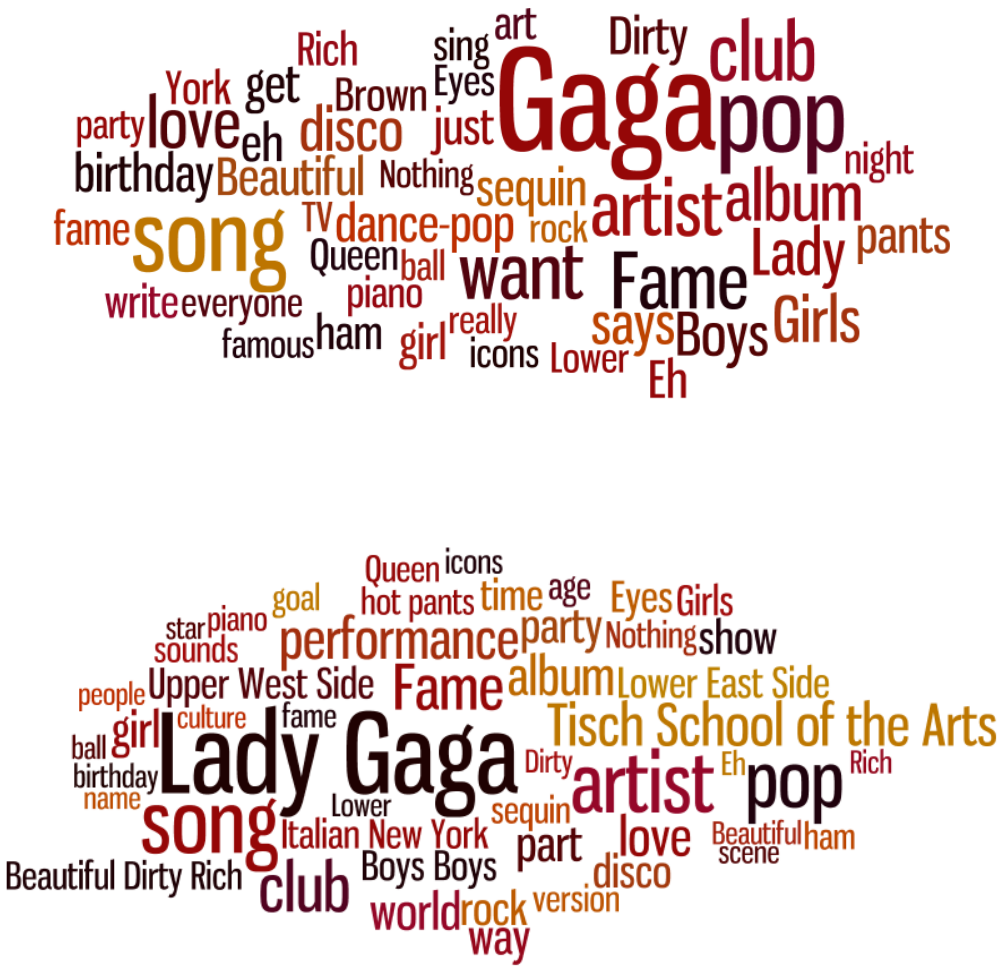

Figure 5.12: Tag cloud visualizations of an online biography of the pop singer Lady Gaga. Top: single-word phrases (unigrams) weighted using G^2. Bottom: multi-word phrases, including significant places and song titles, selected using my corpus-independent model.

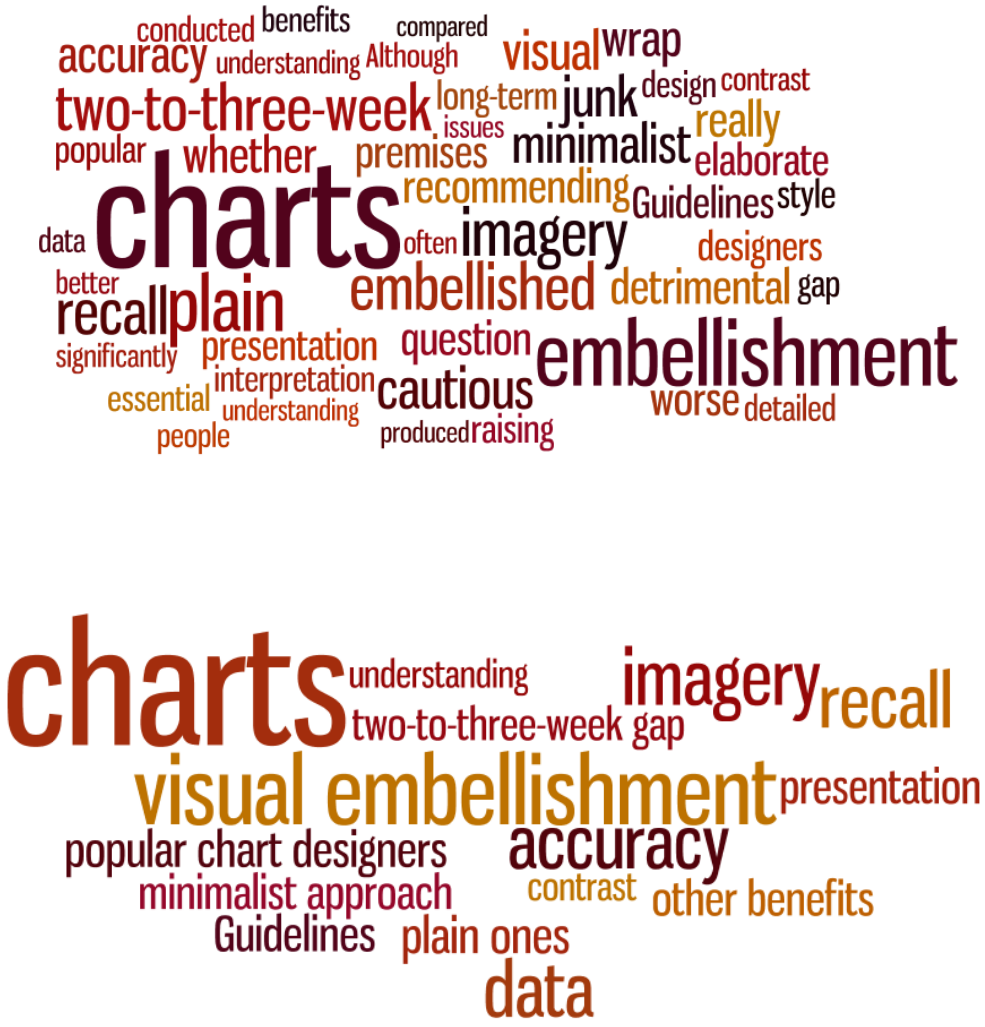

Figure 5.13: Tag clouds for a research paper on chart perception. Top: unigrams weighted using G^2. Bottom: multi-word phrases selected by my method.

Figure 5.14: Tag clouds for a travel article. Top: unigrams weighted using G^2. Bottom: multi-word phrases selected by my method.

Shorter		Keyphrase		Longer
Flash	←	**Flash Player**	→	Flash Player 10.1
devices	←	**mobile devices**	→	Apple mobile devices
happiness	←	national happiness	←	**Gross national happiness**
emotion	←	**emotion words**	→	use of emotion words
networks	←	**social networks**	→	online social networks
Obama	←	President Obama	←	**Barack H. Obama**
Bush	←	President Bush	←	**George H.W. Bush**
		WoW	→	World of Warcraft

Table 5.7: Term length adjustment. Examples of adjusting keyphrase length. Terms in boldface are selected by my corpus-independent model. Adjacent terms show the results of dynamically requesting shorter (←) or longer (→) terms.

a 5 point scale (with '3' indicating a tie) and provide a brief rationale for their choice. I asked raters to "consider to what degree the tag clouds use appropriate words, avoid unhelpful or unnecessary terms, and communicate the gist of the text." One tag cloud consisted of unigrams with term weights calculated using G^2; the other contained keyphrases selected using the corpus-independent model with redundancy reduction and with the default preferred length. I weighted the terms by their regression score: the linear combination of features used as input to the logistic function. Each tag cloud contained the top 50 terms, with font sizes proportional to the square root of the term weight. Occasionally my method selected less than 50 terms with positive weights; I omitted negatively-weighted terms. Tag cloud images were generated by Wordle [164] using the same layout and color parameters for each. I randomized the presentation order of the tag clouds.

I included tag clouds of 24 text documents. To sample a variety of genres, I used documents in four categories: CHI 2010 paper abstracts, short biographies (based on three U.S. presidents and three musicians), blog posts (two each from opinion, travel, and photography blogs), and news articles. Figure 5.12 shows tag clouds from a biography of the singer Lady Gaga; Figures 5.13 and 5.14 show two other clouds used in the study.

I conducted this study using Amazon Mechanical Turk (cf., [70]). Each trial was posted as a task with a US$0.10 reward. I requested 24 assignments per task, resulting in 576 ratings. Upon completion, I tallied the ratings for each tag cloud and coded free-text responses with the criteria invoked by raters' rationales.

Results

On average, raters significantly preferred tag clouds generated using my keyphrase extraction approach (267 ratings vs. 208 for G^2 and 101 ties; $\chi^2(2) = 73.76$, $p < 0.0001$). Moreover, my technique garnered more strong ratings: 49% (132/267) of positive ratings were rated as "MUCH better," compared to 38% (80/208) for G^2.

Looking at raters' rationales, I find that 70% of responses in favor of my technique cite the improved saliency of descriptive terms, compared to 40% of ratings in favor of G^2. More specifically, 12% of positive responses note the presence of terms with multiple words ("It's better to have the words 'Adobe Flash' and 'Flash Player' together"), while 13% cite the use of fewer, unnecessary terms ("This is how tag clouds should be presented, without the clutter of unimportant words"). On the other hand, some (16/208, 8%) rewarded G^2 for showing more terms ("Tag cloud 2 is better since it has more words used in the text.").

Tag clouds in both conditions were sometimes preferred due to visual features such as layout, shape, and density: 29% (60/208) for G^2 and 23% (61/267) for my technique. While visual features were often mentioned in conjunction with remarks about term saliency, G^2 led to more ratings (23% vs. 14%) that mentioned only visual features ("one word that is way bigger than the rest will give a focal point ... it is best if that word is short and in the center").

The study results also reveal limitations of my keyphrase extraction technique. While my approach was rated superior for abstracts, biographies, and blog posts, on average, G^2 fared better for news articles. In one case, this was due to layout issues (a majority of raters preferred the central placement of the primary term in the G^2 cloud), but others specifically cite the quality of the chosen keyphrases. In an article about racial discrimination in online purchasing, my technique disregarded the term "black" due to its commonness and adjective part-of-speech. The tendency of my

technique to give higher scores to people names non-central to the text at times led raters to prefer G^2. In general, prominent mistakes or omissions by either technique were critically cited.

Unsurprisingly, my technique was preferred by the largest margin for research paper abstracts, the domain closest to the training data. This observation suggests that applying my modeling methodology to human-selected keyphrases from other text genres may result in better selections. The study also suggests that we might improve our keyphrase weighting by better handling named entities, so as to avoid giving high scores to non-central actors. Confirming our hypothesis, layout affects tag cloud ratings. The ability to dynamically adjust keyphrase length, however, can produce alternative terms and may allow users to generate tag clouds with better spatial properties.

5.5 Implications for HCI, InfoVis, and NLP

In this section, I highlight my contributions to the fields of human-computer interaction (HCI), information visualization (InfoVis), and natural language processing (NLP). First, I summarize my experiences and distill them in a set of design guidelines. Second, I demonstrate how my work can enable novel interactive visualizations. Finally, my keyphrase extraction algorithm is the cumulative result of applying HCI methods to collect data, analyze, develop, and evaluate text summarization techniques. I review the process through which I arrived at my model and emphasize how HCI concepts and approaches can help advance research in natural language processing and other fields.

Guidelines for Human-Centered Design

I summarize the key lessons from my study and evaluations and distill them in the following set of guidelines on designing text visualizations and model feature selection.

Multi-word phrases. My results find that multi-word phrases — particularly bigrams — often provide better descriptions than unigrams alone. In the case of multiple documents, this decision may need to be traded off against the better aggregation

afforded by unigrams. Designers may wish to give users the option to parameterize phrase length. My grouping approach (Section 5.4) provides a means of parameterizing selection while preserving descriptive quality.

Choice of frequency statistics. In my studies, probabilistic measures such as G^2 significantly outperformed common techniques, such as raw term frequency and tf.idf. Moreover, a simple linear combination of log term frequency and web commonness matches the performance of G^2 without the need of a domain-specific reference corpus. I advocate using these higher-performing frequency statistics when identifying descriptive keyphrases.

Grammar and position. At the cost of additional implementation effort, my results show that keyphrase quality can be further improved through the addition of grammatical annotations (specifically, technical term pattern matching using part-of-speech tags) and positional information. The inclusion of these additional features can improve the choice of keyphrases. More computationally costly statistical parsing provides little additional benefit.

Keyphrase selection. When viewed as a set, keyphrases may overlap or reference the same entity. My results show how text visualizations might make better use of screen space by identifying related terms (including named entities and acronyms) and reducing redundancy. Interactive systems might leverage these groupings to enable dynamic keyphrase selection based on term length or specificity.

Potential effects of layout and collective accuracy. My study comparing tag cloud designs provides examples suggesting that layout decisions (e.g., central placement of the largest term) and collective accuracy (e.g., prominent errors) impact user judgments of keyphrase quality. My results do not provide definitive insights but suggest that further studies on the spatial organization of terms may yield insights for more effective layout and that keyphrase quality should not be assessed in isolation.

Applications to Interactive Visualization

In this section, I illustrate how my keyphrase extraction methods can enable novel interactions with text. I present two example applications: phrase-level text summarization and dynamic adjustment of keyphrase specificity.

Figure 5.15: Parallel tag cloud using my keyphrase extraction algorithm as the underlying text processing step. The columns contain the top 50 keyphrases (without redundancy reduction) in chapters 3 through 12 of Lewis Carroll's *Alice's Adventures in Wonderland*. Longer phrases enable display of entities, such as "*Cheshire Cat*" and "*Lobster Quadrille*", that might be more salient to a reader than unigrams alone. Term grouping can enable novel interaction techniques such as brushing-and-linking conceptually similar terms. When a user selects the word "*tone*", the visualization shows the similar but changing tones in Alice's adventures from "*melancholy tone*" to "*solemn tone*" and from "*encouraging tone*" to "*hopeful tone*" as the story develops.

I apply keyphrase extraction algorithm to Lewis Carroll's *Alice's Adventures in Wonderland*, and compare the text in each chapter using a Parallel Tag Cloud in Figure 5.15. Each column contains the top 50 keyphrases (without redundancy reduction) from a chapter of the book. By extracting longer phrases, my technique enables the display of entities, such as "*Cheshire Cat*" and "*Lobster Quadrille*", that might be more salient to a reader than a display of unigrams alone. My term grouping approach can enable novel interactions. For example, when a user mouses over a term, the visualization highlights all terms that are considered conceptually similar. As shown in Figure 5.15, when the user selects the word "*tone*", the visualization shows the similar but changing tones in Alice's adventures from "*melancholy tone*" to "*solemn tone*" and from "*encouraging tone*" to "*hopeful tone*" as the story develops.

My algorithm can enable text visualizations that respond to different audiences. The tag clouds in Figures 5.16 and 5.17 show the top keyphrases of an article discussing a new subway map by the New York City Metropolitan Transportation Authority. By adjusting the model output to show more specific or more general terms, the tool can adapt the text for readers with varying familiarity with the city's subway system. For example, a user might interactively drag a slider to explore different levels of term specificity. The top tag cloud provides a general gist of the article and of the redesigned map. By increasing term specificity, the middle tag cloud progressively reveals additional terms including neighborhoods, such as "*TriBeCa*", "*NoHo*", and "*Yorkville*", that may be of interest to local residents. The bottom tag cloud provides additional details, such as historical subway maps with the "*Massimo Vignellis abstract design*".

Applications of HCI Methods to Natural Language Processing

In addition to contributing a keyphrase extraction algorithm, I would like to emphasize the process through which the algorithm was developed. I highlight the various steps at which I applied human-centered design methods and point out how HCI concepts helped guide the development. I hope that my experiences can serve as an example for creating algorithms that are responsive to users' tasks and needs.

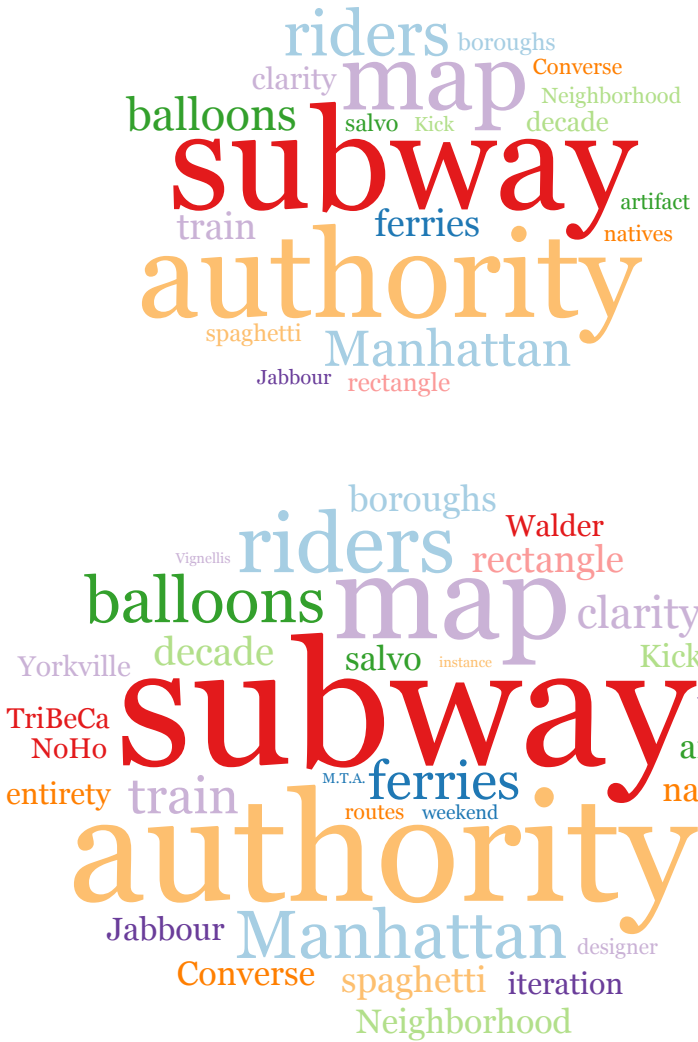

Figure 5.16: Adaptive tag clouds; continue onto Figure 5.17. These tag clouds summarize an article about the new subway map by the New York City Metropolitan Transportation Authority. By adjusting the model output to show more specific or more general terms, a visualization can adapt the text for readers with varying familiarity with the city's subway system. For example, a user might interactively drag a slider to explore different levels of term specificity. The top tag cloud provides a general gist of the article and of the re-designed map. By increasing term specificity, the bottom tag cloud progressively reveals additional terms including neighborhoods such as "*TriBeCa*", "*NoHo*", and "*Yorkville*" that may be of interest to local residents. Tag cloud in Figure 5.17 provides additional details such as historical subway maps with the "*Massimo Vignellis abstract design*".

Figure 5.17: Adaptive tag clouds; continued from Figure 5.16.

My model arose through the cumulative application of HCI methods to collecting data, and analyzing, developing, and evaluating text summarization techniques. First, I collected human-generated keyphrases via a formal experiment. The data enabled us to examine the relationships between the descriptors and the corresponding text in a systematic manner and to determine the effects of three controlled factors. Second, an exploratory analysis yielded insights for designing more effective algorithms. I assessed the quality of various linguistic and grammatical features (e.g., accuracy of existing frequency statistics, computational cost of tagging vs. parsing) and characterized the properties of high-quality descriptors. The characterizations enabled identification of appropriate natural language processing techniques (e.g., technical terms for approximating noun phrases). In turn, the choice of features led to a simple regression model that is competitive with outputs generated by more

advanced statistical models. Third, I designed ecologically valid evaluations. In addition to standard quantitative measures (e.g., precision-recall on exact matches), I evaluated the extracted keyphrases in situations closer to the actual context of use. An analysis using relaxed matching yielded insights on the shortcomings of the standard equality-based precision-recall scores and provided the basis for my redundancy reduction algorithm. Evaluating keyphrase use in tag clouds revealed effects due to visual features as well as the impact of prominent mistakes.

While many of the proceeding concepts may be familiar to HCI practitioners, their uses in natural language processing are not widely adopted. Incorporating HCI methods, however, may benefit various active areas of NLP research.

Summary

In this chapter, I characterize the statistical and grammatical features of human-generated keyphrases and present a model for identifying highly descriptive terms in a text. The model allows for adjustment of keyphrase specificity to meet application and user needs. Based on simple linguistic features, my approach does not require a pre-processed reference corpus, external taxonomies, or genre-specific document structure while supporting interactive applications. Evaluations reveal that my model is preferred by human judges, can match human extraction performance, and performs well even on short texts.

Finally, the process through which I arrived at my algorithm — identifying human strategies via a formal experiment and exploratory analysis, designing my algorithm based on these identified strategies, and evaluating its performance in ecologically-valid settings — demonstrates how human-centered design methods can be applied to the design and development of effective algorithms. A holistic approach to co-designing algorithms and visualizations can enable novel interactive techniques and user interface designs.

Chapter 6

Conclusion

In this dissertation, I presented the results of applying a human-centered iterative design process to a variety of projects: visualizations of statistical topic models, analysis tools to support topical quality assessment, a framework to support large-scale topical relevance assessment, and descriptive phrases for text summarization. My work has produced effective interactive visualizations, enabled more efficient analytic workflows, and contributed to our understanding of human categorization, topic modeling, and text summarization. I demonstrated how we can effectively integrate methods from information visualization, human-computer interaction, and machine learning to support effective model-driven data analysis.

I distilled *design principles* and *design processes* to inform practitioners on how to incorporate increasingly sophisticated models into data analysis tools. I designed, developed, and deployed various *visual analysis tools* for both builders and end users of statistical topic models. In all of my projects, my approach led to not only *improved visualizations* but also the design of *novel modeling techniques*. I contributed *survey methods* and various *datasets* that can enable future studies on human-centered approaches to topic modeling. To conclude, I discuss potential future work.

6.1 Review of Contributions

6.1.1 Design Guidelines

Based on my experiences and a review of relevant literature, I formulated two design principles — *interpretation* and *trust* — for for creating visualizations driven by statistical models. I distilled a set of design processes — *align*, *modify*, *verify*, and *progressive disclose* — for achieving interpretable and trustworthy visualizations.

6.1.2 Visual Analysis Tools

I developed a set of visual analysis tools to help social scientists examine large-scale academic discourse. The *Stanford Dissertation Browser* contributed to an investigation into inter-disciplinary collaborations. My *topic flow visualization tool* revealed modeling issues in an existing topic modeling algorithm. My *visualization of language transfer* allowed social scientists to examine three decades of academic discourse based on topical analyses of over one million Ph.D. dissertations.

I developed *Termite*, a visual analysis tool for evaluating topic model quality. Termite supports rapid visual assessment through the use of a matrix view, the identification of distinctive vocabulary, and term seriation to promote the clustering of related words and the legibility of phrases.

I contributed a *computational framework* to support large-scale assessment of topical relevance. I quantified four types of topical misalignment — junk, fused, missing, and repeated topics — and introduced the correspondence chart, a visualization to provide diagnostic feedback on topical alignment.

6.1.3 Modeling and Visualization Techniques

In collaboration with social scientists and machine learning researchers, I devised a novel *word borrowing* topical similarity measure during the development of the Stanford Dissertation Browser. My measure more closely matched expert judgment of departmental topical relationships, and produced asymmetric relationships that were expressed by the experts but not captured by existing techniques.

I developed a novel *term saliency* measure and a novel *text seriation* technique that were incorporated into Termite. My saliency measure identifies distinctive vocabulary suitable for topical comparison. My seriation technique not only surfaces the clustering of related words but also promotes the legibility of phrases (multi-word terms) through the incorporation of bigram collocation statistics.

I contributed a *matching likelihood* measure predictive how likely a human judge would consider a latent topic and a reference concept to be equivalent. The measure is based on a *rescaled dot product* calculation that outperformed existing techniques in predicting user topical similarity ratings. The matching likelihood measure was incorporated into my framework for assessing topical relevance.

I developed a novel *keyphrase extraction algorithm* based on an analysis of human-generated descriptive phrases. I presented two novel interactive text visualizations that were enabled by my algorithm: *phrase-level text summarization* and a tag cloud with *dynamic adjustment of keyphrase specificity*.

6.1.4 Survey Methods and Datasets

I contributed a survey method for *eliciting topical organization based on freeform responses*. I identified four issues (i.e., bias, recall, input accuracy, and participant exhaustion) associated with collecting open-ended categorization responses, and devised user interface and survey design modifications to address these issues. Using the survey method, I asked ten experienced researchers to describe topics of information visualization research and collected 202 hand-crafted topical responses, each consisting of a title, keyphrases, and representative documents.

I contributed a method for *synthesizing* similar topical concepts a corresponding method for *validating* the combined topics. I identified a set of 28 most coherent topical concepts in information visualization.

I also compiled a corpus of over 5,600 *descriptive phrases*, manually chosen by expert and non-expert readers, for summarizing Ph.D. dissertation abstracts.

6.2 Limitations and Future Work

6.2.1 Interactive Topic Modeling

While I compared topic models against an exhaustive list of reference concept provided by experts in my work, I believe the framework is useful when users specify only a subset of the concepts, or can construct concepts from existing metadata. Multiple research questions need to be addressed to support such a modeling workflow.

What learning technique should we apply? Several semi-supervised topic models permit users to express domain knowledge by specifying exemplary documents [129], constraints on word relationships [4], or by treating a given word distribution as observed [128]. The choice of model needs to be made by considering both the performance of the model (i.e., modeling error) and other human factors. For example, how accurately and efficiently can experts express their domain knowledge in a representation suitable for the model?

I examined the effects of three factors — number of latent topics (N), term smoothing (β), and topic smoothing (α) — on the quality of statistical topic models trained on information visualization publications. How well do these results hold for larger text corpora? Or, for other academic publication datasets? Or, for other domains of text? Topic model quality also depends on other factors as well as pre- and post-processing steps. For example, what are the effects of asymmetric priors [166]? What are the effects of stopword removal or the introduction of domain-specific phrases?

6.2.2 Hierarchy in Human Topical Organization

Previous psychology work suggests that humans organize categories hierarchically and that categories are first created at a basic level before more general and more specific categories emerge. My survey results from the information visualization experts consisted of a mostly flat list of topics. How do we design survey methods to identify hierarchical structures in topical organization?

People are able to assign *subordinate* and *superordinate* relationships to concepts.

Cognitive psychology experiments are typically based on small datasets, where the extracted hierarchical relationships are presented using a taxonomy or *a tree of concepts*. This presentation, however, sidesteps the issue that local subordinate/superordinate relationships may be inconsistent — for larger or more complex datasets or when the data is elicited from people with different domain expertise.

Assuming we are able to elicit hierarchical topical organization from a large number of users, can their responses be fitted to a strict tree structure? If not, what is the appropriate modeling abstraction for representing hierarchical topical organization? Assuming we are able to efficiently elicit and accurately model hierarchical human topical organization, how do we design tools, such as search interfaces, to best support browsing and exploration by topical concepts?

6.2.3 Facets vs. Categories

In user interface design, an established paradigm to support effective browsing of large datasets is through the use of *faceted navigation*. A facet is a superordinate category that groups together several concepts. Concepts belonging to the same facet typically satisfy additional constraints such as being on the same level of organization or confirming to the same hyponym ("is-a") relationships. Faceted organization permit a concept to appear in multiple facets or none at all. Such an organization differs from the fundamental modeling assumption of most statistical topic models. How do we evaluate and select an appropriate model for a given analytic task? What are the implications for visualization and interaction design?

6.2.4 Deployment of Machine Learning Algorithms

Many of the techniques that I developed in this dissertation focused on supporting the evaluation of statistical topic models. More than providing diagnostic feedback, can we design visual analysis tools to encourage best practices?

Many of my tools were developed for experience practitioners — machine learning researchers and investigators who were familiar with the inner workings of a statistical topic model. My work also benefited from close collaborations with model builders

who helped investigate and modify model designs. How do we design topical curation or data analysis tools for users without computing technical backgrounds? How do we turn prototypes, such as Termite, into effective tools and put them in the hands of anyone who wish to perform topic modeling?

6.3 Closing Remarks

Model-driven visual data analysis draws on the work of multiple disciplines including information visualization, human-computer interaction, and machine learning. This dissertation sets an example on how to integrate such a diverse set of techniques into an effective tool. I envision that stronger collaboration among these disciplines will enable us to better understand and explore the growing of amount of data that we face.

Appendix A

Derivations and Implementations

A.1 Mixing as a Convolution Operator

Since events in a Bernoulli process are considered independent, I can re-arrange the order of events without affecting the expected outcome. When computing the number of expected topic-concept matches for the combined definitive and noise charts, I re-arrange all the definitive events to occur first and the noise events later.

Let $\{x^k\}$ be a series of Bernoulli events for $k \geq 1$ where x^k is the probability of observing a positive outcome for event k. I represent X^k as 2-vector $[1 - x^k, x^k]$. Let P^k be the multinomial distribution representing the observed cumulative outcome of the first k events where $P^k(i)$ is the probability that I observed exactly i positive outcomes for the first k events. I represent P^k as an $(k + 1)$-vector with entries $[P^k(0), P^k(1), \cdots, P^k(k)]$. I prove by induction, that $P^{k+1} = P^k * X^{k+1}$.

As the base case:

$$P^0 = 1$$
$$P^1 = X^1 = 1 * X^1 = P^0 * X^1$$

For the inductive step:

$$\begin{aligned}
P_i^{k+1} &= P_{i-1}^k \cdot x^{k+1} + P_i^k \cdot (1 - x^{k+1}) \\
&= P_{i-1}^k \cdot X_1^{k+1} + P_i^k \cdot X_0^{k+1} \\
&= \sum_{t=0}^{1} P_{i-t}^k \cdot X_t^{k+1} \\
P^{k+1} &= P^k * X^{k+1}
\end{aligned}$$

Let $P^{j,k}$ represent the observed cumulative outcome for events j to k (inclusive). Since convolution is communicative:

$$\begin{aligned}
P^{0,n} &= P^{0,k} * X^{k+1} * X^{k+2} * \cdots * X^n \\
&= P^{0,k} * P^{k+1,n}
\end{aligned}$$

It follows that the expected topical-concept matches for the combined chart is:

$$P_{\text{combined}} = P_{\text{definitive}} * P_{\text{noise}}$$

A.2 Setting k and Solving for γ

By construction, the distributions P and P_{noise} have the same mean. I arbitrarily choose k so that $P_{\text{definitive}}$ has the same mean as P. For non integer values of k, $P_{\text{definitive}}$ is zero everywhere except for two values, $P_{\text{definitive}}(\lfloor k \rfloor) = \lceil k \rceil - k$ and $P_{\text{definitive}}(\lceil k \rceil) = k - \lfloor k \rfloor$.

Discrete convolution can be converted to matrix multiplication. I convert the "convolute by P_{noise}" operation into a Toeplitz matrix $A = A_{\text{noise}}$. Let $P' = P_{\text{definitive}}^{k(1-\gamma)}$.

$$\begin{aligned}
&\min_\gamma \mathrm{KL}(P' * P_{\text{noise}}^\gamma || P) \\
&\min_\gamma \mathrm{KL}(AP' || P) \\
&\min_\gamma P'^T A^T \log(AP') - P'^T A^T \log(P)
\end{aligned}$$

I apply gradient descent to determine the optimal value γ that minimizes the objective function.

A.3 Solving for P_{denoised}

Let $P'' = P_{\text{denoised}}$.

$$\min_{P''} \text{KL}(P'' * P_{\text{noise}} || P)$$
$$\min_{P''} \text{KL}(AP'' || P)$$
$$\min_{P''} P''^T A^T \log(AP'') - P''^T A^T \log(P)$$

I apply sequential quadratic programming to determine optimal vector P''. The above optimization involves both equality ($\sum_i P''_i = 1$) and inequality constraints ($0 \leq P''_i \leq 1$). To improve the speed of computation and reduce complexity, I apply barrier method to remove the inequality constraints. I modify the objective function accordingly.

$$P''^T A^T \log(AP'') - P''^T A^T \log(P) + e^{-\alpha P''} + e^{\alpha(1+P'')}$$

I perform 50 iterations and gradually increase α from 500 to 50000.

To ensure better convergence, I solve the linear system of equations $AP'' = P$, to obtain an initial solution $P''^{(0)}$. I clamp the values of $P''^{(0)}$ to within $[0, 1]$ and L^1 normalize the vector to ensure it's a valid probability distribution. I use the resulting vector as the initial solution.

Bibliography

[1] Loulwah Alsumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. Topic significance ranking of LDA generative models. In *European Conference on Machine Learning (ECML)*, pages 67–82, 2009.

[2] Saleema Amershi, Bongshin Lee, Ashish Kapoor, Ratul Mahajan, and Blaine Christian. CueT: Human-guided fast and accurate network alarm triage. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 157–166, 2011.

[3] Nicholas O Andrews and Edward A Fox. Recent developments in document clustering. Technical Report TR-07-35, Virginia Polytechnic Institute and State University, 2007.

[4] David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In *International Conference on Machine Learning (ICML)*, pages 25–32, 2009.

[5] Arthur U Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. On smoothing and inference for topic models. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 27–34, 2009.

[6] Ken Barker and Nadia Cornacchia. Using noun phrase heads to extract document keyphrases. In *Canadian Society on Computational Studies of Intelligence*, pages 40–52, 2000.

[7] Richard A. Becker and William S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[8] Benjamin B. Bederson and James D. Hollan. Pad++: A zooming graphical interface for exploring alternate interface physics. In *ACM Confernece on User Interface Software and Technology (UIST)*, pages 17–26, 1994.

[9] David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Conference on Neural Information Processing Systems (NIPS)*, 2004.

[10] David M. Blei and John D Lafferty. Dynamic topic models. In *International Conference on Machine Learning (ICML)*, pages 113–120, 2006.

[11] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(1):993–1022, 2003.

[12] Branimir Boguraev and Christopher Kennedy. Applications of term identification technology: Domain description and content characterisation. *Natural Language Processing*, 5(1):17–44, 1999.

[13] Johan Bollen, Alberto Pepe, and Huina Mao. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *International Conference on Weblogs and Social Media (ICWSM)*, pages 17–21, 2011.

[14] Katy Börner, Jeegar T Maru, and Robert L Goldstone. The simultaneous evolution of author and paper networks. *Proceedings of the National Academy of Sciences (PNAS)*, 101:5266–5273, 2004.

[15] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. In *IEEE Information Visualization Conference (InfoVis)*, pages 2301–2309, 2011.

[16] Kevin W. Boyack, Katy Börner, and Richard Klavans. Mapping the structure and evolution of chemistry research. In *International Conference of Scientometrics and Informetrics (ICSI)*, pages 112–123, 2007.

[17] Kevin W. Boyack, Ketan Mane, and Katy Börner. Mapping Medline papers, genes, and proteins related to melanoma research. In *IEEE Information Visualization Conference (InfoVis)*, pages 965–971, 2004.

[18] Kevin W. Boyack, David Newman, Russell J Duhon, Richard Klavans, Michael Patek, Joseph R Biberstine, Bob Schijvenaars, André Skupin, Nianli Ma, and Katy Börner. Clustering more than two million biomedical publications: Comparing the accuracies of nine text-based similarity approaches. *PLoS ONE*, 6(3):e18029, 2011.

[19] Kevin W. Boyack, Brian N Wylie, and George S Davidson. Domain visualization using VxInsight for science and technology management. *Journal of the American Society for Information Science and Technology (JASIST)*, 53(9):764–774, 2002.

[20] Thorsten Brants and Alex Franz. Web 1t 5-gram version 1, linguistic data consortium, philadelphia, 2006.

[21] Raluca Budiu, Christiaan Royer, and Peter L. Pirolli. Modeling information scent: A comparison of LSA, PMI and GLSA similarity measures on common tests and corpora. In *Large Scale Semantic Access to Content Conference (RIAO)*, pages 314–332, 2007.

[22] Orkut Buyukkokten, Hector Garcia-Molina, Andreas Paepcke, and Terry Winograd. Power browser: Efficient web browsing for PDAs. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, CHI '00, 2000.

[23] Orkut Buyukkokten, Oliver Kaljuvee, Hector Garcia-Molina, Andreas Paepcke, and Terry Winograd. Efficient web browsing on handheld devices using page and form summarization. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 20(1):82–115, 2002.

[24] Nan Cao, Jimeng Sun, Yu-Ru Lin, David Gotz, Shixia Liu, and Huamin Qu. FacetAtlas: Multifaceted visualization for rich text corpora. In *IEEE Transactions on Visualization and Computer Grahics (TVCG)*, volume 16, pages 1172–1181, 2010.

[25] The Carnegie Classification of Institutions of Higher Education. `http://classifications.carnegiefoundation.org/`.

[26] Claudio Carpineto, Stanislaw Osiński, Giovanni Romano, and Dawid Weiss. A survey of web clustering engines. *ACM Computing Surveys (CSUR)*, 41(3):17:1–17:38, 2009.

[27] Allison Chaney and David M. Blei. Visualizing topic models. In *Conference on Artificial Intelligence (AAAI)*, 2012.

[28] Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. Reading tea leaves: How humans interpret topic models. In *Conference on Neural Information Processing Systems (NIPS)*, pages 288–296, 2009.

[29] William G Chase. Spatial representations of taxi drivers. In D A Rogers and J A Sloboda, editors, *The Acquisition of Symbolic Skills*. Plenum Press, 1983.

[30] William G Chase and Herbert A Simon. Perception in chess. *Cognitive Psychology*, 4(1):55–81, 1973.

[31] Chaomei Chen. CiteSpace II: Detecting and visualizing emerging trends and transient patterns in scientific literature. *Journal of the American Society for Information Science and Technology (JASIST)*, 57(3):359–377, 2006.

[32] Michelene T H Chi, Paul J Feltovich, and Robert Glaser. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5(2):121–152, 1981.

[33] Jason Chuang, Sonal Gupta, Christopher D. Manning, and Jeffrey Heer. Topic model diagnostics: Assessing domain relevance via topical alignment. Working paper (in submission).

[34] Jason Chuang, Christopher D. Manning, and Jeffrey Heer. Interpretation and trust: Designing model-driven visualizations for text analysis. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 443–452, 2012.

[35] Jason Chuang, Christopher D. Manning, and Jeffrey Heer. Termite: Visualization techniques for assessing textual topic models. In *International Working Conference on Advanced Visual Interfaces (AVI)*, pages 74–77, 2012.

[36] Jason Chuang, Christopher D. Manning, and Jeffrey Heer. "without the clutter of unimportant words": Descriptive keyphrases for text visualization. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 19(3):19:1–19:29, 2012.

[37] Jason Chuang, Daniel Ramage, Daniel A. McFarland, Christopher D. Manning, and Jeffrey Heer. Large-scale examination of academic publications using statistical models. In *International Working Conference on Advanced Visual Interfaces (AVI): Workshop on Supporting Asynchronous Collaboration in Visual Analytics Systems*, 2012.

[38] P D Clough and B A Sen. Evaluating tagclouds for health-related information research. In *Health Info Management Research*, 2008.

[39] D A Cohn, Z Ghahramani, and M I Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

[40] Allan M Collins and Elizabeth F Loftus. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407–428, 1975.

[41] Christopher Collins, Sheelagh Carpendale, and Gerald Penn. DocuBurst: Visualizing document content using language structure. *Computer Graphics Forum*, 28(3):1039–1046, 2009.

[42] Christopher Collins, Fernanda B. Viégas, and Martin Wattenberg. Parallel Tag Clouds to explore and analyze faceted text corpora. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 91–98, 2009.

[43] James E Corter and Mark A Gluck. Explaining basic categories: Feature predictability and information. *Psychological Bulletin*, 111(2):291–303, 1992.

[44] P J Crossno, D M Dunlavy, and T M Shead. LSAView: A tool for visual exploration of latent semantic modeling. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 83–90, 2009.

[45] Weiwei Cui, Yingcai Wu, Shixia Liu, Furu Wei, Michelle X Zhou, and Huamin Qu. Context-preserving, dynamic word cloud visualization. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 42–53, 2010.

[46] Douglass R Cutting, David R Karger, and Jan O Pedersen. Constant interaction-time scatter/gather browsing of very large document collections. In *ACM SIGIR Conference on Research and Development on Information Retrieval (SIGIR)*, 1993.

[47] Béatrice Daille, Éric Gaussier, and Jean-Marc Langé. Towards automatic extraction of monolingual and bilingual terminology. In *International Conference on Computational Linguistics (COLING)*, pages 515–521, 1994.

[48] Félix de Moya-Anegón, Benjamín Vargas-Quesada, Zaida Chinchilla-Rodríguez, Elena Corera-Álvarez, Francisco J Munoz-Fernández, and Victor Herrero-Solana. Visualizing the marrow of science. *Journal of the American Society for Information Science and Technology (JASIST)*, 58(14):2167–2179, 2007.

[49] Stephanie Doane, Walter Kintsch, and Peter Polson. Modeling UNIX command production: What experts must know. Technical Report 90-1, University of Colorado, 1990.

[50] Susan T. Dumais. Latent semantic analysis. *Annual Review of Information Science and Technology (ARIST)*, 38(1):188–230, 2004.

[51] Ted Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[52] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed membership models of scientific publications. *Proceedings of the National Academy of Sciences (PNAS)*, 101:5220–5227, 2004.

[53] David K Evans, Judith L Klavans, and Nina Wacholder. Document processing with LinkIT. In *Recherche d'Informations Assistee par Ordinateur*, 2000.

[54] Jerry Alan Fails and Dan R Olsen Jr. Interactive machine learning. In *Intelligence User Interfaces (IUI)*, 2003.

[55] Julian J Faraway. *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. Chapman Hall/CRC, 2006.

[56] Paul J. Feltovich, Paul E. Johnson, James H. Moller, and David B. Swanson. *LCS: The role and development of medical knowledge in diagnostic expertise*, pages 275–319. Addison Wesley, 1984.

[57] Jenny Rose Finkel, Trond Grenager, and Christopher D Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 363–370, 2005.

[58] Matthew J Gardner, Joshua Lutes, Jeff Lund, Josh Hansen, Dan Walker, Eric Ringger, and Kevin Seppi. The Topic Browser: An interactive tool for browsing topic models. In *Conference on Neural Information Processing Systems (NIPS): Workshop on Challenges of Data Visualization*, 2010.

[59] Brynjar Gretarsson, John O'Donovan, Svetlin Bostandjiev, Tobias H Llerer, Arthur Asuncion, David Newman, Padhraic Smyth, and Tobias Höllerer. TopicNets: Visual analysis of large text corpora with topic modeling. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(2):23:1–23:26, 2012.

[60] Thomas L. Griffiths and Mark Steyvers. Prediction and semantic association. In *Conference on Neural Information Processing Systems (NIPS)*, pages 11–18, 2002.

[61] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences (PNAS)*, 101(1):5228–5235, 2004.

[62] David Hall, Daniel Jurafsky, and Christopher D Manning. Studying the history of ideas using topic models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 363–371, 2008.

[63] Taher Haveliwala. Topic-sensitive PageRank. In *Intenational World Wide Web Conference (WWW)*, 2002.

[64] Susan Havre, Beth Hetzler, and Lucy Nowell. ThemeRiver: Visualizing theme changes over time. In *IEEE Information Visualization Conference (InfoVis)*, page 115, 2000.

[65] Marti A. Hearst. TileBars: Visualization of term distribution information in full text information access. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 59–66, 1995.

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

[67] Marti A. Hearst. UIs for faceted navigation: Recent advances and remaining open problems. In *Symposium on Human-Computer Interaction and Information Retrieval (HCIR)*, pages 13–17, 2008.

[68] Marti A. Hearst. *Search User Interfaces*. Cambridge Press, 2009.

[69] Marti A. Hearst and Jan O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *ACM SIGIR Conference on Research and Development on Information Retrieval (SIGIR)*, pages 76–84, 1996.

[70] Jeffrey Heer and Michael Bostock. Crowdsourcing graphical perception: Using Mechanical Turk to assess visualization design. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 203–212, 2010.

[71] Gregor Heinrich. Parameter estimation for text analysis. Technical report, Fraunhofer IGD, 2009.

[72] Bruce William Herr II, Edmund M. Talley, Gully A P C Burns, David Newman, and Gavin LaRowe. The NIH visual browser: An interactive visualization of biomedical research. In *International Conference Information Visualisation (IV)*, pages 505–509, 2009.

[73] Thomas Hofmann. Probabilistic latent semantic indexing. In *ACM SIGIR Conference on Research and Development on Information Retrieval (SIGIR)*, pages 50–57, 1999.

[74] Anette Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 216–223, 2003.

[75] Edwin L Hutchins, James D Hollan, and Donald A Norman. Direct manipulation interfaces. *Human-Computer Interaction*, 1:311–338, 1985.

[76] Tomoharu Iwata, Takeshi Yamada, and Naonori Ueda. Probabilistic latent semantic visualization: Topic model for visualizing documents. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD)*, pages 363–371, 2008.

[77] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[78] John S Justeson and Slava M Katz. Technical terminology: Some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(1):9–27, 1995.

[79] Weimao Ke, Cassidy R Sugimoto, and Javed Mostafa. Dynamicity vs. effectiveness: Studying online clustering for scatter/gather. In *ACM SIGIR Conference*

on Research and Development on Information Retrieval (SIGIR), pages 19–26, 2009.

[80] Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In *ACL SemEval Workshop*, 2010.

[81] Chunyu Kit and Xiaoyue Liu. Measuring mono-word termhood by rank difference via corpus comparison. *Terminology*, 14(2):204–229, 2008.

[82] Dan Klein and Christopher D Manning. Accurate unlexicalized parsing. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 423–430, 2003.

[83] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[84] George Lakoff. *Women, fire and dangerous things: What categories reveal about the mind*. London, UK, 1987.

[85] Thomas K. Landauer and Susan T. Dumais. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240, 1997.

[86] Thomas K. Landauer, Darrell Laham, and Marcia Derr. From paragraph to graph: Latent semantic analysis for information visualization. *Proceedings of the National Academy of Sciences (PNAS)*, 101:5214–5219, 2004.

[87] Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. Automatic labelling of topic models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1536–1545, 2011.

[88] Michael Laver, Kenneth Benoit, and Trinity College. Extracting policy positions from political texts using words as data. *American Political Science Review*, pages 311–331, 2003.

[89] Bongshin Lee, Greg Smith, George G Robertson, Mary Czerwinski, and Desney S Tan. FacetLens: Exposing trends and relationships to support sensemaking within faceted datasets. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1293–1302, 2009.

[90] Innar Liiv. Seriation and matrix reordering methods: An historical overview. *Statistical Analysis and Data Mining*, 3(2):70–91, 2010.

[91] K Lin and Ravikuma Kondadadi. A similarity-based soft clustering algorithm for documents. In *DASFAA: Database Systems for Advanced Applications*, pages 40–47, 2001.

[92] Xia Lin. Visualization for the document space. In *IEEE Conference on Visualization (Vis)*, pages 274–281, 1992.

[93] Patrice Lopez and Laurent Romary. HUMB: Automatic key term extraction from scientific articles in GROBID. In *ACL SemEval Workshop*, 2010.

[94] Hans Peter Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[95] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[96] Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of English: The Penn Treebank. *Computional Linguistics*, 19(2):313–330, 1993.

[97] M S Mayzner and R F Gabriel. Information "chunking" and short-term retention. *Journal of Psychology: Interdisciplinary and Applied*, 56(1):161–164, 1963.

[98] Andrew Kachites McCallum. MALLET: A machine learning for language toolkit. `http://mallet.cs.umass.edu`, 2002.

[99] William T McCormick, Paul J Schweitzer, and Thomas W White. Problem decomposition and data reorganization by a clustering technique. *Operations Research*, 20(5):993–1009, 1972.

[100] Daniel A. McFarland, Daniel Ramage, Jason Chuang, Jeffrey Heer, Christopher D. Manning, and Daniel Jurafsky. Differentiating language usage through topic models. Working paper (accepted).

[101] Olena Medelyan and Ian H Witten. Thesaurus based automatic keyphrase indexing. In *ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 296–297, 2006.

[102] Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. Automatic labeling of multinomial topic models. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD)*, pages 490–499, 2007.

[103] George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97, 1956.

[104] George A Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[105] David Mimno. Reconstructing pompeian households. In *Conference on Neural Information Processing Systems (NIPS): Workshop on Applications of Topic Models*, 2009.

[106] David Mimno, Hanna M. Wallach, Edmund M. Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 262–272, 2011.

[107] Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223, 2001.

[108] Burt Monroe, Michael Colaresi, and Kevin Quinn. Fightin' words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403, 2008.

[109] T Munzner. A nested process model for visualization design and validation. In *IEEE Information Visualization Conference (InfoVis)*, pages 921–928, 2009.

[110] Claudiu Cristian Musat, Julien Velcin, Stefan Trausan-Matu, and Marian-Andrei Rizoiu. Improving topic evaluation using conceptual knowledge. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 1866–1871, 2011.

[111] Ramesh M. Nallapati, Amr Ahmed, Eric P Xing, and William W Cohen. Joint latent topic models for text and citations. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD)*, pages 542–550, 2008.

[112] Ramesh M. Nallapati, Daniel A. McFarland, and Christopher D Manning. TopicFlow model: Unsupervised learning of topic-specific influences of hyperlinked documents. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[113] David Newman, Arthur Asuncion, Chaitanya Chemudugunta, Vasanth Kumar, Padhraic Smyth, and Mark Steyvers. Exploring large document collections using statistical topic models. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD): Demonstration*, 2006.

[114] David Newman, Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. Analyzing entities and topics in news articles using statistical topic models. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 93–104, 2006.

[115] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic evaluation of topic coherence. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 100–108, 2010.

[116] David Newman, Youn Noh, Edmund M. Talley, Sarvnaz Karimi, and Timothy Baldwin. Evaluating topic models for digital libraries. In *ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 215–224, 2010.

[117] Donald A Norman and Stephen W Draper. *User Centered System Design; New Perspectives on Human-Computer Interaction*. CRC Press, Hillsdale, NJ, 1986.

[118] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, 1999.

[119] Palantir Technologies. `http://www.palantirtech.com`.

[120] Rob Procter, Farida Vis, Alex Voss, Marta Cantijoch, Yana Manykhina, Mike Thelwall, Rachel Gibson, Andrew Hudson-Smith, and Steven Gray. Behind the rumours: how we built our Twitter riots interactive. `http://www.guardian.co.uk/news/datablog/2011/dec/08/twitter-riots-interactive`, 2011.

[121] Rob Procter, Farida Vis, Alex Voss, Marta Cantijoch, Yana Manykhina, Mike Thelwall, Rachel Gibson, Andrew Hudson-Smith, and Steven Gray. Riot rumours: How misinformation spread on Twitter during a time of crisis. `http://www.guardian.co.uk/uk/interactive/2011/dec/07/london-riots-twitter`, 2011.

[122] Dragomir R. Radev, Mark Thomas Joseph, Bryan Gibson, and Pradeep Muthukrishnan. A bibliometric and network analysis of the field of computational linguistics. *Journal of the American Society for Information Science and Technology (JASIST)*, 2009.

[123] Dragomir R Radev, Pradeep Muthukrishnan, and Vahed Qazvinian. The ACL anthology network corpus. In *ACL Workshop on NLP and IR for Digital Libraries*, 2009.

[124] Daniel Ramage. Stanford topic modeling toolbox. `http://nlp.stanford.edu/software/tmt/`.

[125] Daniel Ramage. *Studying people, organizations, and the web with statistical text models*. PhD thesis, 2011.

[126] Daniel Ramage, Jason Chuang, Christopher D. Manning, and Daniel A. McFarland. Mapping three decades of intellectual change in academia with statistical topic models. Working paper (in preparation for submission).

[127] Daniel Ramage, Susan T. Dumais, and Dan Liebling. Characterizing microblogs with topic models. In *International Conference on Weblogs and Social Media (ICWSM)*, pages 130–137, 2010.

[128] Daniel Ramage, David Hall, Ramesh M. Nallapati, and Christopher D Manning. Labeled LDA: A supervised topic model for credit attribution in multi-label corpora. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 248–256, 2009.

[129] Daniel Ramage, Christopher D. Manning, and Susan T. Dumais. Partially labeled topic models for interpretable text mining. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD)*, pages 457–465, 2011.

[130] Daniel Ramage, Evan Rosen, Jason Chuang, Christopher D. Manning, and Daniel A. McFarland. Topic modeling for the social sciences. In *Conference on Neural Information Processing Systems (NIPS): Workshop on Applications of Topic Models*, 2009.

[131] Paul Rayson and Roger Garside. Comparing corpora using frequency profiling. In *Workshop on Comparing Corpora (WCC)*, pages 1–6, 2000.

[132] J S Risch, D B Rex, S T Dowson, T B Walters, R A May, and B D Moon. The STARLIGHT information visualization system. In *IEEE Information Visualization Conference (InfoVis)*, 1997.

[133] A W Rivadeneira, Daniel M Gruen, Michael J Muller, and David R Millen. Getting our head in the clouds: Toward evaluation studies of tagclouds. In

ACM SIGCHI Conference on Human Factors in Computing Systems (CHI), 2007.

[134] S E Robertson, C J van Rijsbergen, and M F Porter. Probabilistic models of indexing and searching. In R N Oddy, S E Robertson, C J van Rijsbergen, and P W Williams, editors, *Information Retrieval Research*, pages 35–56, 1981.

[135] Eleanor Rosch. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104(3):192–233, 1975.

[136] Eleanor Rosch and Carolyn B Mervis. Family resemblances: Studies in the internal structure of categories. *Cognitive Psychology*, 7:573–605, 1975.

[137] Eleanor Rosch, Carolyn B. Mervis, Wayne D. Gray, David M. Johnson, and Penny Boyes-Braem. Basic objects in natural categories. *Cognitive Psychology*, 8(3):382–439, 1976.

[138] M Rosvall and C T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences (PNAS)*, 105:1118–1123, 2008.

[139] Daniel M Russell, Mark J Stefik, Peter L. Pirolli, and Stuart K. Card. The cost structure of sensemaking. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 269–276, 1993.

[140] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. In *Information Processing and Management*, pages 513–523, 1988.

[141] Gerard Salton, Andrew Wong, and Chung Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[142] Pamela E. Sandstrom. Scholarly communication as a socioecological system. *Scientometrics*, 51(3):573–605, 2002.

[143] Ariel S. Schwartz and Marti A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Symposium on Biocomputing (PSB)*, 2003.

[144] Lei Shi, Furu Wei, Shixia Liu, Li Tan, Xiaoxiao Lian, and Michelle X Zhou. Understanding text corpora with multiple facets. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 99–106, 2010.

[145] Ben Shneiderman and Martin Wattenberg. Ordered treemap layouts. In *IEEE Information Visualization Conference (InfoVis)*, pages 73–78, 2001.

[146] James Sinclair and Michael Cardew-Hall. The folksonomy tag cloud: When is it useful? *Journal of Information Science*, 34(1):15–29, 2008.

[147] Neil R Smalheiser, Vetle I Torvik, and Wei Zhou. Arrowsmith two-node search interface: A tutorial on finding meaningful links between two disparate sets of articles in MEDLINE. *Computer Methods and Programs in Biomedicine*, 94(2):190–197, 2009.

[148] John Stasko, Carsten Görg, and Zhicheng Liu. Jigsaw: Supporting investigative analysis through interactive visualization. In *IEEE Information Visualization Conference (InfoVis)*, pages 118–132, 2008.

[149] John Stasko, Carsten Görg, Zhicheng Liu, and K Singhal. Jigsaw: Supporting investigative analysis through interactive visualization. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2007.

[150] Keith Stevens, Philip Kegelmeyer, David Andrzejewski, and David Buttler. Exploring topic coherence over many models and many topics. In *Joint Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2012.

[151] Emilia Stoica and Marti A. Hearst. Automating creation of hierarchical faceted metadata structures. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 244–251, 2007.

[152] Edmund M. Talley, David Newman, David Mimno, Bruce William Herr II, Hanna M. Wallach, Gully A P C Burns, A G Miriam Leenders, and Andrew McCallum. Database of NIH grants using machine-learned categories and graphical clustering. *Nature Methods*, 8(6), 2011.

[153] James W Tanaka and Marjorie Taylor. Object categories and expertise: Is the basic level in the eye of the beholder? *Cognitive Psychology*, 23(3):457–482, 1991.

[154] Vinh Tuan Thai, Siegfried Handschuh, and Stefan Decker. IVEA: An information visualization tool for personalized exploratory document collection analysis. In *European Semantic Web Conference (ESWC)*, pages 139–153, 2008.

[155] James J. Thomas and Kristin A. Cook, editors. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Press, 2005.

[156] Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 252–259, 2003.

[157] Jennifer Trant. Social classification and folksonomy in art museums: Early data from the steve.museum tagger prototype. *American Society for Information Science and Technology (ASIST)*, 2006.

[158] Peter D. Turney. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336, 2000.

[159] Amos Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.

[160] Barbara Tversky. Parts, partonomies, and taxonomies. *Developmental Psychology*, 25(6):983–995, 1989.

[161] Frank van Ham, Martin Wattenberg, and Fernanda B. Viégas. Mapping text with phrase nets. In *IEEE Information Visualization Conference (InfoVis)*, pages 1169–1176, 2009.

[162] Fernanda B. Viégas, Scott Golder, and Judith Donath. Visualizing email content: Portraying relationships from conversational histories. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 979–988, 2006.

[163] Fernanda B. Viégas and Martin Wattenberg. Timelines: Tag clouds and the case for vernacular visualization. *Interactions*, 15(4):49–52, 2008.

[164] Fernanda B. Viégas, Martin Wattenberg, and Jonathan Feinberg. Participatory visualization with Wordle. *IEEE Information Visualization Conference (InfoVis)*, 15(6):1137–1144, 2009.

[165] Fernanda B. Viégas, Martin Wattenberg, Frank van Ham, Jesse Kriss, and Matt McKeon. ManyEyes: A site for visualization at internet scale. In *IEEE Information Visualization Conference (InfoVis)*, pages 1121–1128, 2007.

[166] Hanna M. Wallach, David Mimno, and Andrew McCallum. Rethinking LDA: Why priors matter. In *Conference on Neural Information Processing Systems (NIPS)*, 2009.

[167] Hanna M. Wallach, Iain Murray, Ruslan Salakhutdinov, and David Mimno. Evaluation methods for topic models. In *International Conference on Machine Learning (ICML)*, pages 1105–1112, 2009.

[168] Franz Wanner, Christian Rohrdantz, Florian Mansmann, Daniela Oelke, and Daniel A Keim. Visual sentiment analysis of RSS news feeds featuring the US presidental election in 2008. In *Visual Interfaces to the Social and Semantic Web Workshop (VISSW)*, pages 1–7, 2009.

[169] Rob J C Watt. The web concordances. `http://www.concordancesoftware.co.uk`.

[170] Martin Wattenberg and Fernanda B. Viégas. The Word Tree, an interactive visual concordance. In *IEEE Information Visualization Conference (InfoVis)*, pages 1221–1228, 2008.

[171] Furu Wei, Shixia Liu, Yangqiu Song, Shimei Pan, Michelle X Zhou, Weihong Qian, Lei Shi, Li Tan, and Qiang Zhang. TIARA: A visual exploratory text analytic system. In *ACM SIGKDD Conference on Knowledge Discovery and Data (KDD)*, 2010.

[172] Zhen Wen and Ching-yung Lin. Towards finding valuable topics. In *SIAM International Conference on Data Mining (ICDM)*, pages 720–731, 2010.

[173] Leland Wilkinson and Michael Friendly. The history of the cluster heat map. *The American Statistician*, 63(2):179–184, 2009.

[174] J A Wise, James J. Thomas, K Pennock, D Lantrip, M Pottier, A Schur, and V Crow. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In *IEEE Information Visualization Conference (InfoVis)*, pages 51–58, 1995.

[175] Pak Chung Wong, Beth Hetzler, Christian Posse, Mark Whiting, Susan Havre, Nick Cramer, Anuj Shah, Mudita Singhal, Alan Turner, and James J. Thomas. IN-SPIRE contest entry. In *IEEE Information Visualization Conference (InfoVis)*, 2004.

[176] Christopher C Yang and Fu Lee Wang. Fractal summarization for mobile devices to access large documents on the web. In *International Conference on World Wide Web (WWW)*, pages 215–224, 2003.

[177] Koji Yatani, Michael Novati, Andrew Trusty, and Khai N. Truong. Review Spotlight: A user interface for summarizing user-generated reviews using adjective-noun word pairs. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1541–1550, 2011.

[178] Oren Zamir and Oren Etzioni. Grouper: A dynamic clustering interface to web search results. *Computer Networks*, 31(1116):1361–1374, 1999.